

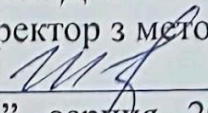
**ЗАКЛАД ВИЩОЇ ОСВІТИ
«УНІВЕРСИТЕТ КОРОЛЯ ДАНИЛА»**

Факультет суспільних і прикладних наук

Кафедра інформаційних технологій

ЗАТВЕРДЖУЮ

Проректор з методичної роботи

 Ярослав ШТАНЬКО

“ 29 ” серпня 2025 р.

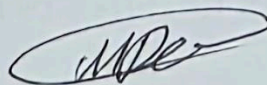
УПРАВЛІННЯ ЯКІСТЮ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Галузь знань:	12 Інформаційні технології
Спеціальність:	121 Інженерія програмного забезпечення
Освітньо-професійна (освітньо-наукова) програма:	Інженерія програмного забезпечення
Освітній рівень:	другий (магістерський)
Статус дисципліни:	обов'язкова
Мова викладання, навчання та оцінювання:	українська

**Івано-Франківськ
2025**

РОЗРОБНИК:



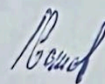
к.т.н., доцент Микола

ДЕМЧИНА

ЗАТВЕРДЖЕНО:

на засіданні кафедри інформаційних технологій, протокол № 1 від 28.08.2025

PhD, завідувач кафедри



Олександр ІВАНОВ

УЗГОДЖЕНО:

Гарант ОП

PhD, завідувач кафедри



Олександр ІВАНОВ

СХВАЛЕНО:

на засіданні Науково-методичної ради, протокол № 1 від 29.08.2025

e-mail	mykola.demchyna@ukd.edu.ua
Номер аудиторії чи кафедри	Кафедра інформаційних технологій, ауд. 206
Посилання на сайт УКД	https://ukd.edu.ua
Сторінка курсу в СДО	https://online.ukd.edu.ua

ВСТУП

Анотація навчальної дисципліни

Мета вивчення дисципліни “Управління якістю ПЗ” – надати студентам всеосяжного розуміння принципів, процесів та методів, які використовуються для забезпечення якості програмних продуктів. Оскільки попит на надійні, ефективні та високопродуктивні програмні додатки продовжує зростати, здатність ефективно управляти якістю програмного забезпечення є критичною для професіоналів у галузі розробки програмного забезпечення.

Основні завдання, що стоять перед авторами курсу, полягають в тому, щоб розкрити здобувачам:

- основи управління якістю програмного забезпечення, включаючи відповідну термінологію, поняття та атрибути якості;
- огляд різних моделей життєвого циклу розробки програмного забезпечення (SDLC) та інтеграцію практик управління якістю в кожну модель;
- дослідити різні методи та техніки, які використовуються для забезпечення якості програм, планування, проектування, тестування та безперервного вдосконалення;
- обговорити важливість якості програмного забезпечення в Agile та DevOps-середовищах та надати практичні стратегії для підтримки якості в процесах розробки;
- ознайомити студентів із застосуваннями інструментів управління якістю та ресурсами, а також навчити їх ефективно вибирати та застосовувати ці інструменти.

В результаті вивчення дисципліни студент повинен знати:

- основні принципи та цілі управління якістю програмного забезпечення.
- атрибути якості у розробці програмного забезпечення;
- моделі життєвого циклу розробки програмного забезпечення та їх взаємозв'язок з управлінням якістю програмного забезпечення;
- процес забезпечення якості програмного забезпечення, включаючи діяльність, методики та міжнародні стандарти, такі як ISO 9001 та CMMI;
- методи, які використовуються для планування, проектування, тестування.

В результаті вивчення дисципліни студент повинен вміти:

- визначати і застосовувати поняття, пов'язані з управлінням якістю програмного забезпечення в своїх програмних проектах;
- розробляти плани якості, які включають цілі, завдання, метрики та очікування для конкретних програмних проектів;
- визначати і вирішувати можливі ризики якості програмного забезпечення на різних етапах життєвого циклу розробки програмного забезпечення;

- впроваджувати різні техніки тестування для забезпечення якості програмного забезпечення та виправлення проблем, до того, як вони вплинуть на користувачів.

Компетентності та результати навчання, яких набувають здобувачі освіти внаслідок вивчення навчальної дисципліни (шифри та зміст компетентностей та програмних результатів навчання вказано відповідно до ОПП/ОНП «Інженерія програмного забезпечення»).

Шифр та назва компетентності	Шифр та назва програмних результатів навчання
ЗК01. Здатність до абстрактного мислення, аналізу та синтезу	РН06. Розробляти і оцінювати стратегії проєктування програмних засобів; обґрунтовувати, аналізувати і оцінювати варіанти проектних рішень з точки зору якості кінцевого програмного продукту, ресурсних обмежень та інших факторів. РН11. Забезпечувати якість на всіх стадіях життєвого циклу програмного забезпечення, у тому числі з використанням релевантних моделей та методів оцінювання, а також засобів автоматизованого тестування і верифікації програмного забезпечення. РН16. Планувати, організовувати та здійснювати тестування, верифікацію та валідацію програмного забезпечення.
ЗК02. Здатність проводити дослідження на відповідному рівні.	
ЗК05. Здатність генерувати нові ідеї (креативність).	
СК01. Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення.	
СК02. Здатність розробляти і реалізовувати наукові та/або прикладні проєкти у сфері інженерії програмного забезпечення.	
СК03. Здатність проєктувати архітектуру програмного забезпечення, моделювати процеси функціонування окремих підсистем і модулів.	
СК04. Здатність розвивати і реалізовувати нові конкурентоспроможні ідеї в інженерії програмного забезпечення.	
СК05. Здатність розробляти, аналізувати та застосовувати специфікації, стандарти, правила і рекомендації в сфері інженерії програмного забезпечення.	
СК06. Здатність ефективно керувати фінансовими, людськими, технічними та іншими проєктними ресурсами у сфері інженерії програмного забезпечення.	

СК07. Здатність критично осмислювати проблеми у галузі інформаційних технологій та на межі галузей знань, інтегрувати відповідні знання та розв'язувати складні задачі у широких або мультидисциплінарних контекстах.	
СК08. Здатність розробляти і координувати процеси, етапи та ітерації життєвого циклу програмного забезпечення на основі застосування сучасних моделей, методів та технологій розроблення програмного забезпечення.	
СК09. Здатність забезпечувати якість програмного забезпечення.	

ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Курс	1		
Семестр	2		
Кількість кредитів ECTS	3		
		Денна форма	Заочна форма
Аудиторні навчальні заняття	лекції	14 (в годинах)	4 (в годинах)
	семінари, практичні	16 (в годинах)	4 (в годинах)
Самостійна робота		60 (в годинах)	82 (в годинах)
Форма підсумкового контролю	Екзамен		

СТРУКТУРНО-ЛОГІЧНА СХЕМА ВИВЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Пререквізити	Постреквізити
Методологія розробки ПЗ	Кваліфікаційна робота

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Перелік тем лекційного матеріалу

Тема 1. Введення в управління якістю програмного забезпечення (2 год.)

1. Визначення якості програмного забезпечення.
2. Важливість управління якістю програмного забезпечення.
3. Цілі та завдання.
4. Принципи управління якістю програмного забезпечення.
5. Проблеми в управлінні якістю програмного забезпечення.

Тема 2. Атрибути якості програмного забезпечення (2 год.)

1. Функціональність, надійність, зручність використання, ефективність, підтримуваність та мобільність в тестуванні.
2. Встановлення та вимірювання атрибутів якості.
3. Роль атрибутів якості в розробці програмного забезпечення.
4. Приклади поганої та хорошої якості програмних систем.

Тема 3. Життєвий цикл розробки програмного забезпечення (SDLC) і якість (2 год.)

1. Огляд SDLC та його різних моделей: Waterfall, Agile, Incremental тощо.
2. Включення управління та забезпечення якості в кожен модель SDLC.
3. Визначення потенційних ризиків якості та проблем на кожному етапі SDLC.
4. Зв'язок між якістю програмного забезпечення та управлінням проектом.

Тема 4. Забезпечення якості програмного забезпечення (SQA) (2 год.)

1. Визначення та цілі SQA.
2. Встановлення процесу SQA для організації.
3. Діяльність і методи SQA, такі як планування та контроль якості, огляди та аудити та управління конфігурацією.
4. Знайомство з міжнародними стандартами та фреймворками SQA: ISO 9001, CMMI, IEEE та ін.

Тема 5. Планування якості програмного забезпечення (2 год.)

1. Встановлення цілей і завдань якості на основі потреб зацікавлених сторін.
2. Визначення ознак і критеріїв якості програмних продуктів.
3. Розробка планів якості, включаючи вибір показників, методів та інструментів.
4. Розподіл ресурсів і визначення ролей і відповідальності для управління якістю.

Тема 6. Вимоги до якості (2 год.)

1. Роль вимог в управлінні якістю програмного забезпечення.
2. Виявлення та визначення вимог для забезпечення високої якості.
3. Методи перевірки та підтвердження вимог.

4. Методи управління змінами вимог та їх вплив на якість.

Тема 7. Якість дизайну програмного забезпечення (2 год.)

1. Розуміння принципів дизайну та атрибутів, які сприяють якості.
2. Вимірювання та оцінка якості дизайну за допомогою показників та інструментів.
3. Методи покращення якості дизайну, такі як модульність, когезія, зчеплення та багаторазове використання.
4. Проектування для різних атрибутів якості: продуктивність, безпека та ремонтпридатність.

Зміст практичних занять

Тема 1. Розуміння атрибутів якості та показників (2 год)

Мета: Ознайомити студентів з різними атрибутами якості та пов'язаними з ними показниками.

Діяльність: Групове обговорення та аналіз різних атрибутів якості в розробці програмного забезпечення, таких як надійність, зручність обслуговування, продуктивність і зручність використання. Студенти визначають відповідні показники для кожного атрибута та обговорюють їхнє значення.

Результати: список атрибутів якості та відповідних їм показників із прикладами.

Тема 2. Розробка плану управління якістю програмного забезпечення (2 год)

Мета: Надати студентам можливість створити комплексний план управління якістю для гіпотетичного проекту розробки програмного забезпечення.

Діяльність: Студентам буде запропоновано розробити план управління якістю для певного сценарію розробки програмного забезпечення, вказавши цілі, завдання, вимоги до якості та розподіл ресурсів.

Результати: Письмовий план управління якістю програмного забезпечення, включаючи плани заходів із забезпечення та контролю якості.

Тема 3. Виконання функціонального та нефункціонального тестування (2 год)

Мета: Надати практичний досвід як функціональних, так і нефункціональних методів тестування в контрольованому середовищі.

Діяльність: Студенти застосовуватимуть методи функціонального тестування, такі як тестування чорної скриньки та тестування білої скриньки, до зразка коду, наданого викладачем. Вони також виконуватимуть нефункціональні тести, включаючи перевірку продуктивності, зручності використання та безпеки.

Результати: Докладні тестові звіти, що документують результати функціонального та нефункціонального тестування.

Тема 4. Розробка та виконання тестового прикладу (Test Case) (2 год)

Мета: Навчити студентів розробляти та виконувати ефективні тестові приклади для різних сценаріїв тестування.

Діяльність: Студенти створюватимуть тестові приклади для різних сценаріїв тестування, таких як модульне тестування, інтеграційне тестування та тестування системи. Потім вони виконають свої тести та проаналізують результати.

Результати: Документація тестового випадку, результати тестування та висновки на основі виконання тесту.

Тема 5. Впровадження управління якістю в гнучкі (agile) проекти (2 год)

Мета: Надати практичний досвід впровадження практик управління якістю в рамках Agile.

Діяльність: Студенти візьмуть участь у симульованому проекті Agile-розробки, що включає рольову гру як Scrum Master, Product Owner та члени команди розробників. Вони плануватимуть і здійснюватимуть заходи з управління якістю протягом усього процесу Agile.

Результати: Письмове обговорення досвіду, отриманих уроків і вдосконалень, які можна внести в майбутні проекти Agile.

Тема 6. Управління ризиками якості програмного забезпечення (2 год)

Мета: Навчити студентів визначати, аналізувати та керувати потенційними ризиками, пов'язаними з якістю програмного забезпечення.

Діяльність: У малих групах студенти обмірковують потенційні ризики для якості програмного забезпечення в певному сценарії проекту. Потім вони обговорять стратегії пом'якшення цих ризиків і розроблять план дій у надзвичайних ситуаціях.

Результати: Перелік виявлених ризиків, стратегії зменшення ризиків і план дій у непередбачених ситуаціях.

Тема 7. Аналіз випадків управління якістю програмного забезпечення в реальному світі (2 год)

Мета: Допомогти студентам критично оцінити успіхи та проблеми управління якістю програмного забезпечення в реальному світі.

Діяльність: Студентам будуть надані тематичні дослідження минулих проектів розробки програмного забезпечення, підкреслюючи впроваджені процеси управління якістю та результати. Потім вони проаналізують кожен випадок, обговорять успіхи та проблеми та запропонують покращення для майбутніх починань.

Результати: Письмовий аналіз і рефлексія тематичних досліджень, що включає найкращі практики та отримані уроки.

Тема 8. Підсумок та відгуки (2 год)

Мета: Консолідувати ключові принципи, методи та найкращі практики управління якістю програмного забезпечення, які обговорювалися протягом курсу, і зібрати відгуки студентів про їхній досвід навчання.

Зміст самостійної роботи здобувачів

Розподіл годин, виділених на вивчення дисципліни:

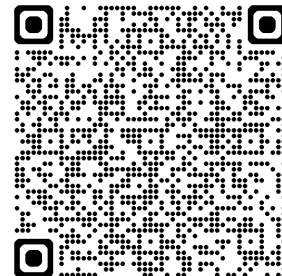
Найменування видів робіт	Розподіл годин	
	денна форма	заочна форма
Самостійна робота, год, у т.ч.:	60	82
Опрацювання матеріалу, викладеного на лекціях	6	8
Підготовка до практичних занять та контрольних заходів	24	29
Підготовка звітів з практичних робіт	20	25
Підготовка до поточного контролю	5	10
Опрацювання матеріалу, винесеного на самостійне вивчення	5	10

ПОЛІТИКА КУРСУ

Коротко, з покликанням на відповідну нормативну базу УКД, висвітлити питання:

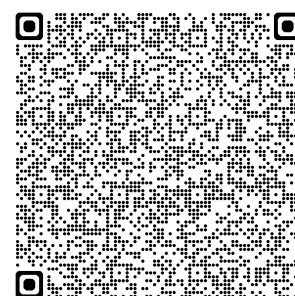
1) щодо системи поточного і підсумкового контролю

Організація поточного та підсумкового семестрового контролю знань студентів, проведення практик та атестації, переведення показників академічної успішності за 100-бальною шкалою в систему оцінок за національною шкалою здійснюється згідно з “Положенням про систему поточного і підсумкового контролю, оцінювання знань та визначення рейтингу здобувачів освіти”. Ознайомитись з документом можна за [покликанням](#).



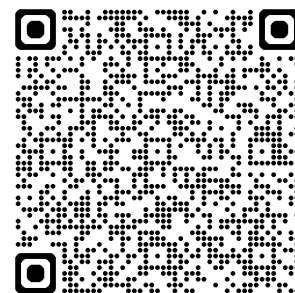
2) щодо оскарження результатів контрольних заходів

Здобувачі вищої освіти мають право на оскарження оцінки з дисципліни отриманої під час контрольних заходів. Апеляція здійснюється відповідно до «Положення про політику та врегулювання конфліктних ситуацій». Ознайомитись з документом можна за [покликанням](#).



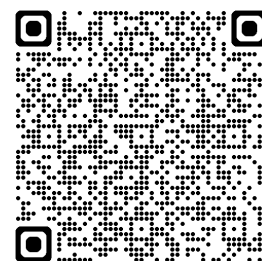
3) щодо відпрацювання пропущених занять

Згідно “Положення про організацію освітнього процесу” здобувач допускається до семестрового контролю з конкретної навчальної дисципліни (семестрового екзамену, диференційованого заліку), якщо він виконав усі види робіт, передбачені на семестр навчальним планом та силабусом/робочою програмою навчальної дисципліни, підтвердив опанування на мінімальному рівні результатів навчання (отримав ≥ 35 бали), відпрацював визначені індивідуальним навчальним планом всі лекційні, практичні, семінарські та лабораторні заняття, на яких він був відсутній. Ознайомитись з документом можна за [покликанням](#).



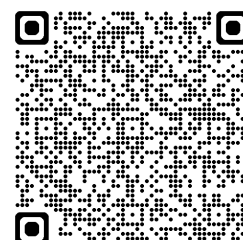
4) щодо дотримання академічної доброчесності

“Положення про академічну доброчесність” закріплює моральні принципи, норми та правила етичної поведінки, позитивного, сприятливого, доброчесного освітнього і наукового середовища, професійної діяльності та професійного спілкування спільноти Університету, викладання та провадження наукової (творчої) діяльності з метою забезпечення довіри до результатів навчання. Ознайомитись з документом можна за [покликанням](#).



5) щодо використання штучного інтелекту

“Положення про академічну доброчесність” визначає політику щодо використання технічних засобів на основі штучного інтелекту в освітньому процесі. Ознайомитись з документом можна за



[покликанням](#).¹ “Положення про систему запобігання та виявлення академічного плагіату, самоплагіату, фабрикації та фальсифікації академічних творів” містить рекомендації щодо використання в академічних текстах генераторів на основі штучного інтелекту. Ознайомитись з документом можна за [покликанням](#).

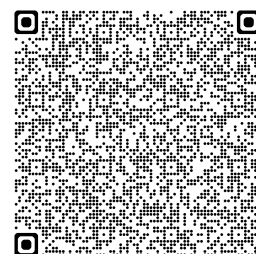
6) щодо використання технічних засобів в аудиторії та правила комунікації

Використання мобільних телефонів, планшетів та інших гаджетів під час лекційних та практичних занять дозволяється виключно у навчальних цілях (для уточнення певних даних, перевірки правопису, отримання довідкової інформації тощо). На гаджетах повинен бути активований режим «без звуку» до початку заняття. Під час занять заборонено надсилання текстових повідомлень, прослуховування музики, перевірка електронної пошти, соціальних мереж тощо. Під час виконання заходів контролю використання гаджетів заборонено (за винятком, коли це передбачено умовами його проведення). У разі порушення цієї заборони результат анулюється без права перескладання.

Комунікація відбувається через електронну пошту і сторінку дисципліни в Moodle.

7) щодо зарахування результатів навчання, здобутих шляхом формальної/інформальної освіти

Процедури визнання результатів навчання, здобутих шляхом формальної/інформальної освіти визначаються «Положенням про порядок визнання результатів навчання, здобутих шляхом неформальної та / або інформальної освіти». Ознайомитись з документом можна за [покликанням](#).²



МЕТОДИ НАВЧАННЯ

При вивченні дисципліни застосовується комплекс методів для організації навчання студентів з метою розвитку їх логічного та абстрактного мислення, творчих здібностей, підвищення мотивації до навчання та формування особистості майбутнього фахівця.

Програмний результат навчання³	Метод навчання	Метод оцінювання
Розробляти і оцінювати стратегії проектування програмних засобів; обґрунтовувати, аналізувати і оцінювати варіанти проектних рішень з точки зору якості кінцевого програмного продукту,	словесні методи (лекція, розповідь пояснення); наочні методи (ілюстрування, комп'ютерні і мультимедійні методи, зразки коду, зразки програм);	екзамен

¹ визначається політика використання ІІІ в навчальній дисципліні - дозволене/заборонене, правила використання

² визначається перелік електронних та інших ресурсів та умови перезарахування

³ для вибірових навчальних дисциплін вказується **результат навчання**

ресурсних обмежень та інших факторів. Забезпечувати якість на всіх стадіях життєвого циклу програмного забезпечення, у тому числі з використанням релевантних моделей та методів оцінювання, а також засобів автоматизованого тестування і верифікації програмного забезпечення. Планувати, організовувати та здійснювати тестування, верифікацію та валідацію програмного забезпечення.	інтерактивні методи (дискусія, мозковий штурм, робота в команді (групах)); метод симулювання проектів;	
---	--	--

ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Контрольні заходи

<i>Вид</i>	<i>Зміст</i>	<i>% від загальної оцінки</i>	<i>Бал</i>	
			<i>min</i>	<i>max</i>
Поточні контрольні заходи	всього	60	35	60
Підсумкові контрольні заходи	екзамен	40	25	40
Всього:		100	60	100

Процедура проведення контрольних заходів, а саме поточного контролю знань протягом семестру та підсумкового семестрового контролю, регулюється «Положенням про систему поточного та підсумкового контролю оцінювання знань та визначення рейтингу студентів».

Фіксація **поточного** контролю здійснюється в “Електронному журналі обліку успішності академічної групи” на підставі чотирибальної шкали - “2”; “3”; “4”; “5”. У разі відсутності студента на занятті виставляється “н”. За результатами поточного контролю у Журналі, автоматично визначається підсумкова оцінка, здійснюється підрахунок пропущених занять.

Усі пропущені заняття, а також негативні оцінки студенти зобов'язані відпрацювати впродовж трьох наступних тижнів. У випадку недотримання цієї норми, замість “н” в журналі буде виставлено “0” (нуль балів), без права перездачі. Відпрацьоване лекційне заняття в електронному журналі позначається літерою «в».

Критерії оцінювання (за необхідності, поточного та/або підсумкового контролю):

«незадовільно»	володіє навчальним матеріалом на рівні елементарного розпізнавання і відтворення окремих фактів, елементів, об'єктів, що позначаються окремими словами чи реченнями; володіє матеріалом на елементарному рівні засвоєння, викладає його уривчастими реченнями, виявляє здатність висловити думку на елементарному рівні; володіє матеріалом на рівні окремих фрагментів, що становлять незначну частину навчального матеріалу;
«задовільно»	володіє матеріалом на початковому рівні, значну частину матеріалу відтворює на репродуктивному рівні; володіє матеріалом на рівні, вищому за початковий, здатний за допомогою викладача логічно відтворити значну його частину; може відтворити значну частину теоретичного матеріалу, виявляє знання і розуміння основних положень, за допомогою викладача може аналізувати навчальний матеріал, порівнювати та робити висновки, виправляти допущені помилки;
«добре»	здатний застосовувати вивчений матеріал на рівні стандартних ситуацій, частково контролювати власні навчальні дії, наводити окремі власні приклади на підтвердження певних тверджень: вміє порівнювати, узагальнювати, систематизувати інформацію під керівництвом викладача, в цілому самостійно застосовувати її на практиці, контролювати власну діяльність, виправляти помилки і добирати аргументи на підтвердження певних думок під керівництвом викладача;

	вільно володіє вивченим обсягом матеріалу, та вміє застосовувати його на практиці; вільно розв'язує задачі в стандартних ситуаціях, самостійно виправляє допущені помилки, добирає переконливі аргументи на підтвердження вивченого матеріалу;
«Відмінно»	виявляє початкові творчі здібності, самостійно визначає окремі цілі власної навчальної діяльності, оцінює окремі нові факти, явища, ідеї; знаходить джерела інформації та самостійно використовує їх відповідно до цілей, поставлених викладачем; вільно висловлює власні думки і відчуття, визначає програму особистої пізнавальної діяльності, самостійно оцінює різноманітні життєві явища і факти, виявляючи особисту позицію щодо них; без допомоги викладача знаходить джерела інформації і використовує одержані відомості відповідно до мети та завдань власної пізнавальної діяльності; використовує набуті знання і вміння в нестандартних ситуаціях; виявляє особливі творчі здібності, самостійно розвиває власні обдарування і нахили, вміє самостійно здобувати знання.

До підсумкового контролю допускаються студенти які за результатами поточного контролю отримали не менше 35 балів. Усі студенти, що отримали 34 балів і менше, не допускаються до складання підсумкового контролю і на підставі укладання додаткового договору, здійснюють повторне вивчення дисципліни впродовж наступного навчального семестру. За результатами підсумкового контролю (диференційований залік/екзамен) студент може отримати 40 балів. Студенти, які під час підсумкового контролю отримали 24 бали і менше, вважаються такими, що не здали екзамен/диференційований залік і повинні йти на перездачу.

Загальна семестрова оцінка з дисципліни, яка виставляється в екзаменаційних відомостях оцінюється в балах (згідно з **Шкалою оцінювання знань за ЄКТС**) і є сумою балів отриманих під час поточного та підсумкового контролю.

Шкала оцінювання знань за ЄКТС:

Оцінка за національною шкалою	Рівень досягнень, %	Шкала ECTS
Національна диференційована шкала		
Відмінно	90 – 100	A
Добре	83 – 89	B
	75 – 82	C
Задовільно	67 – 74	D

	60 – 66	E
Незадовільно	35 – 59	FX
	0 – 34	F
Національна недиференційована шкала		
Зараховано	60 – 100	-
Не зараховано	0 – 59	-

Студенти, які не з'явилися на заліки/екзамени без поважних причин, вважаються такими, що одержали незадовільну оцінку.

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ⁴

Основна література

1. Forsgren, N., & Kersten, M. (2020). The Project to Product Transformation: How to Modernize and Migrate Your Software Portfolio.
2. Klabbers, K., & van Eeden, A. (2020). High-Performance Software Development Teams: Constructing and Leading with an Observable Commitment to Quality.
3. "Refactoring: Improving the Design of Existing Code" (2nd edition), by Martin Fowler, Kent Beck, and Erich Gamma (2021).
4. "Agile Estimating and Planning", by Mike Cohn (2020).
5. "Quality in a World Driven by Digital: The Pursuit of Quality in Software", by Girish Seshagiri (2021).
6. "Software Engineering: A Practitioner's Approach" (9th edition), by Roger Pressman and Bruce Maxim (2020).
7. "Agile Testing Condensed: A Brief Introduction", by Janet Gregory and Lisa Crispin (2022).
8. "Clean Code in Java", by Ralf Westphal (2021).
9. "Accelerating Software Quality: Machine Learning and Artificial Intelligence in the Age of DevOps", by Eran Kinsbruner (2020).
10. "A Practical Guide to Testing in DevOps", by Katrina Clokier (2021).
11. "Engineering Test-Driven Development: Realizing Test-Driven Development for Organizations", by Kyhor, Fabisch, Linz, and Stelzner (2020).
12. "Continuous Delivery Pipelines: How to Build Better Software Faster", by Dave Farley (2021).

⁴ обов'язково: враховувати вимоги [ДСТУ 8302:2015](#) (відповідно до [Наказу № 65, від 4.03. 2016](#)), [рекомендації](#) Національного агентства з забезпечення якості вищої освіти, використовувати літературу за останні 5-7 років, наводити власні публікації за змістом навчальної дисципліни.

Додаткова література

1. Chen, L., Xia, X., Wang, Q., & Yang, Z. (2020). Continuous Software Quality Assurance Based on Developer Social Network Analysis. *IEEE Transactions on Software Engineering*.
2. García, J., Vivanco, J., & Ebert, C. (2020). Quality Intelligence in Software Development. *IEEE Software*, 37(3), 68-72.