

Міністерство освіти і науки України
Сумський національний аграрний університет
Факультет економіки і менеджменту
Кафедра кібернетики та інформатики

Робоча програма (силабус) освітнього компонента

Тестування та забезпечення якості ПЗ (QA)

Спеціальність	Е6 Інформаційні системи і технології
Освітня програма	Інформаційні системи та технології
Рівень вищої освіти	1 (бакалаврський)

Розробник: Лариса Лариса СИТНИК, кандидат технічних наук, доцент
кафедри кібернетики та інформатики

Розглянуто, схвалено та затверджено на засіданні кафедри кібернетики та інформатики	протокол від 04.06.2025 р. №19
	Завідувач <u>Агаджанова</u> Світлана АГАДЖАНОВА кафедри (підпис)

Погоджено:

Гарант освітньої програми Агаджанова Світлана АГАДЖАНОВА

Декан факультету,
де реалізується освітня програма Лукаш Світлана ЛУКАШ

Рецензія на робочу програму (додається) надана: О.В.Васильченко

Методист відділу якості освіти,
ліцензування та акредитації Наріє Нараніє (Наріє Нараніє)

Зареєстровано в електронній базі: дата: 26.06. 2025 р.

Інформація про перегляд робочої програми (силабусу):

Навчальний рік, в якому вносяться зміни	Номер додатку до робочої програми з описом змін	Зміни розглянуто і схвалено		
		Дата та номер протоколу засідання кафедри	Завідувач кафедри	Гарант освітньої програми

1. ЗАГАЛЬНА ІНФОРМАЦІЯ ПРО ОСВІТНІЙ КОМПОНЕНТ

1.	Назва ОК	Тестування програмного забезпечення(QA engineering)			
2.	Факультет/кафедра	Факультет економіки та менеджменту / Кафедра кібернетики та інформатики			
3.	Статус ОК	Обов'язковий			
4.	Програма/Спеціальність (програми), складовою яких є ОК (заповнюється для обов'язкових ОК)	ОП - Інформаційні системи та технології/ Спеціальність – F6 «Інформаційні системи і технології»			
5.	ОК може бути запропонований для (заповнюється для вибіркових ОК)				
6.	Рівень НРК	6 рівень			
7.	Семестр та тривалість вивчення	3 курс, 6 семестр			
8.	Кількість кредитів ЄКТС	5 кредитів (150 годин)			
9.	Загальний обсяг годин та їх розподіл	Контактна робота(заняття)			Самостійна робота
		Лекційні	Практичні /семінарські	Лабораторні	
	6 семестр	30	30	–	90
10.	Мова навчання	українська			
11.	Викладач / Координатор освітнього компонента	доцент кафедри кібернетики та інформатики, кандидат технічних наук Ситник Лариса Григорівна			
11.1	Контактна інформація	larysasytnyk@soippo.edu ; ауд. 308е.			
12.	Загальний опис освітнього компонента	Дисципліна "Тестування програмного забезпечення" охоплює теоретичні та практичні аспекти процесу тестування програмних продуктів і сервісів на різних етапах їх розробки, з метою виявлення та усунення дефектів, а також забезпечення високої якості програмного забезпечення (ПЗ). У рамках курсу розглядаються ключові принципи тестування, методи та підходи, що використовуються для перевірки функціональності, надійності, продуктивності та безпеки ПЗ, вивчаються різні види тестування, включно з функціональним, регресійним, інтеграційним, навантажувальним, а також інструменти для автоматизації процесу тестування. Окрема увага приділяється управлінню дефектами, їх моніторингу та усуненню, а також використанню сучасних систем керування дефектами (Jira, Bugzilla). Дисципліна також знайомить з поняттям забезпечення якості (QA), його методами та стандартами, що допомагають гарантувати відповідність програмного продукту вимогам користувача та індустрії, що дозволяє опанувати навички планування тестування, розробки тест-кейсів, оцінки ефективності тестування через метрики якості та використання інструментів автоматизації для підвищення швидкості та надійності процесу тестування. Таким чином, ця дисципліна сприяє формуванню професійних навичок, необхідних для забезпечення якості програмних продуктів у реальних ІТ-проектах.			
13.	Мета освітнього компонента	Метою освітнього компонента є формування та розвиток у здобувачів вищої освіти загальних і професійних навичок з тестування та забезпечення якості програмних засобів інформаційних систем і технологій з урахуванням експлуатаційних умов, властивостей, призначення і технічних характеристик протягом їх життєвого циклу.			

14.	Передумови вивчення ОК, зв'язок з іншими освітніми компонентами ОП	1. Освітній компонент базується на вивченні таких дисциплін як: ОК21 Об'єктно-орієнтоване програмування 2. Освітній компонент є основою для: ОК33 «Кваліфікаційна робота»
15.	Політика академічної доброчесності	При виконання практичних робіт, написанні рефератів та при написання проміжних контрольних, атестаційних, залікових та екзаменаційних робіт здобувач вищої освіти обов'язково має дотримуватись правил академічної доброчесності. При виявленні фактів списування або академічної не доброчесності робота виконана здобувачем вищої освіти робота анулюється.
16.	Ключові слова	Програмне забезпечення, тестування ПЗ, тест-план
17.	Посилання на курс у Moodle	https://cdn.snau.edu.ua/moodle/course/view.php?id=4640

2. РЕЗУЛЬТАТИ НАВЧАННЯ ЗА ОСВІТНІМ КОМПОНЕНТОМ ТА ЇХ ЗВ'ЯЗОК З ПРОГРАМНИМИ РЕЗУЛЬТАТАМИ НАВЧАННЯ

Результати навчання за ОК: Після вивчення освітнього компонента здобувач вищої освіти очікувано буде здатен...»	Програмні результати навчання, на досягнення яких спрямований ОК (зазначити номер згідно з нумерацією, наведеною в ОП)		Як оцінюється ДРН
	РН09	РН14	
ДРН 1 Знати види, методи та підходи до тестування для виявлення можливих помилок і дефектів на різних етапах розробки програмного забезпечення (ПЗ)	×	×	Звіт за результатами виконання практичних робіт №1 та №2 Проміжний контроль №1: тести множинного вибору Підсумковий контроль: іспит
ДРН 2 Складати тест-планів, тест-кейсів і тестових сценаріїв для ефективного покриття функціональності ПЗ	×	×	Звіт за результатами виконання практичної роботи №3 Проміжний контроль №1: тести множинного вибору Підсумковий контроль: іспит
ДРН 3 Ефективно використовувати інструменти автоматизації тестування ПЗ	×	×	Звіт за результатами виконання практичної роботи №4 Проміжний контроль №1: тести множинного вибору Підсумковий контроль: іспит
ДРН 4 Застосовувати на практиці верифікацію та валідацію для перевірки правильності роботи ПЗ.	×	×	Звіт за результатами виконання практичної роботи №7 Проміжний контроль №2: тести множинного вибору Підсумковий контроль: іспит
ДРН 5 Ідентифікувати, класифікувати та управляти дефектами ПЗ з використанням відповідних інструментів для їх моніторингу та усунення	×	×	Звіт за результатами виконання практичної роботи №10 Проміжний контроль №2: тести множинного вибору Підсумковий контроль: іспит
ДРН 6 Виконувати повне тестування програмного продукту або сервісу в різних умовах з застосуванням основних технік та видів тестування	×	×	Звіт за результатами виконання практичних робіт №5, 6, 8 та 9 Проміжний контроль №1-2: тести множинного вибору Підсумковий контроль: іспит
ДРН 7 Застосовувати метрики якості для аналізу продуктивності ПЗ, кількості дефектів та ефективності тестування.	×	×	Звіт за результатами виконання практичної роботи №12 Проміжний контроль №2: тести множинного вибору Підсумковий контроль: іспит
ДРН 8 Розуміти процеси та стандарти забезпечення якості та відповідності ПЗ вимогам користувачів і	×	×	Звіт за результатами виконання практичної роботи №11 Проміжний контроль №2: тести

стандартам індустрії.			множинного вибору Підсумковий контроль: іспит
-----------------------	--	--	--

3. ЗМІСТ ОСВІТНЬОГО КОМПОНЕНТА (ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ)

Тема. Перелік питань, що будуть розглянуті в межах теми	Розподіл в межах загального бюджету часу				Рекомендована література
	Аудиторна робота			Самостійна робота	
	Лекції	Практ. з. / семін. з.	Лаб. з.		
Тема 1. Основи тестування програмного забезпечення План 1.1 Визначення тестування та його значення 1.2 Види тестування: ручне і автоматизоване 1.3 Основні принципи тестування (наприклад, тестування демонструє наявність дефектів) 1.4 Роль тестування в життєвому циклі ПЗ 1.5 Тестування на різних рівнях розробки (юніт, інтеграційне, системне) 1.6 Помилки, дефекти та несправності: термінологія 1.7 Підходи до тестування: аналітичний, методичний, процесний	2	2	—		1, 2, 4, 5, 7, 10, 11, 16
Тема 2. Життєвий цикл програмного забезпечення План 2.1 Етапи життєвого циклу ПЗ 2.2 Роль тестування на кожному етапі SDLC 2.3 Моделі розробки ПЗ (водоспад, Agile, спіральна модель) 2.4 Взаємодія між командами розробки та тестування 2.5 Вплив процесів управління змінами на тестування 2.6 Управління ризиками на різних етапах життєвого циклу 2.7 Інтеграція тестування в Agile-розробку	2	4	—	10	1, 2, 4, 5, 7, 10, 11, 16, 19
Тема 3. Моделі процесів тестування План 3.1 Характеристика V-моделі та її зв'язок із тестуванням 3.2 Особливості тестування у Waterfall моделі 3.3 Тестування в умовах Agile-розробки 3.4 Переваги та недоліки кожної моделі тестування 3.5 Інкрементне та ітеративне тестування 3.6 Вплив вибору моделі розробки на процес тестування 3.7 Гнучкість та адаптивність тестування в умовах різних моделей	2	4	—	10	2, 4, 6, 8, 11, 16, 19
Тема 4. Планування та проектування тестів План 4.1 Створення тест-планів: структура та компоненти 4.2 Визначення критеріїв успішності тестування 4.3 Розробка тест-кейсів на основі вимог	2	4	—	10	2, 4, 5, 6, 8, 11, 16, 22

4.4 Техніки проектування тестів (еквівалентні класи, граничні значення) 4.5 Створення тестових сценаріїв для різних видів тестування 4.6 Пріоритезація тестів на основі критичності функціоналу 4.7 Оцінка та аналіз покриття тестування					
Тема 5. Методи тестування програмного забезпечення План 5.1 Тестування чорної скриньки: принципи та підходи 5.2 Тестування білої скриньки: аналіз коду та покриття 5.3 Сірий ящик: поєднання чорної та білої скриньок 5.4 Переваги та недоліки кожного підходу 5.5 Динамічне тестування: аналіз результатів виконання 5.6 Статичне тестування: аналіз коду та документації 5.7 Комбіновані методи тестування: переваги для складних систем	2	2	–	10	2, 4, 5, 6, 8, 10, 13, 16, 22
Тема 6. Автоматизація тестування План 6.1 Переваги автоматизації над ручним тестуванням 6.2 Вибір інструментів автоматизації (Selenium, JUnit) 6.3 Автоматизація регресійного тестування 6.4 Розробка та підтримка автоматизованих тестових сценаріїв 6.5 Підходи до організації автоматизації: Data-Driven, Keyword-Driven 6.6 Оцінка ефективності автоматизованого тестування 6.7 Виклики та обмеження автоматизованого тестування	2	2	–	10	2, 4, 6, 8, 10, 13, 16, 21, 23, 24
Тема 7. Верифікація та валідація програмного забезпечення План 7.1 Верифікація: відповідність розробки вимогам 7.2 Валідація: оцінка кінцевого продукту на відповідність потребам користувача 7.3 Методи та інструменти для верифікації 7.4 Техніки валідації: приймальне тестування, бета-тестування 7.5 Коли застосовувати верифікацію, а коли валідацію 7.6 Роль документування у процесах верифікації та валідації 7.7 Вплив вимог та їх змін на процес верифікації та валідації	2	2	–	10	1, 2, 4, 5, 6, 9, 12, 25
Тема 8. Різновиди тестування План 8.1 Юніт-тестування: призначення, інструменти (JUnit, NUnit), кращі практики 8.2 Інтеграційне тестування: визначення та	2	2	–	14	2, 4, 7, 8, 9, 12, 16

види (висхідне, низхідне, "big bang") 8.3 Системне тестування: функціональність та взаємодія всіх компонентів 8.4 Регресійне тестування: автоматизація регресійних тестів, вибіркове тестування 8.5 Приймальне тестування (UAT): роль користувачів у тестуванні та критерії успішності 8.6 Smoke-тестування: швидка перевірка базової функціональності ПЗ 8.7 Alpha та Beta тестування: відмінності та роль у циклі розробки					
Тема 9. Тестування продуктивності та навантаження План 9.1 Тестування продуктивності: мета та основні показники (швидкість, пропускну здатність) 9.2 Навантажувальне тестування: інструменти (JMeter, LoadRunner) та стратегії 9.3 Стрес-тестування: перевірка поведінки системи при максимальних навантаженнях 9.4 Тестування стабільності: оцінка довготривалої роботи системи 9.5 Тестування масштабованості: аналіз здатності системи до збільшення навантаження 9.6 Граничні значення: тестування системи при екстремальних вхідних даних 9.7 Тестування пікових навантажень: симуляція максимального використання системи	2	2	—		1, 2, 4, 6, 8, 9, 13, 17, 18
Тема 10. Управління дефектами План 10.1 Життєвий цикл дефекту: створення, перевірка, усунення, перевірка після виправлення 10.2 Класифікація дефектів: критичні, високі, середні, низькі за пріоритетом та впливом 10.3 Інструменти управління дефектами: Jira, Bugzilla, їх функціональні можливості 10.4 Дефектні звіти: стандарти складання та важливі поля (опис, кроки відтворення, очікуваний результат) 10.5 Причини появи дефектів: людський фактор, технічні обмеження, неправильні вимоги 10.6 Трекінг дефектів та метрики: кількість відкритих/закритих дефектів, час на їх усунення 10.7 Пріоритизація дефектів: як визначити, які дефекти виправляти першочергово	2	2	—		2, 4, 7, 8, 9, 12, 19, 20
Тема 11. Забезпечення якості програмного забезпечення План 11.1 Відмінності між Quality Assurance та Quality Control 11.2 Процеси забезпечення якості: планування, моніторинг, аналіз ризиків 11.3 Методології QA: Waterfall, Agile, DevOps	4	2	—		2, 3, 4, 5, 6, 9, 14, 15, 26, 27

підходи до забезпечення якості 11.4 Стандарти якості (ISO, IEEE): що таке стандарти та як їх застосовують 11.5 Ролі та відповідальності у команді QA: тестувальник, менеджер якості, інженер автоматизації 11.6 Процеси контролю якості: ревізії коду, тестування, огляди 11.7 Метрики для оцінки якості: дефектна щільність, пропускна здатність, кількість дефектів на тисячу рядків коду					
Тема 12. Метрики якості програмного забезпечення План 12.1 Класифікація метрик якості: процесні, продуктні, проектні 12.2 Метрики продуктивності: середній час відгуку, пропускна здатність, використання ресурсів 12.3 Метрики дефектів: кількість знайдених дефектів на етапі тестування, час на їх виправлення 12.4 Метрики покриття тестами: відсоток коду, що охоплений тестуванням 12.5 Метрики користувацького досвіду: рівень задоволеності користувачів, частота збоїв 12.6 Метрики регресійного тестування: ефективність виправлення старих дефектів та виявлення нових 12.7 Метрики автоматизації тестування: відсоток тестових сценаріїв, автоматизованих успішно	6	2	–	16	2, 3, 4, 6, 9, 14, 15, 26, 27
Всього годин	30	30		90	

4. МЕТОДИ ВИКЛАДАННЯ ТА НАВЧАННЯ

ДРН	Методи викладання (робота, що буде проведена викладачем <u>під час аудиторних занять</u> , консультацій)	Кількість годин	Методи навчання (які види навчальної діяльності має виконати <u>студент самостійно</u>)	Кількість годин
ДРН 1 Знати види, методи та підходи до тестування для виявлення можливих помилок і дефектів на різних етапах розробки програмного забезпечення (ПЗ)	- проведення лекційних та практичних занять з мультимедійними презентаціями, - практикоорієнтоване навчання аналізу життєвого циклу і тестових випадків для простого додатку з ідентифікацією точок для інтеграції тестування	4	- додаткове опрацювання лекційного матеріалу; - підготовка до захисту звітів практичних робіт; - самостійне вивчення питання застосування евристичних технік тестування у складних системах та тестування на етапі планування і розробки вимог	4
ДРН 2 Скласти тест-планів, тест-кейсів і тестових сценаріїв для ефективного покриття функціональності ПЗ	- проведення лекційних та практичних занять з мультимедійними презентаціями, - практикоорієнтоване навчання шляхом порівняння процесу	4	- додаткове опрацювання лекційного матеріалу; - підготовка до захисту звітів практичних робіт; - самостійне вивчення питання вбудованих механізмів зворотного	4

	тестування в рамках V-моделі та Agile та створення тестових сценаріїв для обох підходів.		зв'язку в Agile для швидкого виявлення дефектів	
ДРН 3 Ефективно використовувати інструменти автоматизації тестування ПЗ	- проведення лекційних та практичних занять з мультимедійними презентаціями, - практикоорієнтоване навчання моделюванню тестового процесу зі налаштуванням та інтеграцією елементів автоматизованого тестування.	4	- додаткове опрацювання лекційного матеріалу; - підготовка до захисту звітів практичних робіт; - самостійне вивчення питання інтеграції автоматизованого тестування в конвеєр CI/CD	4
ДРН 4 Застосовувати на практиці верифікацію та валідацію для перевірки правильності роботи ПЗ.	- проведення лекційних та практичних занять з мультимедійними презентаціями, - практикоорієнтоване навчання шляхом розробки юніт-тестів для окремих модулів програми та інтеграційних тестів для перевірки взаємодії між модулями.	4	- додаткове опрацювання лекційного матеріалу; - підготовка до захисту звітів практичних робіт; - самостійне вивчення питання тестування інтеграцій в середовищах з мікросервісами та API	4
ДРН 5 Ідентифікувати, класифікувати та управляти дефектами ПЗ з використанням відповідних інструментів для їх моніторингу та усунення	- проведення лекційних та практичних занять з мультимедійними презентаціями, - практикоорієнтоване навчання аналізу дефектних звітів, трекінгу дефектів за допомогою систем управління дефектами.	2	- додаткове опрацювання лекційного матеріалу; - підготовка до захисту звітів практичних робіт; - самостійне вивчення питання управління пріоритизацією дефектів у великомасштабних проєктах	6
ДРН 6 Виконувати повне тестування програмного продукту або сервісу в різних умовах з застосуванням основних технік та видів тестування	- проведення лекційних та практичних занять з мультимедійними презентаціями, - практикоорієнтоване навчання шляхом оцінки та порівняння ефективності інструментів для автоматизації різних типів тестування, включно з навантажувальним тестуванням, тестуванням на проникнення.	10	- додаткове опрацювання лекційного матеріалу; - підготовка до захисту звітів практичних робіт; - самостійне вивчення питання використання Docker для ізоляції середовищ тестування, визначення оптимальних критеріїв для тестування продуктивності, технологій імітацій кіберзагроз для перевірки безпеки системи	20
ДРН 7 Застосовувати метрики якості для аналізу продуктивності ПЗ, кількості дефектів та ефективності тестування.	- проведення лекційних та практичних занять з мультимедійними презентаціями, - практикоорієнтоване	12	- додаткове опрацювання лекційного матеріалу; - підготовка до захисту звітів практичних робіт; - самостійне вивчення	24

	навчання вимірюванню метрик якості, таких як дефектна щільність, середній час на виправлення дефектів.		питання аналізу вартості виправлення дефектів на різних етапах розробки	
ДРН 8 Розуміти процеси та стандарти забезпечення якості та відповідності ПЗ вимогам користувачів і стандартам індустрії.	- проведення лекційних та практичних занять з мультимедійними презентаціями, - практикоорієнтоване навчання шляхом аналізу комплексної системи забезпечення якості, що включає перевірку відповідності стандартам якості.	20	- додаткове опрацювання лекційного матеріалу; - підготовка до захисту звітів практичних робіт; - самостійне вивчення питання впливу взаємодії команди QA з іншими командами (розробки, продуктового менеджменту, підтримки)	24
Всього:		60		90

5. ОЦІНЮВАННЯ ЗА ОСВІТНІМ КОМПОНЕНТОМ

5.1. Сумативне оцінювання

5.1.1 Для оцінювання очікуваних результатів навчання передбачено

№	Методи сумативного оцінювання	Бали / Вага у загальній оцінці	Дата складання
Весняний семестр			
1.	Практична робота 1. Розробка тестових випадків для простого додатку	4 бали / 4%	До 2 тижня
2.	Практична робота 2. Аналіз життєвого циклу ІТ- проєкту	4 бали / 4%	До 3 тижня
3.	Практична робота 3. Створення тестових сценаріїв в рамках V-моделі та Agile	4 бали / 4%	До 4 тижня
4.	Практична робота 4. Створення автоматизованих тестів	4 бали / 4%	До 5 тижня
5.	Практична робота 5. Інструменти для управління тестовими випадками	4 бали / 4%	До 6 тижня
6.	Практична робота 6. Тестування на проникнення	4 бали / 4%	До 7 тижня
7.	Проміжний контроль (Тести множинного вибору)	6 балів / 6%	До 8 тижня
8.	Самостійна робота здобувача вищої освіти	5 балів / 5%	До 8 тижня
9.	Практична робота 7. Розробка модульних та інтеграційних тестів	4 бали / 4%	До 9 тижня
10.	Практична робота 8. Розробка тестових сценаріїв для юніт-тестування та регресійного тестування	4 бали / 4%	До 10 тижня
11.	Практична робота 9. Інструменти навантажувального тестування	4 бали / 4%	До 11 тижня
12.	Практична робота 10. Трекінг дефектів ПЗ	4 бали / 4%	До 12 тижня
13.	Практична робота 11. Розробка плану забезпечення якості для ІТ-проєкту	4 бали / 4%	До 13 тижня
14.	Практична робота 12. Розробка системи метрик для оцінки якості автоматизованого тестування	4 бали / 4%	До 14 тижня
15.	Проміжний контроль №2 (Тести множинного вибору)	6 балів / 6%	До 15 тижня

16.	Самостійна робота студента	5 балів / 5%	До 15 тижня
17.	Іспит	30 балів /30%	

5.1.2 Критерії оцінювання

Компонент	Незадовільно	Задовільно	Добре	Відмінно
Весняний семестр				
Практична робота 1. Розробка тестових випадків для простого додатку	<1 балів	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі
Практична робота 2. Аналіз життєвого циклу IT-проєкту	<1 балів	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі
Практична робота 3. Створення тестових сценаріїв в рамках V-моделі та Agile	<1 балів	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі
Практична робота 4. Створення автоматизованих тестів	<1 балів	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі
Практична робота 5. Інструменти для управління тестовими випадками	<1 балів	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі

Практична робота 6. Тестування на проникнення	<1 балів	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі
Проміжний контроль №1 (Тести множинного вибору)	<3 балів	4 бали	5 балів	6 балів
	Кількість правильних відповідей на тест із 12-ти, kv: kv ≤ 5	Кількість правильних відповідей на тест, kv: 6 ≤ kv ≤ 8.	Кількість правильних відповідей на тест, kv: 9 ≤ kv ≤ 10	Кількість правильних відповідей на тест, kv: kv ≥ 11
Практична робота 7. Розробка модульних та інтеграційних тестів	<1 балів	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі
Практична робота 8. Розробка тестових сценаріїв для юніт-тестування та регресійного тестування	<1 балів	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі
Практична робота 9. Інструменти навантажувального тестування	<1 балів	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі
Практична робота 10. Трекінг дефектів ПЗ	<1 балів	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі
Практична робота 11. Розробка плану забезпечення якості для ІТ-проекту	<1 балів	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі

Практична робота 12. Розробка системи метрик для оцінки якості автоматизованого тестування	<i><1 балів</i>	1-2 бали	3 балів	4 балів
	Практична робота не виконана або виконана не правильно	Виконані не всі завдання практичної роботи, завантажений недостатній обсяг даних, порушена логіка	Завдання та описові схеми виконані з незначними помилками, здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі	Всі завдання практичної роботи виконано, здобувач вищої освіти добре орієнтується в теоретичному матеріалі
Проміжний контроль №2 (Тести множинного вибору)	<i><3 балів</i>	4 бали	5 балів	6 балів
	Кількість правильних відповідей на тест із 12-ти, kv: kv ≤ 5	Кількість правильних відповідей на тест, kv: $6 \leq kv \leq 8$.	Кількість правильних відповідей на тест, kv: $9 \leq kv \leq 10$	Кількість правильних відповідей на тест, kv: $kv \geq 11$
Іспит	<i>0-9 балів</i>	<i>10-16 балів</i>	<i>17-24 балів</i>	<i>25-30 балів</i>
	Здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі, розрахункове завдання не виконано	Здобувач вищої освіти недостатньо орієнтується в теоретичному матеріалі, розрахункове завдання виконано з помилками	Здобувач вищої освіти достатньо орієнтується в теоретичному матеріалі, розрахункове завдання виконано	Здобувач вищої освіти гарно орієнтується в теоретичному матеріалі, розрахункове завдання виконано

5.2. Формативне оцінювання:

№	Елементи формативного оцінювання	Дата
1	Усне опитування після вивчення кожної теми	Після завершення вивчення теми
2	Проходження тестування з атестації та проміжного контролю зі зворотнім зв'язком з викладачем	Відповідно до графіку навчального процесу
3	Проходження тестування після закінчення вивчення кожної теми для самостійного контролю знань та підготовки до складання заліку (іспиту)	Регулюється здобувачем вищої освіти самостійно
4	Захист практичних робіт	Через тиждень після їх здачі
5	Усний зворотний зв'язок від викладача під час роботи над практичними роботами протягом занять	Протягом всього семестру

6. НАВЧАЛЬНІ РЕСУРСИ (ЛІТЕРАТУРА)

6.1. Основні джерела

1. Цибульник С. О., Барандич К. С. Технології розроблення програмного забезпечення. Частина 1. Життєвий цикл програмного забезпечення. Підручник. Київ КПІ ім. Ігоря Сікорського 2022. – 270 с. – https://ela.kpi.ua/bitstream/123456789/50623/1/TRPZ_Ch1_ZhTsPZ.pdf

2. Kulikov S. Software Testing. 3rd edition. EPAM System. Book version 3.2.5 as of 2023-12-20. – 280 p. - http://svyatoslav.biz/software_testing_book_download_en/
3. ДСТУ ISO/IEC 25023:2019 Інженерія систем і програмних засобів. Вимоги до якості систем програмних засобів та їхнього оцінювання (SQuaRE). Вимірювання якості систем та програмних продуктів (ISO/IEC 25023:2016, IDT). https://online.budstandart.com/ua/catalog/doc-page?id_doc=85793
4. Старух А.І. Конспект лекцій з навчальної дисципліни "Методологія тестування програмного забезпечення". Львів. 2020. - 43 с.
<https://financial.lnu.edu.ua/wp-content/uploads/2019/09/konspekt-testuvannia.pdf>

6.2. Методичне забезпечення

1. Тестування та забезпечення якості програмного забезпечення(QA) – електронний курс на платформі Moodle - <https://cdn.snau.edu.ua/moodle/course/view.php?id=4640>
2. Тестування та забезпечення якості програмного забезпечення(QA): конспект лекцій. Суми: СНАУ, 2023. - 83 с.
3. Тестування та забезпечення якості програмного забезпечення(QA): Методичні рекомендації для виконання практичних робіт. Суми: СНАУ, 2022. - 87 с.
4. Тестування та забезпечення якості програмного забезпечення(QA): : Методичні рекомендації для виконання самостійної роботи. Для здобувачів першого(бакалаврського) рівня вищої освіти спеціальності 126 «Інформаційні системи та технології» освітнього ступеню «бакалавр» Суми, 2023. - 55с.

6.3. Інші джерела

5. Безкоштовний онлайн-курс «Основи тестування програмного забезпечення» - <https://training.qatestlab.com/course/software-testing-fundamentals/>
6. Witte F. Strategy, Planning and Organization of Test Processes: Basis for Successful Project Execution in Software Testing – Springer, 2022. — 257 p.
7. Bierig R., Brown S., Galván E., Timoney J. Essentials of Software Testing – Cambridge: Cambridge University Press, 2021. — 323 p.
8. O'Regan O. Concise Guide to Software Testing – New York: Springer, 2019. — 309 p.
9. Abu Sayed Mahfuz Software Quality Assurance: Integrating Testing, Security, and Audit – Auerbach, 2021. – 392 p.

6.4. Програмне забезпечення

10. JUnit (Java) та NUnit (.NET) – фреймворки для юніт-тестування
11. Apache JMeter – інструмент для тестування продуктивності веб-додатків
12. LoadRunner – інструмент для навантажувального тестування великих додатків
13. Jira – система для управління проектами та відслідковування дефектів,
14. Bugzilla – система відслідковування дефектів з відкритим кодом,
15. Selenium –інструмент для автоматизації веб-тестування,
16. TestRail – інструмент для управління тестовими випадками
17. OWASP ZAP – інструмент для тестування на проникнення і виявлення вразливостей
18. Burp Suite – інструмент для аналізу та тестування безпеки веб-додатків.
19. Jenkins – інструмент для побудови CI/CD конвеєрів, що дозволяє інтегрувати тестування в процес розробки
20. Allure – система для генерації звітів про результати автоматизованих тестів.
21. SonarQube – інструмент для аналізу якості коду та тестування.