

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ



ЗАТВЕРДЖУЮ

Декан факультету інформаційних технологій

Тетяна ГОВОРУЩЕНКО

09 2025 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
Функціональне тестування програмного забезпечення

Призначення робочої програми
Рівень вищої освіти
Мова навчання
Обсяг дисципліни, кредитів ЕКТС
Статус дисципліни
Факультет
Кафедра

Для освітніх програм різних спеціальностей
Перший (бакалаврський)
Українська
8
Вибіркова
Інформаційних технологій
Інженерії програмного забезпечення

Форма здобуття освіти	Загальне навантаження		Кількість годин				Курсовий проект	Курсова робота	Залік	Іспит
			Аудиторні заняття			Самостійна робота				
	Європ. кредит	Години	Всього	Лекцій	Лабораторні роботи					
Д	8	240	82	32	50	158			+	

Робоча програма складена на основі освітніх програм підготовки бакалавра та стандарту вищої освіти спеціальності.

Робоча програма складена _____ канд. пед наук, доцент Онишко О.Г.

Схвалено на засіданні кафедри інженерії програмного забезпечення

Протокол від 30.08 2025 №1

Зав. кафедри інженерії програмного забезпечення _____ Леонід БЕДРАТЮК

Хмельницький, 2025 р.

ФУНКЦІОНАЛЬНЕ ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Тип дисципліни	Вибіркова
Рівень вищої освіти	Перший
Мова викладання	Українська
Семестр	-
Обсяг кредитів ЄКТС	8,0
Форма здобуття освіти	Денна/очна

Результати навчання. Після вивчення дисципліни студент повинен: розуміти основи функціонального тестування: знати цілі, принципи та етапи функціонального тестування, розуміти роль функціонального тестування в забезпеченні якості ПЗ, вміти розробляти тестову документацію, складати тест-плани, тест-кейси, чек-листи, формувати звіти про результати тестування, застосовувати техніки тест-дизайну, а саме еквівалентне розбиття, аналіз граничних значень, таблиці прийняття рішень тощо. Виконувати ручне функціональне тестування, проводити тестування веб і десктопних застосунків згідно із специфікацією, використовувати. Оцінювати результати тестування та аналізувати виявлені дефекти, визначати ступінь готовності ПЗ до релізу

Зміст навчальної дисципліни. Основні поняття тестування Види тестування Функціональне тестування: визначення та цілі Життєвий цикл розробки програмного забезпечення Методи функціонального тестування Тестування вимо Типи тестів Тестування користувацького інтерфейсу Методи проектування тестів

Тестові випадки та тестові сценарії Автоматизація функціонального тестування. Інструменти для функціонального тестування Виконання тестів та ведення звітності Управління дефектами Тестування на сумісність Тестування безпеки Тестування продуктивності Тестування в Agile та DevOps Кейс-стаді та практичні заняття

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.

Методи навчання: методи проблемного викладання, словесні, наочні (лекції); пояснювально-ілюстративні, проблемного викладання, дослідницькі, частково-пошукові (лабораторні заняття), проблемного викладання, дослідницькі, частково-пошукові (самостійна робота: індивідуальні завдання).

Форми і методи оцінювання результатів навчання: усне опитування, захист лабораторних робіт, тестування

Вид семестрового контролю: залік

Навчальні ресурси:

1. Ian Sommerville. Software Engineering (10th Edition). Pearson Education, 2021. – 816 p.
2. Adzic, G. Specification by Example: How Successful Teams Deliver the Right Software (2nd Edition). Manning Publications, 2021. – 320 p.
3. Laplante, P., Kassab, M. Requirements Engineering for Software and Systems (4th Edition). CRC Press, 2022. – 420 p.
4. Pohl, K., Rupp, C. Requirements Engineering: Fundamentals, Principles, and Techniques. Springer, 2022. – 400 p.
5. Tavana, M., Pick, K. W. Managing Requirements Knowledge. Springer, 2022. – 300 p.
6. Модульне середовище для навчання. Доступ до ресурсу: <https://msn.khmn.edu.ua/>

Викладач: кандидат педагогічних наук, доцент Онишко О.Г.

3. ПОЯСНЮВАЛЬНА ЗАПИСКА

Дисципліна “Функціональне тестування програмного забезпечення” є дисципліною професійної та практичної підготовки.

Мета дисципліни – полягає в формуванні у студентів теоретичних знань і практичних навичок, необхідних для ефективного планування, розробки та виконання функціонального тестування програмних продуктів. Дисципліна спрямована на розвиток умінь виявляти, документувати та класифікувати дефекти, аналізувати результати тестування та оцінювати якість програмного забезпечення.

Предмет дисципліни. Предмет дисципліни «Функціональне тестування програмного забезпечення» охоплює теоретичні знання, методи, засоби та техніки функціонального тестування, які застосовуються для перевірки правильності роботи програмного забезпечення відповідно до вимог користувача та специфікацій. Дисципліна охоплює процеси планування тестування, розробки тест-кейсів, аналізу результатів та застосування інструментів для автоматизації тестування..

Завдання дисципліни: Основними завданнями вивчення дисципліни «Функціональне тестування програмного забезпечення» є Ознайомлення студентів з основами тестування, його цілями та основними техніками, такими як тестування чорної скриньки, аналіз граничних значень та тестування еквівалентних класів. Формування умінь аналізувати вимоги, створювати тест-кейси та сценарії, що покривають функціональність ПЗ, і реєструвати результати тестування. Знайомство з основними інструментами автоматизації, вивчення технік автоматизованого функціонального тестування та написання базових скриптів для автоматизації. Формування вмінь аналізувати результати тестування та документувати дефекти. Ознайомлення з кращими практиками та стандартами тестування ПЗ

Результати навчання. Після вивчення дисципліни студент повинен розуміти основи функціонального тестування: знати цілі, принципи та етапи функціонального тестування, розуміти роль функціонального тестування в забезпеченні якості ПЗ, вміти розробляти тестову документацію, складати тест-плани, тест-кейси, чек-листи, формувати звіти про результати тестування, застосовувати техніки тест-дизайну, а саме еквівалентне розбиття, аналіз граничних значень, таблиці прийняття рішень тощо. Виконувати ручне функціональне тестування, проводити тестування веб і десктопних застосунків згідно із специфікацією, використовувати. Оцінювати результати тестування та аналізувати виявлені дефекти, визначати ступінь готовності ПЗ до релізу.

4. Структура залікових кредитів дисципліни

Теми	Лекції	Лаб. роб.	СРС
Тема 1. Основи тестування програмного забезпечення	6	8	26
Тема 2. Методи та техніки функціонального тестування	6	8	26
Тема 3. Тестові сценарії та документація.	6	8	26
Тема 4. Автоматизація тестування та інструменти	6	8	26
Тема 5. Спеціалізоване тестування	6	8	26
Тема 6. Тестування в сучасних підходах розробки	2	10	28
Разом за семестр	32	50	158

5. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

5.1. Зміст лекційного курсу*

№ лекції	Перелік тем лекцій, їх анотація	Кількість годин
Тема 1. Основи тестування програмного забезпечення		
1	Лекція 1. Введення в тестування програмного забезпечення Основні поняття тестування: визначення тестування, його цілі та роль у розробці. Види тестування: функціональне, нефункціональне, регресійне та інші. Життєвий цикл тестування: етапи, такі як планування, виконання, аналіз результатів. Основні терміни: дефект, помилка, тестовий випадок, тестовий сценарій.. Література: [1-3; 7].	2
2	Лекція2. Функціональне тестування: визначення та цілі Що таке функціональне тестування: мета та значення. Вимоги як основа тестування: як формуються вимоги, їх вплив на тестування. Різниця між функціональним і нефункціональним тестуванням: приклади. Переваги функціонального тестування: чому воно важливе для якості ПЗ. Література: [2; 5].	2
3	Лекція 3. . Життєвий цикл розробки програмного забезпечення Моделі життєвого циклу: каскадна модель, спіральна модель, Взаємозв'язок між розробкою та тестуванням: інтеграція на різних етапах. Вплив вимог на тестування: як вимоги формують тестову документацію. Критичні точки життєвого циклу: де особливо важливе тестування.. Література: [3-4; 8].	2
Тема 2. Методи та техніки функціонального тестування		
4	Лекція 4. Методи функціонального тестування Чорна скринька: принципи, переваги та недоліки. Тестування на основі вимог: створення тестових випадків на основі специфікацій. Методи	2

	сценарного тестування: як розробляти сценарії. Аналіз ризиків: виявлення та тестування критичних функцій.. Література: [4; 7].	
5	Лекція 5 Тестування вимог Аналіз вимог: методи вивчення та документування. Типи вимог: функціональні та нефункціональні вимоги. Створення тестових випадків на основі вимог: кроки та приклади. Перевірка відповідності вимогам: методи верифікації та валідації.. Література: [2; 4-6; 8].	2
6	Лекція 6 Типи тестів Модульне тестування: що це таке і як його проводити. Інтеграційне тестування: мета, типи, методи. Системне тестування: особливості, підходи до тестування. Регресійне тестування: коли і як його виконувати. Тестування користувацького інтерфейсу Література: [1-3; 7]	2
Тема 3. Тестові сценарії та документація		
7	Лекція 8. Методи проектування тестів Еквівалентне класування: принципи, приклади. Граничні значення: теорія, як застосовувати. Дерево рішень: методи та приклади. Комбіновані методи проектування тестів: використання декількох підходів. Література: [1; 4].	2
8	Лекція 9. Тестові випадки та тестові сценарії Структура тестового випадку: основні елементи, документація. Процес створення тестових сценаріїв: кроки, на що звертати увагу. Аналіз тестових випадків: валідація, ефективність. Ведення документації: стандарти, формати звітів. Література: [3-4; 6].	2
9	Лекція 10. Автоматизація функціонального тестування Критерії для автоматизації тестування: коли автоматизувати. Переваги автоматизації: економія часу, підвищення якості. Інструменти автоматизації: огляд популярних рішень (Selenium, JUnit). Розробка та підтримка автоматизованих тестів: основи та практичні поради. Література: [3; 5; 8].	2
Тема 4. Автоматизація тестування та інструменти		
10	Лекція 11. Інструменти для функціонального тестування Огляд інструментів тестування: Selenium, TestNG, JUnit та інші. Вибір інструментів: фактори, які варто враховувати. Налаштування середовища для тестування: кроки для успішної роботи. Інтеграція інструментів у CI/CD: принципи та приклади. Література: [1-2; 7].	2
11	Лекція 12. Виконання тестів та ведення звітності Планування тестування: етапи та важливі моменти. Виконання тестових сценаріїв: процес, контроль якості. Звітність про результати тестування: формати, зміст звітів. Аналіз і використання результатів: покращення процесів на основі отриманих даних. Література: [5-6].	2

12	Лекція 13. Управління дефектами Життєвий цикл дефекту: етапи, документування. Класифікація дефектів: серйозність, пріоритет, категорії. Інструменти для управління дефектами: Jira, Bugzilla, інші. Комунікація з командою: методи обміну інформацією про дефекти. Література: [8].	2
Тема 5. . Спеціалізоване тестування		
13	Лекція 14. Тестування на сумісність Основи тестування на сумісність: значення та мета. Параметри для тестування: операційні системи, браузері, пристрої. Методи тестування: ручне та автоматизоване тестування. Документування результатів: як оформлювати звіти про сумісність.. Література: [7-8].	2
14	Лекція 15. Тестування безпеки Основи тестування безпеки: мета та методи. Типи вразливостей: SQL-ін'єкції, XSS, CSRF та інші. Методи тестування безпеки: статичне та динамічне тестування. Інструменти для тестування безпеки: огляд популярних рішень. Література: [2; 4].	2
15	Лекція 16. Тестування продуктивності Вимірювання продуктивності: метрики, індикатори. Методи навантажувального тестування: техніки та підходи. Аналіз результатів продуктивності: як інтерпретувати дані. Оптимізація продуктивності: рекомендації для підвищення. Література: [1-4, 6].	2
Тема 6. Тестування в сучасних підходах розробки		
16	Лекція17. Тестування в Agile та DevOps Основи Agile тестування: принципи, методи, інструменти. Тестування в DevOps: інтеграція тестування в CI/CD. Методики тестування в Agile: TDD, BDD, ATDD. Роль тестувальника в Agile команді: завдання та виклики. Література: [5, 7].	2
Разом за 7-й семестр		32

5.2 Зміст лабораторних занять

№ з/п	Теми лабораторних занять	Години
1	Введення в тестування	6
2	Аналіз вимог..	6
3	Проектування тестів	6
4	Виконання тестових випадків	6
5	Автоматизація тестування	6
6	Тестування користувацького інтерфейсу	6
7	Управління дефектами.	6
8	Тестування на сумісність.	4
9	Аналіз результатів тестування	4

Разом	50
--------------	-----------

5.3 Зміст самостійної роботи

Об'єм самостійної роботи з дисципліни “Функціональне тестування програмного забезпечення” становить 158 години. Вони включають опрацювання лекційного матеріалу, теоретичних і лабораторних завдань, підготовку до виконання лабораторних робіт, їх захисту і поточне тестування.

Номер тижня	Вид самостійної роботи	Кількість годин
1-2	Опрацювання лекційного матеріалу. Підготовка до ЛР1 та ПЗ1. Підготовка до захисту ЛР1.	16
3-4	Опрацювання лекційного матеріалу. Підготовка до ЛР2 та ПЗ2. Підготовка до захисту ЛР2.	16
5-6	Опрацювання лекційного матеріалу. Підготовка до ЛР3 та ПЗ3. Підготовка до захисту ЛР3	16
7-8	Опрацювання лекційного матеріалу. Підготовка до ЛР4 та ПЗ4. Підготовка до захисту ЛР4.	16
9-10	Опрацювання лекційного матеріалу. Підготовка до ЛР5 та ПЗ5. Підготовка до захисту ЛР5	16
11-12	Опрацювання лекційного матеріалу. Підготовка до ЛР6 та ПЗ6. Підготовка до захисту ЛР6.	16
13-14	Опрацювання лекційного матеріалу. Підготовка до ЛР7 та ПЗ7. Підготовка до захисту ЛР7.	16
15-16	Опрацювання лекційного матеріалу. Підготовка до ЛР8 та ПЗ8. Підготовка до захисту ЛР8	16
17	Опрацювання лекційного матеріалу. Підготовка до ПЗ9 та тестування.	14
	Разом 1-й семестр:	158

Примітки: ЛР – лабораторна робота, ПЗ– практичне заняття.

6. Технології та методи навчання

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних методів: методи проблемного викладання, словесні, наочні (лекції); пояснювально-ілюстративні, проблемного викладання, дослідницькі, частково-пошукові (лабораторні

заняття), проблемного викладання, дослідницькі, частково-пошукові (самостійна робота: індивідуальні завдання). Всі заняття проводяться з використанням інформаційних технологій і мають за мету – набуття здобувачами першого (бакалаврського) рівня вищої освіти практичних навичок з конструювання програмного забезпечення, тестування та налагодження відладки, використовувати метрики для оцінки розробленого ПЗ.

1. 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:

- тестовий контроль засвоєння теоретичного та практичного матеріалу з теми;
- оцінювання результатів захисту лабораторних робіт;
- тестовий контроль теоретичного матеріалу;

Підсумкова семестрова оцінка виставляється за результатами поточного контролю. Здобувач вищої освіти, який набрав з будь-якого виду навчальної роботи, суму балів нижчу за 60 відсотків від максимального балу, 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 практичних та лабораторних занять (вивчення теоретичного матеріалу з теми, попередню підготовку протоколу роботи, підготовку до усного опитування для допуску до заняття (наведені у Методичних рекомендаціях до лабораторних занять)), активно працювати на занятті, брати участь у дискусіях щодо прийнятих 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 тестування.

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 здобувачам вищої освіти передбачено визнання і зарахування результатів навчання, набутих шляхом неформальної освіти, що розміщені на доступних платформах, які сприяють формування компетентностей і поглибленню результатів навчання, визначених робочою програмою дисципліни, або забезпечують вивчення відповідної теми та/або виду робіт з програми навчальної дисципліни (детальніше у Положенні про порядок визнання та зарахування результатів навчання здобувачів вищої освіти у ХНУ).

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, перекручує 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									Семестровий контроль
Лабораторні заняття №:								Тестовий контроль	Залік
1*	2	3	4	5	6	7	8	Т 1-4	
Кількість балів за кожний вид навчальної роботи (мінімум-максимум)									
6-10	6-10	6-10	6-10	6-10	6-10	6-10	6-10	6-10	За рейтингом
48-80								12-20	60-100**

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 результатів тестового контролю

Кожний з тестів, передбачених Робочою програмою, складається із 20 тестових завдань, кожне з яких є рівнозначним.

Відповідно до таблиці структурування видів робіт за тестовий контроль здобувач залежно від кількості правильних відповідей може отримати від 12 до 20 балів.

Розподіл балів в залежності від наданих правильних відповідей на тестові завдання

Кількість правильних відповідей	0-11	12-13	14	15-16	17	18-20
Відсоток правильних відповідей	0-59	60-65	66-72	73-82	83-89	90-100
Кількість отриманих балів	0	12	14	16	18	20

На тестування відводиться 30-40 хвилин. Студент проходить тестування в он-лайн режимі у Модульному середовищі для навчання. Також, студент може проходити тестування письмово, записуючи правильні відповіді у талоні відповідей. При отриманні негативної оцінки тест слід перездати до терміну *наступного* контролю.

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».

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		Добре/Good – середній (максимально достатній) рівень досягнення запланованих результатів навчання з навчальної дисципліни та готовності до подальшого навчання та/або професійної діяльності за фахом Задовільно/Satisfactory – Наявні мінімально достатні для подальшого навчання та/або професійної діяльності за фахом результати навчання з навчальної дисципліни
C	73-82		
D	66-72		
E	60-65		
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

Семестровий залік виставляється на останньому занятті за умови якщо загальна сума балів, яку накопичив здобувач з дисципліни (іншого освітнього компонента) за результатами **поточного** контролю, знаходиться у межах від 60 до 100 балів. При цьому за інституційною шкалою ставиться оцінка «**зараховано**», а за шкалою ЄКТС – буквене позначення оцінки, що відповідає набраній студентом сумі балів відповідно до таблиці Співвідношення. Присутність здобувача у цьому випадку не є обов’язковою.

8. Питання для самоперевірки та заліку з дисципліни “Функціональне тестування програмного забезпечення”

1. Що таке тестування програмного забезпечення, і яка його основна мета?
2. Які є види тестування та їх призначення?
3. З яких етапів складається життєвий цикл тестування?
4. Яка різниця між помилкою, дефектом та збоєм у тестуванні?
5. Що таке тестовий випадок, і яка його роль у тестуванні?
6. Які є основні ролі тестувальника у процесі розробки програмного забезпечення?
7. Що таке функціональне тестування, і яка його основна мета?
8. Яким чином вимоги впливають на функціональне тестування?
9. Чим функціональне тестування відрізняється від нефункціонального?
10. У чому полягають переваги функціонального тестування для якості програмного забезпечення?
11. Які моделі життєвого циклу розробки ПЗ ви знаєте? У чому особливості кожної?
12. Як інтегрується тестування на різних етапах життєвого циклу розробки?
13. Як вимоги впливають на тестову документацію?
14. У яких критичних точках життєвого циклу тестування є особливо важливим?
15. Що таке тестування методом "чорної скриньки"?
16. Як створюються тестові випадки на основі вимог?
17. У чому суть тестування за сценаріями, і як воно виконується?
18. Як аналіз ризиків допомагає визначити критичні функції для тестування?
19. Які методи використовуються для аналізу вимог?

20. У чому різниця між функціональними та нефункціональними вимогами?
21. Які кроки необхідні для створення тестових випадків на основі вимог?
22. Що таке верифікація та валідація у контексті тестування вимог?
23. Що таке модульне тестування, і коли його використовують?
24. Які типи інтеграційного тестування існують?
25. У чому полягає суть системного тестування?
26. Що таке регресійне тестування, і коли його доцільно проводити?
27. Чому тестування інтерфейсу користувача є важливим для якості ПЗ?
28. Які основні принципи забезпечення зручності використання?
29. Як перевіряють відповідність дизайну інтерфейсу вимогам?
30. Які інструменти використовують для автоматизації тестування UI?
31. Що таке еквівалентне класування, і як його застосовують?
32. Як використовуються граничні значення у тестуванні?
33. Як метод "дерево рішень" допомагає в тестуванні?
34. Коли слід використовувати комбіновані методи проектування тестів?
35. Які основні елементи містить структура тестового випадку?
36. Які кроки слід виконати для створення тестових сценаріїв?
37. Як виконується валідація та аналіз тестових випадків?
38. Які стандарти ведення документації для тестування існують?
39. Які критерії використовуються для автоматизації тестування?
40. Назвіть популярні інструменти автоматизації та основні їхні переваги.

9. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна:

1. Ian Sommerville. Software Engineering (10th Edition). Pearson Education, 2021. – 816 p.
2. Adzic, G. Specification by Example: How Successful Teams Deliver the Right Software (2nd Edition). Manning Publications, 2021. – 320 p.
3. Laplante, P., Kassab, M. Requirements Engineering for Software and Systems (4th Edition). CRC Press, 2022. – 420 p.
4. Pohl, K., Rupp, C. Requirements Engineering: Fundamentals, Principles, and Techniques. Springer, 2022. – 400 p.
5. Tavana, M., Pick, K. W. Managing Requirements Knowledge. Springer, 2022. – 300 p.
6. La Rocca, M. Advanced Algorithms and Data Structures. Manning Publications, 2021. – 978 p.
7. Piotr Sliż. Organizacja procesowo-projektowa. Istota, modelowanie, pomiar dojrzałości. Difin, 2021. – 292 p.
8. D'Andrade, Brian. Software Engineering: Artificial Intelligence, Compliance, and Security. Nova Science Publishers, 2021. – 512 p.

Допоміжна:

1. ISO/IEC/IEEE 25010:2023 – Systems and Software Quality Requirements and Evaluation (SQuaRE): System and Software Quality Models. IEEE Standards Association.
2. ISO/IEC/IEEE 29148:2022 – Systems and Software Engineering — Life Cycle Processes — Requirements Engineering. IEEE Standards Association.
3. Bedratiuk, L. P., Bedratiuk, H. I., Hurman, I. V. Method for Increasing Individual Image

Using Multi-Output Regression and Moment Metric. Measurement and Computing Technology in Technological Processes, № 1, 2023, pp. 149–157.

4. Petrov, A. Software Quality Assurance: Modern Practices and Tools. CRC Press, 2023. – 280 p.

5. Marijan, D., Gotlieb, A. Software Testing: Automation, Practices, and Tools. Springer, 2022. – 350 p.

6. Kaur, A., and Sharma, R. Fundamentals of Software Testing. Wiley, 2022. – 312 p.

7. Garousi, V., Felderer, M. Software Test Design: Case Studies and Methodologies. Springer, 2023. – 400 p.

10. ІНФОРМАЦІЙНІ РЕСУРСИ

1. Модульне середовище для навчання. Доступ до ресурсу: <https://msn.khnu.km.ua>.
2. Електронна бібліотека університету . Доступ до ресурсу: http://lib.khnu.km.ua/asp/php_f/plage_lib.php.
3. Репозитарій ХНУ. Доступ до ресурсу: <http://elar.khnu.km.ua/jspui/?locale=uk>.