

ЗАТВЕРДЖУЮ



Декан факультету інформаційних технологій

Тетяна ГОВОРУЩЕНКО
Ім'я, ПРІЗВИЩЕ

09

2025 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

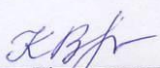
Основи програмної інженерії та тестування програмного забезпечення

Назва дисципліни

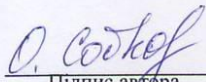
Галузь знань – 12 Інформаційні технології
Спеціальність – 122 Комп'ютерні науки
Рівень вищої освіти – Перший (бакалаврський)
Освітньо-професійна програма – Комп'ютерні науки
Обсяг дисципліни – 5 кредитів ЄКТС
Шифр дисципліни – ОПП.10
Мова навчання – Українська
Статус дисципліни: Обов'язкова (професійної підготовки)
Факультет – Інформаційних технологій
Кафедра – Комп'ютерних наук

Форма здобуття освіти	Курс	Семестр	Загальний обсяг		Кількість годин						Курсовий проєкт	Курсова робота	Форма семестрового контролю	
					Аудиторні заняття					Самостійна робота, у т.ч. ІРС			Залік	Іспит
			Кредити ЄКТС	Години	Разом	Лекції	Лабораторні роботи	Практичні заняття	Семінарські заняття					
Д	2	4	5	150	66	32	34			84				+
Разом ДФН			5	150	66	32	34			84				4

Робоча програма складена на основі освітньо-професійної програми «Комп'ютерні науки» за спеціальністю 122 «Комп'ютерні науки»

Робоча програма складена 
Підпис автора

асистент каф. КН Валерія КЛІМЕНКО
Науковий ступінь, вчене звання, ім'я, ПРІЗВИЩЕ автора


Підпис автора

д-р філософії Олена СОВКО
Науковий ступінь, вчене звання, ім'я, ПРІЗВИЩЕ автора


Підпис автора

канд. техн. наук, доц. Олександр МАЗУРЕЦЬ
Науковий ступінь, вчене звання, ім'я, ПРІЗВИЩЕ автора

Схвалена на засіданні кафедри Комп'ютерних наук

Протокол від 29.08.2025 № 1.

Зав. кафедри 
Підпис

Олександр БАРМАК
Ім'я, ПРІЗВИЩЕ

Робоча програма розглянута та схвалена вченою радою факультету інформаційних технологій

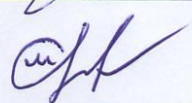
Голова вченої ради факультету

Підпис

Тетяна ГОВОРУЩЕНКО
Ім'я, ПРІЗВИЩЕ

Хмельницький 2025

ЛИСТ ПОГОДЖЕННЯ

Посада	Назва кафедри	Підпис	Ініціали, ПРІЗВИЩЕ
Завідувач кафедри, д-р. техн. наук, проф.	Комп'ютерних наук		Олександр БАРМАК
Гарант освітньо-професійної програми, канд. техн. наук, доц.	Комп'ютерних наук		Олександр МАЗУРЕЦЬ

3. Пояснювальна записка

Дисципліна «Основи програмної інженерії та тестування програмного забезпечення» входить до циклу обов'язкових професійних дисциплін при навчанні студентів на першому (бакалаврському) освітньому рівні спеціальності 122 «Комп'ютерні науки». Курс розглядає застосування методологій та засобів проектування інформаційних систем, а також різних видів та методологій тестування для оцінювання результатів розробки інформаційних систем і можливостей їх використання відповідно до вимог замовника.

Пререквізити – Алгоритмізація та програмування (ОЗП.05), Теорія алгоритмів (ОЗП.08), Проектування баз даних (ОФП.07), Об'єктно-орієнтоване проектування (ОФП.08).

Кореквізити – Методи та системи штучного інтелекту (ОФП.01), Методи та системи штучного інтелекту (курсний проєкт) (ОФП.02), Вебтехнології (ОФП.03), Вебтехнології (курсний проєкт) (ОФП.04), Технології розподілених систем та паралельних обчислень (ОФП.11), Системне та мережеве програмне забезпечення (ОФП.14), Управління ІТ-проєктами (ОФП.15), Проєктно-технологічна практика (ОФП.16), Професійна практика (ОФП.17), Кваліфікаційна робота (ОФП.18).

Відповідно до освітньої програми дисципліна сприяє забезпеченню:

компетентностей: здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів інформаційних технологій і характеризується комплексністю та невизначеністю умов (ІК); здатність до абстрактного мислення, аналізу та синтезу (ЗК 01); здатність застосовувати знання у практичних ситуаціях (ЗК 02); здатність працювати в команді (ЗК 09); здатність бути критичним і самокритичним (ЗК 10); здатність оцінювати та забезпечувати якість виконуваних робіт (ЗК 12); здатність застосовувати теоретичні та практичні основи методології та технології моделювання для дослідження характеристик і поведінки складних об'єктів і систем, проводити обчислювальні експерименти з обробкою й аналізом результатів (ФК 07); здатність проєктувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління (ФК 08); здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника (ФК 10); здатність до аналізу та функціонального моделювання бізнес-процесів, побудови та практичного застосування функціональних моделей організаційно-економічних і виробничо-технічних систем, методів оцінювання ризиків їх проектування (ФК 15).

програмних результатів навчання: розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук (ПРН 09); володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проєктну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт) (ПРН 11); застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем (ПРН 14).

Мета дисципліни. формування у студентів навичок застосування методологій та засобів проектування інформаційних систем, а також різних видів та методологій тестування для оцінювання результатів розробки інформаційних систем і можливостей їх використання відповідно до вимог замовника.

Предмет дисципліни. Інженерія та тестування програмного забезпечення.

Завдання дисципліни. По завершенні курсу студент має володіти всіма основними навичками, необхідними для налагодження та тестування програмних засобів інформаційних систем, застосовувати знання методології та засоби проектування інформаційних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем.

Результати навчання. Студент, який успішно завершив вивчення дисципліни, повинен: вміти застосовувати методології та засоби проектування інформаційних систем, застосовувати різні види та методології тестування для оцінювання результатів розробки інформаційних систем і можливостей їх використання відповідно до вимог замовника.

4. Структура залікових кредитів дисципліни

Назва розділу	Кількість годин, відведених на:		
	лекції	лабораторні заняття	СРС
Розділ 1. Методології розробки ПЗ	4	4	8
Розділ 2. Засоби та середовища створення ПЗ	4	8	9
Розділ 3. Проектування інтерфейсу користувача.	2	4	5
Розділ 4. Моделювання програмного забезпечення	6	4	16
Розділ 5. Методи забезпечення та контролю якості ПЗ	16	16	46
Разом:	32	34	84

5. Програма навчальної дисципліни

5.1 Зміст лекційного курсу

Номер лекції	Перелік тем лекцій, їх анотації	Кількість годин
	<i>Розділ 1. Методології розробки ПЗ</i>	4
1	Основи програмної інженерії. Життєвий цикл розробки програмного забезпечення (SDLC). Технічне завдання. Основи програмної інженерії та проблеми розробки. Цільові об'єкти програмної інженерії. Життєвий цикл розробки ПЗ. Міжнародні стандарти життєвого циклу ПЗ. Структура програмного забезпечення. Моделі та методології розробки ПЗ. Методи та інструменти програмної інженерії. Розробка проєктної документації (техніко-економічне обґрунтування, технічне завдання) відповідно до вимог і обмежень замовника. Технічне завдання (Technical Specification). Літ.: [1] с. 13-20, [2], с. 7-17, [3], с. 129-142	2
2	Основи інженерії вимог до програмного забезпечення. Специфікація (SRS) та тестування (RST) програмних вимог. Види вимог до ПЗ. Етапи складання вимог до ПЗ. Функціональні та нефункціональні вимоги. Галузеві вимоги. Вимоги до програмного забезпечення. Аналіз потреб користувачів та збір вимог. Специфікація вимог до ПЗ (SRS). Компоненти SRS. Тестування вимог. Способи подання вимог до ПЗ. Літ.: [1] с. 22-26, [2], с. 7-17, [3], с. 182-185	2
	<i>Розділ 2. Засоби та середовища створення ПЗ</i>	4
3	Гнучкі методології створення програмного забезпечення (Scrum, Kanban). Системи керування проєктами (Jira, Trello). Управління життєвим циклом ПЗ. Гнучкі методології створення ПЗ. Переваги та недоліки методології гнучкої розробки. Сфери застосування Agile. Scrum. Заходи та артефакти Scrum. Kanban. Системи керування проєктами (Jira, Trello). Літ.: [1] с. 28-36, [2], с. 18-24	2
4	Системи керування версіями (GIT). Сервіси для спільної роботи (GitLab, GitHub, GitHub Desktop). Загальне про системи керування версіями документів. Типи систем керування версіями. Система керування версіями GIT. Вебсервіси спільної роботи GitLab та GitHub. Компоненти та команди GitHub. Поняття конфлікту злиття. Робота з програмним кодом за допомогою GitHub Desktop. Літ.: [1] с. 37-45, [2], с. 25-32, [3], с. 9-25	2
	<i>Розділ 3. Проєктування інтерфейсу користувача</i>	2
5	Проєктування інтерфейсу користувача. UI та UX. Інструменти прототипування (Figma). Призначення та типи інтерфейсів користувача. Введення до UI/UX. Етапи проєктування UI. Дослідження потреб та очікувань користувачів для формування UX. Прототипування та креативність. Види прототипів. Засоби прототипування (Figma, Sketch, Axure RP, Adobe Photoshop). Компоненти дизайну (сітка, колір, типографіка). Принципи UX Flow та UI Fluidity Importance. Літ.: [1] с. 46-55, [2], с. 33-42	2
	<i>Розділ 4. Моделювання програмного забезпечення</i>	4
6	Основні поняття проєктування архітектури ПЗ. Проєктування архітектури ПЗ. Принципи проєктування архітектури. Послідовність проєктування архітектури ПЗ. Критерії для оцінки спроектованої архітектури. Концептуальна модель ПЗ. Структура системи та її властивості. Типи структур систем ПЗ. Шабини проєктування програмного забезпечення. Літ.: [1] с. 57-65, [2], с. 43-59, [3], с. 27-47	2
7	Моделювання ПЗ. Мови та засоби моделювання ПЗ. Моделювання бізнес-процесів. Підходи до моделювання ПЗ. Методи структурного аналізу систем Мови моделювання (UML, SysML, ArchiMate). Засоби інформаційного моделювання (діаграми сценаріїв, взаємодії, станів, класів, компонентів, часові). Стандарт моделювання бізнес-процесів BPMN. Особливості аналізу та функціонального моделювання бізнес-процесів. Літ.: [1] с. 67-74, [2], с. 43-59	2
8	Методології та технології моделювання ПЗ (SADT, IDEF, DFD). CASE-технології та CASE-засоби. Методології моделювання ПЗ. Особливості дослідження характеристик і поведінки складних об'єктів і систем. Моделювання потоків даних (DFD). Нотації DFD-діаграм. Функціональне моделювання бізнес-процесів (SADT, IDEF0). Методи оцінювання ризиків проєктування функціональних моделей. CASE-технології. Класифікація CASE-засобів. Переваги CASE-технологій. Сучасні засоби автоматизації моделювання ПЗ. Літ.: [1] с. 76-86, [2], с. с. 43-59	2
	<i>Розділ 5. Методи забезпечення та контролю якості ПЗ</i>	16
9	Оцінювання та забезпечення якості ПЗ. Метрики і стандарти якості ПЗ. Верифікація та валідація ПЗ. Поняття якості ПЗ. Метрики і стандарти якості ПЗ. Стандарти ISO 9000/ IEEE.	2

	Завдання тестування ПЗ. Основні принципи тестування. Система моніторингу, контролю та забезпечення якості. Шляхи оцінювання та забезпечення якості ПЗ (Verification and Validation). Тестувальник і QA інженер. Літ.: [1] с. 88-92, [2], с. 60-72, [4], с. 4-11	
10	Тестування ПЗ. План тестування (Test Plan) та робота з дефектами. Тестування ПЗ. Базова термінологія тестування ПЗ. Види тестування. Життєвий цикл тестування (STLC). Документація з тестування ПЗ. План тестування (Test Plan). Причини появи дефектів ПЗ. Пошук та документування дефектів у ПЗ. Пріоритетність та критичність дефектів. Життєвий цикл дефектів. Інструментарій Jira для документування дефектів. Літ.: [1] с. 94-101, [2], с. 60-72, [4], с. 4-11	2
11	Підходи до тестування. Рівні тестування. Підходи до тестування. Ознайомлювальне тестування, тестування за сценарієм. Ручне і автоматизоване тестування. Позитивне і негативне тестування. Функціональний та структурний підходи до тестування. Рівні тестування. Модульне тестування (Unit testing). Інтеграційне тестування (Integration testing). Системне тестування (System testing). Приймальне тестування (Acceptance testing). Регресивне тестування та комбінування рівнів тестування. Тестування продуктивності. Інструментальні засоби тестування. Класифікація помилок. Літ.: [1] с. 102-107, [2], с. 60-72	2
12	Засоби тестування (Checklist, Test Case та Test Case Suite). Ручне тестування. Типи ручного тестування. Переваги та недоліки. Процес ручного тестування. Правила ручного тестування. Інструменти для ручного тестування. Використання Checklist для планування розробки тест-кейсів Test Case. Приклади Checklist. Тестування поведінки ПЗ засобами Test Case. Життєвий цикл Test Case. Атрибути (поля) Test Case. Приклади Test Case. Test Case Suite. Приклади Test Case Suite. Інструментальні засоби для створення Test Case (TestLink). Літ.: [1] с. 108-120, [2], с. 73-86, [3], с. 164-170, [3], с. 187-189, [4], с. 24-33	2
13	Автоматизація тестування. Сервера постійної інтеграції. Автоматизоване тестування ПЗ. Переваги та недоліки. Інструменти та технології автоматизації тестування. Генератори тестів. Структура тестового набору для автоматизованого прогону. Додатки для автоматизованого прогону тестів. Сервера постійної інтеграції (Hudson, CruiseControl). Порівняння витрат і ефективності різних методів тестування. Літ.: [1] с. 121-125, [2], с. 73-87	2
14	Застосування фреймворків для модульного тестування засобами Unit Test. Основні принципи модульного тестування. Типи модульних тестів. Переваги та недоліки модульного тестування. Інструменти модульного тестування. Фреймворки для модульного тестування. Тестування ПЗ за допомогою Unit Test. Правила написання Unit Test. Генерація звітів з тестування Unit Test. Літ.: [1] с. 126-131, [2], с. 87-100, [4], с. 42-45	2
15	Тестування інтерфейсу користувача (UI). Завдання і цілі тестування UI. Типи вимог до UI. Функціональне тестування UI. Тестування зручності використання UI. Методи тестування. Ітеративний підхід до покращення UI/UX. Проблеми тестування UI. Літ.: [1] с. 132-137, [2], с. с. 87-100, [4], с. 46-53	2
16	Тестування ПЗ засобами штучного інтелекту. Напрямки застосування ШІ для тестування ПЗ. Автоматизоване візуальне тестування ПЗ (Automated Visual Software Testing). Інтелектуальне програмування агентів тестування. IntelliTest, як засіб для автоматизації тестування з використанням ШІ. Літ.: [1] с. 138-145, [2], с. с. 87-100	2
Разом за семестр:		32

5.2 Зміст лабораторних занять

№ п/п	Тема лабораторного заняття	Кількість годин
1	Створення технічного завдання для розробки ПЗ Літ.: [1] с. 13-20, [1] с. 22-26, [2], с. 7-17, [2], с. 7-17, [3], с. 129-142, [3], с. 182-185	4
2	Використання Jira для керування проектами за методологією Scrum. Літ.: [1] с. 28-36, [2], с. 18-24	4
3	Використання GitHub та GitHub Desktop для керування версіями та спільної роботи. Літ.: [1] с. 37-45, [2], с. 25-32, [3], с. 9-25	4
4	Проектування та креативне прототипування UI з використанням Figma Літ.: [1] с. 46-55, [2], с. 33-42	4
5	Прикладне моделювання потоків даних (DFD) та функціональних моделей бізнес-процесів (IDEF0). Літ.: [1] с. 57-65, [1] с. 67-74, [1] с. 76-86, [2], с. 43-59, [3], с. 27-47	4
6	Пошук та документування дефектів у ПЗ.	4

	Літ.: [1] с. 88-92, [1] с. 94-101, [2], с. 60-72, [4], с. 4-11	
7	Розробка Checklist, Test Case та Test Case Suite для тестування ПЗ. Літ.: [1] с. 108-120, [2], с. 73-86, [3], с. 164-170, [3], с. 187-189, [4], с. 24-33	4
8	Розробка Unit Test та тестування ПЗ за допомогою фреймворків. Літ.: [1] с. 126-131, [2], с. 87-100, [4], с. 42-45	6
Разом:		34

5.3 Зміст самостійної (у т.ч. індивідуальної) роботи здобувача вищої освіти

Самостійна робота студентів усіх форм здобуття освіти полягає у систематичному опрацюванні програмного матеріалу з відповідних джерел інформації, підготовці до лабораторних занять, контрольних робіт, виконанні індивідуальних завдань. Крім цього до послуг студентів сторінка навчальної дисципліни у Модульному середовищі для навчання, де розміщені Робоча програма дисципліни та необхідні документи з її навчально-методичного забезпечення.

Номер тижня	Вид самостійної роботи	Кількість годин
1	Опрацювання теоретичного матеріалу з лекції 1, підготовка до виконання лабораторної роботи №1	4
2	Опрацювання теоретичного матеріалу з лекції 2, підготовка до захисту лабораторної роботи №1.	4
3	Опрацювання теоретичного матеріалу з лекції 3, підготовка до виконання лабораторної роботи №2, початок виконання ІЗ (завдання 1)	5
4	Опрацювання теоретичного матеріалу з лекції 4, підготовка до захисту лабораторної роботи №2, виконання ІЗ (завдання 1)	4
5	Опрацювання теоретичного матеріалу з лекції 5, підготовка до виконання лабораторної роботи №3, виконання ІЗ (завдання 1)	5
6	Опрацювання теоретичного матеріалу з лекції 6, підготовка до захисту лабораторної роботи №3, виконання ІЗ (завдання 1)	5
7	Опрацювання теоретичного матеріалу з лекції 7, підготовка до лабораторної роботи №4, виконання ІЗ (завдання 1).	5
8	Опрацювання теоретичного матеріалу з лекції 8, підготовка до захисту лабораторної роботи №3, виконання ІЗ (завдання 1). Підготовка до КР №1	6
9	Опрацювання теоретичного матеріалу з лекції 9, підготовка до виконання лабораторної роботи №4, початок виконання ІЗ (завдання 2)	5
10	Опрацювання теоретичного матеріалу з лекції 10, підготовка до захисту лабораторної роботи №4, виконання ІЗ (завдання 2)	4
11	Опрацювання теоретичного матеріалу з лекції 11, підготовка до виконання лабораторної роботи №5, виконання ІЗ (завдання 2)	4
12	Опрацювання теоретичного матеріалу з лекції 12, підготовка до захисту лабораторної роботи №5, виконання ІЗ (завдання 2)	4
13	Опрацювання теоретичного матеріалу з лекції 13, підготовка до виконання лабораторної роботи №6, початок виконання ІЗ (завдання 3)	5
14	Опрацювання теоретичного матеріалу з лекції 14, підготовка до захисту лабораторної роботи №6, виконання ІЗ (завдання 3). Підготовка до КР №2	6
15	Опрацювання теоретичного матеріалу з лекції 15, підготовка до виконання лабораторної роботи №7 та №8, початок виконання ІЗ (завдання 4). Підготовка до іспиту.	6
16	Опрацювання теоретичного матеріалу з лекції 16, підготовка до захисту лабораторних робіт №7 та №8, оформлення звіту по ІЗ (завдання 1-4). Підготовка до іспиту.	6
17	Опрацювання теоретичного матеріалу з лекції 17, підготовка до здачі ІЗ (завдання 1-4). Підготовка до іспиту.	6
Разом:		84

Примітки: КР – контрольна робота, ІЗ – індивідуальне завдання.

На самостійне опрацювання студентів виносяться визначені у методичних рекомендаціях до лабораторних занять та самостійної роботи питання з відповідних тем. Керівництво самостійною роботою та контроль за виконанням індивідуального завдання здійснюється викладачем згідно з розкладом консультацій у позаурочний час.

Вимоги до виконання індивідуального домашнього завдання викладені в Модульному середовищі для навчання на сторінці навчальної дисципліни.

6. Технології та методи навчання

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій та методів навчання, зокрема: лекції (з використанням методів візуалізації, проблемного й інтерактивного навчання, мотиваційних прийомів, інформаційно-комунікаційних технологій); лабораторні заняття (з використанням інструктування, демонстрування, розв'язування типових і прикладних задач, аналізу кейсів, ситуаційних завдань, елементів дискусії тощо); самостійна робота (опрацювання теоретичного матеріалу, підготовка до виконання лабораторних робіт, поточного та підсумкового контролю, виконання індивідуальних завдань), з використанням інформаційно-комп'ютерних технологій та технологій дистанційного навчання.

7. Методи контролю

Поточний контроль здійснюється під час аудиторних лабораторних занять, а також у дні проведення контрольних заходів, встановлених робочою програмою і графіком освітнього процесу, в т.ч. з використанням Модульного середовища для навчання. При цьому використовуються такі методи поточного контролю:

– оцінювання результатів виконання індивідуального завдання;

- оцінювання контрольних робіт;
- оцінювання результатів захисту лабораторних робіт.

При виведенні підсумкової семестрової оцінки враховуються результати як поточного контролю, так і підсумкового контролю, який проводиться з усього матеріалу дисципліни за білетами, попередньо розробленими і затвердженими на засіданні кафедри. Здобувач вищої освіти, який набрав з будь-якого виду навчальної роботи, суму балів нижчу за 60 відсотків від максимального балу, **не допускається** до семестрового контролю, поки не виконає обсяг роботи, передбачений Робочою програмою. Здобувач вищої освіти, який набрав позитивний середньозважений бал (60 відсотків і більше від максимального балу) з усіх видів поточного контролю і не склав іспит, вважається таким, який **має** академічну заборгованість. Ліквідація академічної заборгованості із семестрового контролю здійснюється у період екзаменаційної сесії або за графіком, встановленим деканатом відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ».

8. Політика дисципліни

Політика навчальної дисципліни загалом визначається системою вимог до здобувача вищої освіти, що передбачені чинними положеннями Університету про організацію і навчально-методичне забезпечення освітнього процесу. Зокрема, проходження інструктажу з техніки безпеки; відвідування занять з дисципліни є обов'язковим. За об'єктивних причин (підтверджених документально) теоретичне навчання за погодженням із лектором може відбуватись в он-лайн режимі. Успішне опанування дисципліни і формування фахових компетентностей і програмних результатів навчання передбачає необхідність підготовки до лабораторного заняття (вивчення теоретичного матеріалу з теми роботи, попередню підготовку протоколу роботи (наведені у Методичних рекомендаціях до лабораторних занять)), активно працювати на занятті, якісно підготувати звіт, захистити результати виконаної роботи, брати участь у дискусіях щодо прийнятих конструктивних рішень при виконанні здобувачами лабораторних робіт тощо.

Здобувачі вищої освіти мають дотримуватися встановлених термінів виконання всіх видів навчальної роботи відповідно до робочої програми навчальної дисципліни. Термін захисту лабораторної роботи вважається своєчасним, якщо студент захистив її на наступному після виконання роботи занятті. Пропущене лабораторне заняття студент зобов'язаний відпрацювати у встановлений викладачем термін, але не пізніше, ніж за два тижні до кінця теоретичних занять у семестрі.

Засвоєння студентом теоретичного матеріалу з дисципліни оцінюється за результатами опитування під час лабораторних занять, виконання індивідуального завдання та контрольних робіт. Виконання індивідуального завдання завершується його здачею на перевірку у терміни, встановлені графіком самостійної роботи.

Здобувач вищої освіти, виконуючи самостійну роботу або індивідуальну роботу з дисципліни, має дотримуватися політики доброчесності (заборонені списування, плагіат (в т.ч. із використанням мобільних девайсів)). У разі виявлення порушення політики академічної доброчесності в будь-яких видах навчальної роботи здобувач вищої освіти отримує незадовільну оцінку і має повторно виконати завдання з відповідної теми (виду роботи), що передбачені робочою програмою. Будь-які форми порушення академічної доброчесності **не допускаються**.

У межах вивчення навчальної дисципліни допускається визнання та зарахування результатів навчання, здобутих шляхом неформальної освіти на відкритих освітніх платформах. Такі результати можуть враховуватися за умови їх відповідності компетентностям і програмним результатам навчання, визначеним робочою програмою дисципліни, або у разі забезпечення вивчення окремих тем і видів робіт, що передбачені програмою. (детальніше у Положенні про порядок визнання та зарахування результатів навчання здобувачів вищої освіти у ХНУ).

Як результат виконання ЛР №2 зараховується онлайн купс <https://www.coursera.org/learn/agile-atlassian-jira>

Як результат виконання ЛР №3 зараховується онлайн купс <https://www.coursera.org/learn/getting-started-with-git-and-github?specialization=devops-and-software-engineering>

Як результат виконання ЛР №4 зараховується онлайн купс <https://www.coursera.org/learn/user-experience-design>

Як результат виконання ЛР №6 зараховується онлайн купс https://prometheus.org.ua/course/course-v1:LITS+115+2017_T4

9. Оцінювання результатів навчання студентів у семестрі

Оцінювання академічних досягнень здобувача вищої освіти здійснюється відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ». При поточному оцінюванні виконаної здобувачем роботи з кожної структурної одиниці і отриманих ним результатів викладач виставляє йому певну кількість балів із встановлених Робочою програмою для цього виду роботи. При цьому кожна структурна одиниця навчальної роботи може бути зарахована, якщо здобувач набрав не менше 60 відсотків (мінімальний рівень для позитивної оцінки) від максимально можливої суми балів, призначеної структурній одиниці.

При оцінюванні результатів навчання здобувачів вищої освіти з будь-якого виду навчальної роботи (структурної одиниці) рекомендується використовувати наведені нижче узагальнені критерії:

Таблиця – Критерії оцінювання навчальних досягнень здобувача вищої освіти

Оцінка та рівень досягнення здобувачем запланованих ПРН та сформованих компетентностей	Узагальнений зміст критерія оцінювання
--	--

Відмінно (високий)	Здобувач вищої освіти глибоко і у повному обсязі опанував зміст навчального матеріалу, легко в ньому орієнтується і вміло використовує понятійний апарат; уміє пов'язувати теорію з практикою, вирішувати практичні завдання, впевнено висловлювати і обґрунтовувати свої судження. Відмінна оцінка передбачає логічний виклад відповіді мовою викладання (в усній або у письмовій формі), демонструє якісне оформлення роботи і володіння спеціальними приладами та інструментами, прикладними програмами. Здобувач не вагається при видозміні запитання, вміє робити детальні та узагальнюючі висновки, демонструє практичні навички з вирішення фахових завдань. При відповіді допустив дві–три несуттєві <i>похибки</i> .
Добре (середній)	Здобувач вищої освіти виявив повне засвоєння навчального матеріалу, володіє понятійним апаратом, орієнтується у вивченому матеріалі; свідомо використовує теоретичні знання для вирішення практичних задач; виклад відповіді грамотний, але у змісті і формі відповіді можуть мати місце окремі неточності, нечіткі формулювання правил, закономірностей тощо. Відповідь здобувача вищої освіти будується на основі самостійного мислення. Здобувач вищої освіти у відповіді допустив дві–три <i>несуттєві помилки</i> .
Задовільно (достатній)	Здобувач вищої освіти виявив знання основного програмного матеріалу в обсязі, необхідному для подальшого навчання та практичної діяльності за професією, справляється з виконанням практичних завдань, передбачених програмою. Як правило, відповідь здобувача вищої освіти будується на рівні репродуктивного мислення, здобувач вищої освіти має слабкі знання структури навчальної дисципліни, допускає неточності і <i>суттєві помилки</i> у відповіді, вагається при відповіді на видозмінене запитання. Разом з тим, набув навичок, необхідних для виконання нескладних практичних завдань, які відповідають мінімальним критеріям оцінювання і володіє знаннями, що дозволяють йому під керівництвом викладача усунути неточності у відповіді.
Незадовільно (недостатній)	Здобувач вищої освіти виявив розрізнені, безсистемні знання, не вміє виділяти головне і другорядне, допускається помилок у визначенні понять, перекидає їх зміст, хаотично і невпевнено викладає матеріал, не може використовувати знання при вирішенні практичних завдань. Як правило, оцінка «незадовільно» виставляється здобувачеві вищої освіти, який не може продовжити навчання без додаткової роботи з вивчення навчальної дисципліни.

Структурування дисципліни за видами навчальної роботи і оцінювання результатів навчання студентів денної форми здобуття освіти у семестрі

Аудиторна робота								Контрольні заходи		Самостійна робота		Семестровий контроль	Разом
Сьомий семестр													
Лабораторні роботи №:								Контрольні роботи №:		ІЗ №:		Іспит	Сума балів
1	2	3	4	5	6	7	8	1	2	1	2		
Кількість балів за вид навчальної роботи (мінімум-максимум)													
3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	24-40	60-100
24-40								6-10		6-10		24-40	

Примітки: *За набрану з будь-якого виду навчальної роботи з дисципліни кількість балів, нижче встановленого мінімуму, здобувач отримує незадовільну оцінку і має її перездати у встановлений викладачем (деканом) термін. Інституційна оцінка встановлюється відповідно до таблиці «Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС».

Оцінювання результатів захисту лабораторної роботи

Виконана й оформлена відповідно до встановлених Методичними рекомендаціями вимог лабораторна робота комплексно оцінюється викладачем при її захисті з урахуванням таких критеріїв: самостійність та правильність виконання; повнота відповіді та знання.

Результат виконання і захисту здобувачем вищої освіти кожної лабораторної роботи оцінюється відповідно до таблиці Критеріїв оцінювання навчальних досягнень здобувача вищої освіти.

У випадку виявлення здобувачем рівня знань, нижчого ніж 60 відсотків від максимального балу, встановленого Робочою програмою для кожної структурної одиниці, лабораторна робота йому *не зараховується* і для її захисту він має детальніше опрацювати матеріал з теми роботи, методику її виконання, виправити грубі помилки та повторно вийти на її захист у призначений для цього викладачем час.

Оцінювання контрольної роботи

Контрольна робота передбачає виконання п'яти завдань. При оцінюванні контрольної роботи враховуються: повнота відповіді та якість виконання. Кожне завдання оцінюється 1 балом, загальна сума балів на позитивну оцінку становить від 3 до 5.

Розподіл балів при оцінюванні завдань контрольної роботи

Кількість правильних відповідей	1	2	3	4	5
Відсоток правильних відповідей	0-59		60	80	100
Кількість отриманих балів	0		3	4	5

При отриманні негативної оцінки контрольну роботу слід перездати до терміну *наступного* контролю.

Оцінювання результатів виконання індивідуального завдання

Виконане та оформлене відповідно до вимог, визначених Методичними рекомендаціями, індивідуальне завдання (ІЗ) комплексно оцінюється викладачем з урахуванням таких критеріїв: самостійність виконання; правильність розв'язання поставлених задач; обґрунтованість вибору методів розв'язання; повнота пояснень та аргументованість відповідей; якість оформлення та дотримання вимог до структури і змісту роботи.

Результат виконання здобувачем вищої освіти кожного ІЗ оцінюється відповідно до таблиці **Критеріїв оцінювання навчальних досягнень здобувача вищої освіти** з урахуванням рівня досягнення запланованих програмних результатів навчання та сформованих компетентностей. За підсумками захисту присвоюється відповідна сума балів (мінімальний позитивний бал – 3 бали, максимальний – 5 балів).

У разі, якщо здобувач вищої освіти виявив рівень знань і виконання ІЗ, що нижчий ніж 60 відсотків від максимальної кількості балів, встановленої Робочою програмою для цієї структурної одиниці, завдання не зараховується. У такому випадку студент має повторно опрацювати зміст завдання, усунути помилки та здати на перевірку доопрацьоване ІЗ у терміни, погоджені з викладачем.

Оцінювання результатів підсумкового семестрового контролю (іспит)

Освітня програма передбачає підсумковий семестровий контроль з дисципліни у формі іспиту, завданням якого є системне й об'єктивне оцінювання як теоретичної, так і практичної підготовки здобувача з навчальної дисципліни. Складання іспиту відбувається за попередньо розробленими і затвердженими на засіданні кафедри білетами. Відповідно до цього в екзаменаційному білеті пропонується поєднання питань як теоретичного (в т.ч. у тестовій формі), так і практичного характеру.

Таблиця – Оцінювання результатів підсумкового семестрового контролю здобувачів денної форми навчання (40 балів для підсумкового контролю)

Види завдань	Для кожного окремого виду завдань		
	Мінімальний (достатній) бал (задовільно)	Потенційні позитивні бали* (середній бал) (добре)	Максимальний (високий) бал (відмінно)
Завдання № 1	6	8	10
Завдання № 2	6	8	10
Завдання № 3	6	8	10
Завдання № 4	6	8	10
Разом:	24		40

Примітка. *Позитивний бал за іспит, відмінний від мінімального (24 бали) та максимального (40 балів), знаходиться в межах 25-39 балів та розраховується як сума балів за усі структурні елементи (завдання) іспиту.

Для кожного окремого виду завдань підсумкового семестрового контролю застосовуються критерії оцінювання навчальних досягнень здобувача вищої освіти, наведені вище (**Таблиця – Критерії оцінювання навчальних досягнень здобувача вищої освіти**).

Підсумкова семестрова оцінка за інституційною шкалою і шкалою ЄКТС визначається в автоматизованому режимі після внесення викладачем результатів оцінювання у балах з усіх видів навчальної роботи до електронного журналу. Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС наведені нижче у таблиці «Співвідношення».

Семестровий іспит виставляється, якщо загальна сума балів, яку набрав студент з дисципліни за результатами поточного контролю, знаходиться у межах від 60 до 100 балів. При цьому за інституційною шкалою ставиться оцінка «відмінно/добре/задовільно», а за шкалою ЄКТС – буквене позначення оцінки, що відповідає набраній студентом сумі балів відповідно до таблиці Співвідношення.

Таблиця – Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

Оцінка ЄКТС	Рейтингова шкала балів	Інституційна оцінка (рівень досягнення здобувачем вищої освіти запланованих результатів навчання з навчальної дисципліни)	
		Залік	Іспит/диференційований залік
A	90-100	Зараховано	<i>Відмінно/Excellent</i> – високий рівень досягнення запланованих результатів навчання з навчальної дисципліни, що свідчить про безумовну готовність здобувача до подальшого навчання та/або професійної діяльності за фахом
B	83-89		<i>Добре/Good</i> – середній (максимально достатній) рівень досягнення запланованих результатів навчання з навчальної дисципліни та готовності до подальшого навчання та/або професійної діяльності за фахом
C	73-82		
D	66-72		<i>Задовільно/Satisfactory</i> – Наявні мінімально достатні для подальшого

E	60-65		навчання та/або професійної діяльності за фахом результати навчання з навчальної дисципліни
FX	40-59	Незараховано	Незадовільно/Fail – Низка запланованих результатів навчання з навчальної дисципліни відсутня. Рівень набутих результатів навчання є недостатнім для подальшого навчання та/або професійної діяльності за фахом
F	0-39		Незадовільно/Fail – Результати навчання відсутні

10. Питання для самоконтролю результатів навчання

1. Які є підходи до пошуку інформаційних джерел для прийняття обґрунтованих рішень при розв'язанні задач у галузі комп'ютерних наук?
2. Які інформаційні джерела вважаються надійними?
3. Чому важливо використовувати надійні інформаційні джерела для прийняття обґрунтованих рішень при розв'язанні задач у галузі комп'ютерних наук?
4. Для чого залишати посилання у тексті роботи при аналізі інформації з різних джерел для прийняття обґрунтованих рішень при розв'язанні задач у галузі комп'ютерних наук?
5. Що таке постановка завдання до розробки програмного застосунку?
6. Як формулювати постановку завдання до розробки програмного застосунку?
7. З чого повинна складатися постановка завдання до розробки програмного застосунку?
8. Що таке життєвий цикл програмного забезпечення?
9. Які є моделі життєвого циклу програмного забезпечення?
10. Які є етапи розроблення програмного забезпечення?
11. Етап проектування програмного забезпечення.
12. З яких кроків складається визначення вимог до програмного забезпечення?
13. З яких кроків складається проектування архітектури до програмного забезпечення?
14. З яких кроків складається детальне проектування програмного забезпечення?
15. Які є вимоги до спроектованого дизайну програмного забезпечення?
16. Які є рівні проектування дизайну програмного забезпечення?
17. Етап розробки програмного забезпечення.
18. З яких кроків складається реалізація програмного забезпечення?
19. З яких кроків складається інтеграція програмного забезпечення?
20. З яких кроків складається тестування програмного забезпечення?
21. Що таке засіб розробки програмного забезпечення?
22. Які є категорії засобів розробки програмного забезпечення?
23. Що таке UI (користувацький інтерфейс)?
24. Що таке UX (користувацький досвід)?
25. Що таке ескіз інтерфейсу програмного забезпечення?
26. Що таке прототип інтерфейсу програмного забезпечення?
27. Яка мета створення прототипу інтерфейсу програмного забезпечення?
28. Розкрийте суть розроблення алгоритму методом покрокового уточнення.
29. Наведіть кроки при розробленні алгоритму методом покрокового уточнення.
30. Чому важливо забезпечувати високу якість коду при розробці програмного забезпечення?
31. Який код вважається якісним?
32. Які є стандарти кодування, та яке загалом призначення таких стандартів?
33. Розкрийте суть принципу хорошого коду KISS (Keep It Simple, Stupid).
34. Розкрийте суть принципу хорошого коду DRY (Don't Repeat Yourself).
35. Розкрийте суть принципу хорошого коду Open/Closed Principle.
36. Розкрийте суть принципу хорошого коду SRP (Single Responsibility Principle).
37. Розкрийте суть принципу хорошого коду SOLID.
38. Наведіть ознаки хорошого коду.
39. Наведіть ознаки поганого коду.
40. Як покращити якість коду?
41. Розкрийте поняття рефакторингу коду.
42. Яка мета рефакторингу коду?
43. Які є техніки рефакторингу коду?
44. Розкрийте суть техніки рефакторингу "витяг методу".
45. Розкрийте суть техніки рефакторингу "введення змінної".
46. Розкрийте суть техніки рефакторингу "переміщення фрагменту коду".
47. Розкрийте суть техніки рефакторингу "додавання коментарів".
48. Які є інструменти для рефакторингу коду?
49. Дайте визначення якості програмного забезпечення.
50. Хто відповідає за якість програмного забезпечення?
51. Які є метрики якості програмного забезпечення?
52. Які є основні проблеми, з якими стикаються під час забезпечення якості програмного забезпечення?
53. Дайте визначення помилки у програмного забезпеченні.

54. Дайте визначення дефекту (багу) у програмного забезпеченні.
55. Дайте визначення збою у програмного забезпеченні.
56. Що таке тестування програмного забезпечення?
57. Які процеси включає в себе тестування програмного забезпечення?
58. Яка роль тестування в процесі розробки програмного забезпечення?
59. Які є рівні тестування?
60. Які є типи тестування?
61. Які є види тестування?
62. Розкрийте суть функціонального тестування програмного забезпечення.
63. На основі чого може проводитися функціональне тестування програмного забезпечення?
64. Розкрийте суть процесу визначення вимог до використання програмного забезпечення.
65. Які є типи вимог до розгортання програмного забезпечення?
66. В чому полягає суть оцінювання відповідності розробленого програмного забезпечення вимогам?
67. Що включає в себе перевірка відповідності розробленого програмного забезпечення вимогам?
68. Яку роль відіграє самокритичність у оцінювання відповідності розробленого програмного забезпечення вимогам?
69. Що таке засіб розробки програмного забезпечення?
70. Які категорії засобів розробки програмного забезпечення ви знаєте?
71. Які особливості пошуку, оброблення та аналізу інформації з різних джерел для прийняття обґрунтованих рішень щодо вибору засобів розробки програмного забезпечення?
72. Які є критерії вибору зручної парадигми програмування за поставленого завдання?
73. Яка документація створюється під час проектування та розробки програмного забезпечення?
74. Як відбувається документування процесу та результатів проектування програмного забезпечення?

11. Навчально-методичне забезпечення

Освітній процес з дисципліни «Основи програмної інженерії та тестування програмного забезпечення» повністю і в достатній кількості забезпечений необхідною навчально-методичною літературою. Зокрема, викладачами кафедри підготовлені і видані такі роботи:

1. Основи програмної інженерії та тестування програмного забезпечення : теоретичний курс та лабораторний практикум : навчальний посібник / О. В. Бармак, В. І. Кліменко, О. В. Мазурець, М. О. Молчанова, О. В. Собко. – Хмельницький : ХНУ, 2023. – 236 с.
2. Основи програмної інженерії та тестування програмного забезпечення : методичні рекомендації до лабораторних робіт для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки» / В. І. Кліменко, О. В. Бармак, О. В. Мазурець, О. В. Собко, М. О. Молчанова. Хмельницький : ХНУ, 2023. 123 с.

12. Матеріально-технічне та програмне забезпечення дисципліни (за потреби)

Інформаційна та комп'ютерна підтримка: ПК, планшет, смартфон або інший мобільний пристрій, проектор. Програмне забезпечення: програми Microsoft Office або аналогічні, доступ до мережі Інтернет, робота з презентаціями.

Вивчення навчальної дисципліни не потребує використання спеціального програмного прикладного забезпечення, крім загальноновживаних програм і операційних систем, оскільки практичні завдання виконуються із використанням хмарних сервісів та платформ (Jira, GitHub, Figma, Lucidchart, Draw.io, TestLink, TestRail, Katalon Studio, Zephyr), що доступні онлайн.

13. Рекомендована література:

Основна

1. Основи програмної інженерії та тестування програмного забезпечення : теоретичний курс та лабораторний практикум : навчальний посібник / О. В. Бармак, В. І. Кліменко, О. В. Мазурець, М. О. Молчанова, О. В. Собко. – Хмельницький : ХНУ, 2023. – 236 с.
2. Основи програмної інженерії та тестування програмного забезпечення : методичні рекомендації до лабораторних робіт для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки» / В. І. Кліменко, О. В. Бармак, О. В. Мазурець, О. В. Собко, М. О. Молчанова. Хмельницький : ХНУ, 2023. 123 с.
3. Трофименко О. Г., Манаков С. Ю., Ларін Д. Г. Основи програмної інженерії: навч.-метод. посібник. Одеса, Фенікс. 2022. 197 с. URL: http://dspace.onua.edu.ua/bitstream/handle/11300/22773/osnovu_programnoi_injenerii_nmp_2022%20%281%29.pdf?sequence=3&isAllowed=y
4. Петровський С.С. Основи програмної інженерії та тестування програмного забезпечення: лабораторний практикум для студентів напряму підготовки «Комп'ютерні науки». Хмельницький, ХНУ. 2020. 57 с. URL: http://lib.khmnu.edu.ua/EL_LIBRARY/vidavn/metod/mtd2020_1e/38e/index.pdf

Додаткова

5. Скорін Ю.І. Тестування програмного забезпечення: методичні рекомендації до самостійної роботи студентів спеціальності 122 «Комп'ютерні науки». Харків, ХНЕУ ім. С.Кузнеця. 2022. 47 с. URL:

<http://repository.hneu.edu.ua/bitstream/123456789/26832/1/2022%20-%20Скорін%20Ю%20І.pdf>

6. Старух А.І. Методологія тестування програмного забезпечення. Конспект лекцій. Львів, ЛНУ ім. І. Франка. 2020. 43 с. URL: <https://financial.lnu.edu.ua/wp-content/uploads/2019/09/konspekt-testuvannia.pdf>
7. Tsui F., Karam O., Bernal B. Essentials of Software Engineering (4th Edition). Jones & Bartlett Learning. 2023. 332p. ISBN: 978-1284228991. URL: <https://books.google.com.ua/books?id=ln9XEAAAQBAJ>
8. O'Regan G. Concise Guide to Software Testing (Undergraduate Topics in Computer Science). Springer. The MIT Press. Massachusetts Institute of Technology Press. 2019. 293p. ISBN: 9783030284930. URL: <https://mitpressbookstore.mit.edu/book/9783030284930>
9. Scrum. Scrum Community. URL: <https://www.scrum.org/community>
10. Hron M., Obwegeser N. Why and how is Scrum being adapted in practice: A systematic review, Journal of Systems and Software, Volume 183, 2022. URL: <https://doi.org/10.1016/j.jss.2021.111110>
11. Atlassian. Atlassian Community. URL: <https://community.com/>
12. Oggl B., Kofler M. Github. Git: Project Management for Developers and DevOps (1-st Edition). Rheinwerk Computing. 2022. 408p. ISBN: 978-1493222896. URL: <https://www.amazon.com/Git-Project-Management-Developers-DevOps/dp/1493222896>
13. Git Guide. URL: <https://github.com/git-guides>
14. Softwaretestinghelp. Sample Test Case Template With Test Case Examples. URL: <https://www.softwaretestinghelp.com/test-case-template-examples/>
15. Mohan G. Full Stack Testing. A Practical Guide for Delivering High Quality Software. O'Reilly. 2022. 406p. ISBN: 9781098108137. URL: <https://www.amazon.com/Full-Stack-Testing-Practical-Delivering/dp/1098108137>
16. Microsoft. Use the MSTest framework in unit tests. URL: <https://learn.microsoft.com/uk-ua/visualstudio/test/using-microsoft-visualstudio-testtools-unittesting-members-in-unit-tests?view=vs-2022>
17. Creately. The Easy Guide to UML Deployment Diagrams. URL: <https://creately.com/guides/deployment-diagram-tutorial/>
18. Dossantos B. CI/CD: Building a Seamless Software Delivery Pipeline: Learn the basics of A Continuous Integration and Continuous Deployment. Independently published. 2023. 497p. ISBN-13: 979-8397279451. URL: <https://www.amazon.com/dp/B0C6VZ2W8V?tag=uuid10-20>

14. Інформаційні ресурси

1. Модульне середовище для навчання. URL : <https://msn.khmnu.edu.ua/course/view.php?id=4375>
2. Електронна бібліотека університету. URL: <http://library.khmnu.edu.ua/>
3. Репозитарій ХНУ. URL : <https://elar.khmnu.edu.ua/home>

ОСНОВИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Тип дисципліни	Обов'язкова
Рівень вищої освіти	Перший (бакалаврський)
Мова викладання	Українська
Семестр	Другий (четвертий)
Кількість призначених кредитів ЄКТС	5,0
Форми здобуття освіти, для яких викладається дисципліна	Очна (денна)

Результати навчання. Студент, який успішно завершив вивчення дисципліни, повинен: вміти застосовувати методології та засоби проектування інформаційних систем, застосовувати різні види та методології тестування для оцінювання результатів розробки інформаційних систем і можливостей їх використання відповідно до вимог замовника.

Зміст навчальної дисципліни. Основи програмної інженерії. Життєвий цикл розробки програмного забезпечення (SDLC). Технічне завдання. Основи інженерії вимог до програмного забезпечення. Специфікація (SRS) та тестування (RST) програмних вимог. Гнучкі методології створення програмного забезпечення (Scrum, Kanban). Системи керування версіями (GIT). Сервіси для спільної роботи (GitLab, GitHub, GitHub Desktop). Проектування інтерфейсу користувача. UI та UX. Інструменти прототипування (Figma). Основні поняття проектування архітектури ПЗ. Моделювання ПЗ. Мови та засоби моделювання ПЗ. Моделювання бізнес-процесів. Методології та технології моделювання ПЗ (SADT, IDEF, DFD). CASE-технології та CASE-засоби. Оцінювання та забезпечення якості ПЗ. Метрики і стандарти якості ПЗ. Верифікація та валідація ПЗ.

Пререквізити – Алгоритмізація та програмування (ОЗП.05), Теорія алгоритмів (ОЗП.08), Проектування баз даних (ОФП.07), Об'єктно-орієнтоване проектування (ОФП.08).

Кореквізити – Методи та системи штучного інтелекту (ОФП.01), Методи та системи штучного інтелекту (курсний проєкт) (ОФП.02), Вебтехнології (ОФП.03), Вебтехнології (курсний проєкт) (ОФП.04), Технології розподілених систем та паралельних обчислень (ОФП.11), Системне та мережеве програмне забезпечення (ОФП.14), Управління ІТ-проєктами (ОФП.15), Проєктно-технологічна практика (ОФП.16), Професійна практика (ОФП.17), Кваліфікаційна робота (ОФП.18).

Запланована навчальна діяльність: Мінімальний обсяг навчальних занять в одному кредиті ЄКТС навчальної дисципліни для *першого* (бакалаврського) рівня вищої освіти за денною формою здобуття освіти становить 10 годин на 1 кредит ЄКТС.

Форми (методи) навчання: Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій та методів навчання, зокрема: лекції (з використанням методів візуалізації, проблемного й інтерактивного навчання, мотиваційних прийомів, інформаційно-комунікаційних технологій); лабораторні заняття (з використанням інструктування, демонстрування, розв'язування типових і прикладних задач, аналізу кейсів, ситуаційних завдань, елементів дискусії тощо); самостійна робота (опрацювання теоретичного матеріалу, підготовка до виконання лабораторних робіт, поточного та підсумкового контролю, виконання індивідуальних завдань), з використанням інформаційно-комп'ютерних технологій та технологій дистанційного навчання.

Форми оцінювання результатів навчання: оцінювання лабораторних та контрольних робіт; виконання індивідуального завдання; підсумковий контрольний захід.

Вид семестрового контролю: іспит – 1 семестр.

1. Основи програмної інженерії та тестування програмного забезпечення : теоретичний курс та лабораторний практикум : навчальний посібник / О. В. Бармак, В. І. Кліменко, О. В. Мазурець, М. О. Молчанова, О. В. Собко. – Хмельницький : ХНУ, 2023. – 236 с.

2. Основи програмної інженерії та тестування програмного забезпечення : методичні рекомендації до лабораторних робіт для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки» / В. І. Кліменко, О. В. Бармак, О. В. Мазурець, О. В. Собко, М. О. Молчанова. Хмельницький : ХНУ, 2023. 123 с.

3. Трофименко О. Г., Манаков С. Ю., Ларін Д. Г. Основи програмної інженерії: навч.-метод. посібник. Одеса, Фенікс. 2022. 197 с. URL: http://dspace.onua.edu.ua/bitstream/handle/11300/22773/osnovu_programnoi_injenerii_nmp_2022%20%281%29.pdf?sequence=3&isAllowed=y

4. Петровський С.С. Основи програмної інженерії та тестування програмного забезпечення: лабораторний практикум для студентів напряму підготовки «Комп'ютерні науки». Хмельницький, ХНУ. 2020. 57 с. URL:

5. Скорін Ю.І. Тестування програмного забезпечення: методичні рекомендації до самостійної роботи студентів спеціальності 122 «Комп'ютерні науки». Харків, ХНЕУ ім. С.Кузнеця. 2022. 47 с. URL: <http://repository.hneu.edu.ua/bitstream/123456789/26832/1/2022%20-%20Скорін%20Ю%20І.pdf>
6. Старух А.І. Методологія тестування програмного забезпечення. Конспект лекцій. Львів, ЛНУ ім. І. Франка. 2020. 43 с. URL: <https://financial.lnu.edu.ua/wp-content/uploads/2019/09/konspekt-testuvannia.pdf>
7. Tsui F., Karam O., Bernal B. Essentials of Software Engineering (4th Edition). Jones & Bartlett Learning. 2023. 332p. ISBN: 978-1284228991. URL: <https://books.google.com.ua/books?id=IN9XEAAAQBAJ>
8. O'Regan G. Concise Guide to Software Testing (Undergraduate Topics in Computer Science). Springer. The MIT Press. Massachusetts Institute of Technology Press. 2019. 293p. ISBN: 9783030284930. URL: <https://mitpressbookstore.mit.edu/book/9783030284930>
9. Scrum. Scrum Community. URL: <https://www.scrum.org/community>
10. Hron M., Obwegeser N. Why and how is Scrum being adapted in practice: A systematic review, Journal of Systems and Software, Volume 183, 2022. URL: <https://doi.org/10.1016/j.jss.2021.111110>.
11. Atlassian. Atlassian Community. URL: <https://community.com/>
12. Oggl B., Kofler M. Github. Git: Project Management for Developers and DevOps (1-st Edition). Rheinwerk Computing. 2022. 408p. ISBN: 978-1493222896. URL: <https://www.amazon.com/Git-Project-Management-Developers-DevOps/dp/1493222899>
13. Git Guide. URL: <https://github.com/git-guides>
14. Softwaretestinghelp. Sample Test Case Template With Test Case Examples. URL: <https://www.softwaretestinghelp.com/test-case-template-examples/>
15. Mohan G. Full Stack Testing. A Practical Guide for Delivering High Quality Software. O'Reilly. 2022. 406p. ISBN: 9781098108137. URL: <https://www.amazon.com/Full-Stack-Testing-Practical-Delivering/dp/1098108132>
16. Microsoft. Use the MSTest framework in unit tests. URL: <https://learn.microsoft.com/uk-ua/visualstudio/test/using-microsoft-visualstudio-testtools-unittesting-members-in-unit-tests?view=vs-2022>
17. Creately. The Easy Guide to UML Deployment Diagrams. URL: <https://creately.com/guides/deployment-diagram-tutorial/>
18. Dossantos B. CI/CD: Building a Seamless Software Delivery Pipeline: Learn the basics of A Continuous Integration and Continuous Deployment. Independently published. 2023. 497p. ISBN-13: 979-8397279451. URL: <https://www.amazon.com/dp/B0C6VZ2W8V?tag=uuid10-20>
19. Модульне середовище для навчання. URL : <https://msn.khmnu.edu.ua/course/view.php?id=4375>
20. Електронна бібліотека університету. URL: <http://library.khmnu.edu.ua/>
21. Репозитарій ХНУ. URL : <https://elar.khmnu.edu.ua/home>

Викладачі: асистент, Кліменко В.І.
Ph.D, ст.викл., Собко О.В.
к.т.н., доцент, Мазурець О.В..