


ЗАКЛАД ВИЩОЇ ОСВІТИ
«УНІВЕРСИТЕТ КОРОЛЯ ДАНИЛА»

Факультет суспільних і прикладних наук

Кафедра інформаційних технологій

ЗАТВЕРДЖУЮ

Проректор з методичної роботи

 Ярослав ШТАНЬКО

“ — ” 2024 р.

МЕТОДОЛОГІЯ РОЗРОБКИ 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

Розробник:
професор кафедри ІТ
к.т.н., доц.



Олег ПАШКЕВИЧ

ЗАТВЕРДЖЕНО:

на засіданні кафедри інформаційних технологій,
протокол № 1 від 28.08.2024 р.

Завідувач кафедри



Сергій ВАЩИШАК

УЗГОДЖЕНО:

Гарант ОПП



Сергій ВАЩИШАК

на засіданні кафедри інформаційних технологій,
протокол № 1 від 28.08.2024 р.

Завідувач кафедри



Сергій ВАЩИШАК

СХВАЛЕНО:

на засіданні Науково-методичної ради, протокол № 1 від 30.08.2024 р.

e-mail	oleh.pashkevych@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/course/view.php?id=5868

ВСТУП

Анотація навчальної дисципліни

Навчальна дисципліна “Методологія розробки програмного забезпечення” є складовою освітньо-професійної програми підготовки фахівців за освітнім ступенем “бакалавр” галузі знань 12 “Інформаційні технології” спеціальності 121 “Інженерія програмного забезпечення”, освітньої програми “Розробка та тестування програмного забезпечення”.

Дисципліна надає студентам знання про сучасні методології та підходи до розробки ПЗ, включаючи класичні та гнучкі методології. Курс охоплює життєвий цикл ПЗ, управління проектами, вибір методологій, і забезпечує практичні навички в управлінні командами та створенні технічної документації.

Мета вивчення дисципліни – навчити студентів принципам, процесам та практикам розробки програмного забезпечення з використанням різних методологій, підготувати їх до професійної діяльності у сфері інженерії програмного забезпечення.

Для досягнення мети поставлені такі основні **завдання**:

- навчити студентів понять, принципів, методології та технологій створення програмних продуктів як сукупності процесів розроблення програмних систем на засадах життєвого циклу (ЖЦ) програмного забезпечення інформаційних систем;
- вивчення стандартів програмної інженерії; вивчення методів створення вимог під час розроблення програмних продуктів;
- вивчення методів та засобів тестування програмне забезпечення, вивчення стандартів якості під час створення програмного забезпечення інформаційних систем;
- вивчення інструментальних засобів програмної інженерії та їх практичне застосування під час проектування програмного забезпечення.

У результаті засвоєння курсу студент повинен **знати**:

- основні методології розробки ПЗ (класичні, гнучкі, DevOps);
- принципи управління життєвим циклом ПЗ;
- сучасні інструменти для розробки та підтримки програмного забезпечення.

студент повинен **вміти**:

- застосовувати знання для розв’язування задач аналізу та синтезу засобів, характерних для спеціальності;
- правильно вибрати структуру даних для конкретної задачі;
- інтегрувати необхідні структури даних в свої програми для вирішення поставлених задач.

Професійні компетентності та результати навчання, яких набувають здобувачі внаслідок вивчення навчальної дисципліни «Методологія розробки програмного забезпечення» (шифри та зміст компетентностей та програмних результатів вказані відповідно до освітньої програми “Інженерія програмного забезпечення”, введеної в дію ЗВО “Університет Короля Данила” 01.09.2024 року)

Шифр та назва компетентності	Шифр та назва результату навчання
ІК. Здатність особи розв’язувати складні задачі і проблеми у певній галузі професійної діяльності або у процесі навчання, що передбачає проведення досліджень та/або здійснення інновацій та характеризується невизначеністю умов і вимог.	РН01 Знати і застосовувати сучасні професійні стандарти і інші нормативно-правові документи з інженерії програмного забезпечення. РН02 Оцінювати і вибирати ефективні методи і моделі розроблення, впровадження, супроводу програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу.
ЗК01. Здатність до абстрактного мислення, аналізу та синтезу	РН04 Виявляти інформаційні потреби і класифікувати дані для проєктування програмного забезпечення.
ЗК02. Здатність спілкуватися іноземною мовою як усно, так і письмово.	РН06 Розробляти і оцінювати стратегії проєктування програмних засобів; обґрунтовувати, аналізувати і оцінювати варіанти проєктних рішень з точки зору якості кінцевого програмного продукту, ресурсних обмежень та інших факторів.
СК01. Здатність аналізувати предметні області, формувати, класифікувати вимоги до програмного забезпечення	РН12 Приймати ефективні організаційно-управлінські рішення в умовах невизначеності та зміни вимог, порівнювати альтернативи, оцінювати ризики.
СК03. Здатність проєктувати архітектуру програмного забезпечення, моделювати процеси функціонування окремих підсистем і модулів	РН13 Конфігурувати програмне забезпечення, керувати його змінами та розробленням програмної документації на всіх етапах життєвого циклу.
СК05. Здатність розробляти, аналізувати та застосовувати специфікації, стандарти, правила і рекомендації в 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	
СК09. Здатність забезпечувати якість програмного забезпечення	

ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Курс	1		
Семестр	2		
Кількість кредитів ЄКТС	6		
Аудиторні навчальні заняття		денна форма	заочна форма
	лекції	30 (в годинах)	10 (в годинах)
	семінари, практичні	30 (в годинах)	8 (в годинах)
Самостійна робота		120 (в годинах)	162 (в годинах)
Форма підсумкового контролю	Екзамен	-	-

Структурно-логічна схема вивчення навчальної дисципліни:

Пререквізити	Постреквізити
Управління проектами	Кваліфікаційна робота

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Перелік тем лекційного матеріалу

Тема 1. Поняття “Життєвого циклу розробки програмного забезпечення” (2 год.)

SDLC як основа для планування проекту та оцінки необхідних ресурсів. Стандартний набір заходів і результатів. Управління та контроль ходом виконання проекту.

Самостійне вивчення: обрати реальний або вигаданий проєкт та описати, як проходить його життєвий цикл відповідно до різних моделей (Waterfall, Agile, V-Model тощо).

Тема 2. Історія виникнення та ключові ідеї гнучких методологій управління (4 год.)

Основні ідеї, принципи та характеристики Agile. Різні підходи та методології Agile. Переваги використання Agile методологій управління.

Самостійне вивчення: вивчити та пояснити 12 принципів маніфесту Agile. Проаналізувати значення цих принципів для практики управління проєктами та надати приклади їх реалізації в реальних проєктах.

Тема 3. Проектні команди в Agile (4 год.)

Визначення характеристик Agile команди, розподілення ролей та обов'язків в ній. Як створити високоефективну гнучку команду. Розпізнавання потенційних конфліктних ситуацій в командній роботі та шляхи їх подолати в Agile-команді. Використання кращої комунікації та тісного залучення клієнтів для покращення продуктивності команди.

Самостійне вивчення: дослідити різні ролі в Agile команді (наприклад, Scrum Master, Product Owner, команда розробників) і описати їх основні обов'язки, а також взаємодію між ними. Дослідити ключові аспекти ефективної командної роботи в Agile, такі як самоорганізація, комунікація, взаємодія та зворотний зв'язок. Навести приклади з реальних або вигаданих проєктів, де ці аспекти були важливими для успішного виконання завдань, та проаналізувати, як ці принципи можуть бути покращені.

Тема 4. Організація процесу планування (4 год.)

Різницю між гнучким і традиційним плануванням при управлінні проєктами. Поняття швидкості команди та її вплив на планування. Інструменти та методи гнучкого планування.

Самостійне вивчення: дослідити процес планування в Scrum, зокрема, різницю між Sprint Planning та Product Backlog Refinement. Створити набір User Stories для вигаданого Agile проекту та розробити план на основі цих історій. Вони повинні визначити критерії прийнятності для кожної історії, а також здійснити оцінку складності (Story Points або інші методи).

Тема 5. Процеси управління та контролю (4 год.)

Як Agile-метрики підвищують прозорість проекту. Зіставлення показників із цілями проекту та обрання правильних показників для свого проекту. Оцінка продуктивності проекту за допомогою ефективних показників.

Самостійне вивчення: дослідити різні інструменти та методи моніторингу та контролю прогресу в Agile-проектах, такі як burndown charts, velocity charts, Kanban boards тощо.

Тема 6. Управління якістю продукту (4 год.)

Забезпечення якості за допомогою підходу Agile. Відмінності між гнучким забезпеченням якості та іншими підходами. Застосування покращення якості як до якості процесу, так і до якості продукції. Розподіл відповідальності за якість у команді Agile.

Самостійне вивчення: дослідити різні методи забезпечення якості в Agile-проектах, такі як TDD (Test-Driven Development), автоматизоване тестування, code reviews тощо. Порівняти переваги та недоліки кожного методу, а також описати, як ці методи застосовуються на різних етапах розробки.

Тема 7. Роль керівника в Agile (4 год.)

Роль менеджера проекту в Agile. Навички та методи, необхідні для керівництва Agile-проектом. Створення високоефективної команди. Різні стилі лідерства та їхній вплив на команду.

Самостійне вивчення: розробити стратегію, як керівник може підтримувати свою команду в успішному виконанні Agile-проекту. Як керівник визначає і підтримує пріоритети, як мотивує команду, а також як забезпечує взаємодію між різними ролями в команді (Scrum Master, Product Owner, розробники). Як керівник може реагувати на проблеми та зміни в процесі роботи.

Тема 8. Практичне застосування Agile-методологій (4 год.)

Проблеми з впровадженням Agile. Цінність впровадження Agile. Ключові критерії успіху, необхідні для впровадження Agile. Типові

підводні камені, з якими стикаються організації під час впровадження практик Agile.

Самостійне вивчення: розробити Kanban-дошку для організації роботи команди на прикладі реального або вигаданого проекту. Визначити етапи роботи, пріоритети завдань та оптимізувати процес.

Зміст практичних занять

Тема 1. Поняття “Життєвого циклу розробки програмного забезпечення” (2 год.)

Аналіз життєвого циклу розробки програмного забезпечення. Створення діаграми життєвого циклу розробки програмного забезпечення.

Самостійне вивчення: обрати реальний або вигаданий проєкт та описати, як проходить його життєвий цикл відповідно до різних моделей (Waterfall, Agile, V-Model тощо).

Тема 2. Історія виникнення та ключові ідеї гнучких методологій управління (4 год.)

Дослідження етапів розвитку гнучких методологій управління. Практичне застосування принципів Agile до вигаданого проєкту.

Самостійне вивчення: вивчити та пояснити 12 принципів маніфесту Agile. Проаналізувати значення цих принципів для практики управління проєктами та надати приклади їх реалізації в реальних проєктах.

Тема 3. Проектні команди в Agile (4 год.)

Створення та організація проектної команди в Agile. Оцінка ефективності Agile-команди через симуляцію проєкту.

Самостійне вивчення: дослідити різні ролі в Agile команді (наприклад, Scrum Master, Product Owner, команда розробників) і описати їх основні обов'язки, а також взаємодію між ними. Дослідити ключові аспекти ефективної командної роботи в Agile, такі як самоорганізація, комунікація, взаємодія та зворотний зв'язок.

Тема 4. Організація процесу планування (4 год.)

Розробка плану проєкту за допомогою Agile-методів. Планування спринту з використанням техніки Planning Poker. Порівняння підходів до планування в Agile та Waterfall.

Самостійне вивчення: дослідити процес планування в Scrum, зокрема, різницю між Sprint Planning та Product Backlog Refinement. Створити набір User Stories для вигаданого Agile проєкту та розробити

план на основі цих історій. Вони повинні визначити критерії прийнятності для кожної історії, а також здійснити оцінку складності (Story Points або інші методи).

Тема 5. Процеси управління та контролю (4 год.)

Розробка процесу управління проектом та контролю за виконанням. Моніторинг і контроль за виконанням проекту з використанням інструментів.

Самостійне вивчення: дослідити різні інструменти та методи моніторингу та контролю прогресу в Agile-проектах, такі як burndown charts, velocity charts, Kanban boards тощо.

Тема 6. Управління якістю продукту (4 год.)

Розробка стратегії управління якістю для проекту. Розробка функціональності з використанням TDD, Pair Programming і CI.

Самостійне вивчення: дослідити різні методи забезпечення якості в Agile-проектах, такі як TDD (Test-Driven Development), автоматизоване тестування, code reviews тощо. Порівняти переваги та недоліки кожного методу, а також описати, як ці методи застосовуються на різних етапах розробки.

Тема 7. Роль керівника в Agile (4 год.)

Роль керівника в процесі організації Agile-процесів. Керівник як фасилітатор та лідер в Agile-команді. Симуляція проведення ретроспективи в Agile-команді.

Самостійне вивчення: розробити стратегію, як керівник може підтримувати свою команду в успішному виконанні Agile-проекту. Як керівник визначає і підтримує пріоритети, як мотивує команду, а також як забезпечує взаємодію між різними ролями в команді (Scrum Master, Product Owner, розробники). Вплив керівника на проблеми та зміни в процесі роботи.

Тема 8. Практичне застосування Agile-методологій (4 год.)

Розробка стратегії успішного переходу на Agile в організації. Оцінка та покращення Agile-процесів у команді.

Самостійне вивчення: розробити Kanban-дошку для організації роботи команди на прикладі реального або вигаданого проекту. Визначити етапи роботи, пріоритети завдань та оптимізувати процес.

Зміст самостійної роботи здобувачів

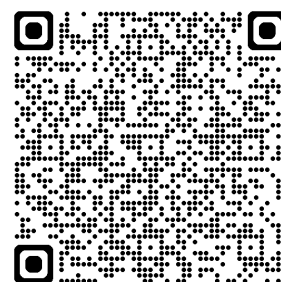
Розподіл годин, виділених на вивчення дисципліни:

Найменування видів робіт	Розподіл годин за формами навчання	
	денна	заочна
Самостійна робота, год, у т.ч.:	120	162
Опрацювання матеріалу, викладеного на лекціях	30	10
Підготовка до практичних занять та контрольних заходів	30	20
Підготовка звітів з практичних робіт	20	10
Опрацювання матеріалу, винесеного на самостійне вивчення	40	122

ПОЛІТИКА КУРСУ

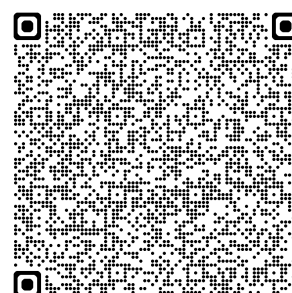
1) щодо системи поточного і підсумкового контролю

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

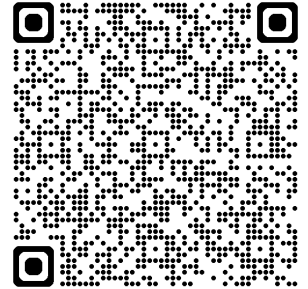
Здобувачі вищої освіти мають право на оскарження оцінки з дисципліни отриманої під час контрольних заходів. Апеляція здійснюється відповідно до «Положення



про політику та врегулювання конфліктних ситуацій». Ознайомитись з документом можна за [покликанням](#).

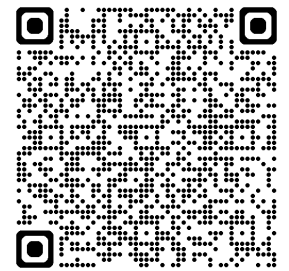
3) щодо відпрацювання пропущених занять

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](#).



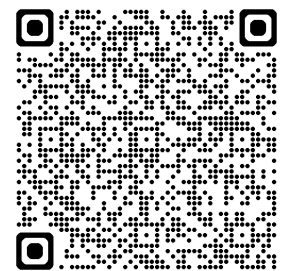
4) щодо дотримання академічної доброчесності

“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

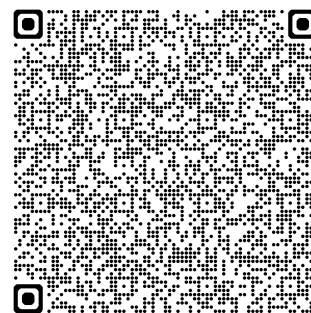
¹ визначається політика використання ШІ в навчальній дисципліні - дозволене/заборонене, правила використання

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

Програмний результат навчання	<u>Метод навчання</u>	<u>Метод оцінювання</u>
РН01 Знати і застосовувати сучасні професійні стандарти і інші нормативно-правові документи з інженерії програмного забезпечення.	Лекція, розповідь-пояснення, бесіда, комп'ютерні і мультимедійні методи, практичні роботи, дедуктивний, порівняння, проблемно-пошуковий, дослідницький, робота під керівництвом викладача, бесіда-діалог.	Екзамен, поточний контроль, усний контроль, тестовий контроль.
РН02 Оцінювати і вибирати ефективні методи і моделі розроблення, впровадження, супроводу програмного забезпечення та управління відповідними процесами на всіх етапах життєвого циклу.	Лекція, розповідь-пояснення, бесіда, комп'ютерні і мультимедійні методи, практичні роботи, дедуктивний, порівняння, проблемно-пошуковий, дослідницький, робота під керівництвом викладача, бесіда-діалог.	Екзамен, поточний контроль, усний контроль, тестовий контроль.
РН04 Виявляти інформаційні потреби і класифікувати дані для проектування програмного забезпечення.	Лекція, розповідь-пояснення, демонстрування, комп'ютерні і мультимедійні методи, індуктивний метод, інтерактивні методи (дискусія, диспут, мозковий штурм), виокремлення основного	Поточний контроль, тестовий контроль, екзамен

² визначається перелік електронних та інших ресурсів та умови перезарахування

PH06 Розробляти і оцінювати стратегії проектування програмних засобів; обґрунтовувати, аналізувати і оцінювати варіанти проектних рішень з точки зору якості кінцевого програмного продукту, ресурсних обмежень та інших факторів.	Лекція, розповідь-пояснення, бесіда, комп'ютерні і мультимедійні методи, практичні роботи, дедуктивний, порівняння, проблемно-пошуковий, дослідницький, робота під керівництвом викладача, бесіда-діалог.	Поточний контроль, усний контроль, тестовий контроль.
PH12 Приймати ефективні організаційно-управлінські рішення в умовах невизначеності та зміни вимог, порівнювати альтернативи, оцінювати ризики.	Лекція, розповідь-пояснення, бесіда, комп'ютерні і мультимедійні методи, практичні роботи, дедуктивний, порівняння, проблемно-пошуковий, дослідницький, робота під керівництвом викладача, бесіда-діалог.	Поточний контроль, усний контроль, тестовий контроль.
PH13 Конфігурувати програмне забезпечення, керувати його змінами та розробленням програмної документації на всіх етапах життєвого циклу.	Лекція, розповідь-пояснення, демонстрування, комп'ютерні і мультимедійні методи, індуктивний метод, інтерактивні методи (дискусія, диспут, мозковий штурм), виокремлення основного	Поточний контроль, усний контроль, тестовий контроль.

ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Вид	Зміст	% від загальної оцінки	Бал	
			min	max
Поточні контрольні заходи	всього	60	35	60
Підсумкові контрольні заходи	екзамен	40	24	40
Всього:	-	100	60	100

Процедура проведення контрольних заходів, а саме поточного контролю знань протягом семестру та підсумкового семестрового контролю (екзамена) – регулюється «Положенням про систему поточного та підсумкового контролю оцінювання знань та визначення рейтингу студентів», яке розміщено на сайті університету в розділі [«Публічна інформація»](#).

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

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 ІНФОРМАЦІЇ

1. Manifesto for Agile Software Development / К. Beck et al. Manifesto for Agile Software Development. URL: <http://agilemanifesto.org> (date of access: 18.12.2024).
2. Agile Alliance. URL: <https://www.agilealliance.org/> (date of access: 19.12.2024).
3. Scrum Alliance - Find Courses for Scrum and Agile Certifications. Scrum Alliance - Find Courses for Scrum and Agile Certifications. URL: <https://www.scrumalliance.org/> (date of access: 19.12.2024).
4. ГОЛОВНА | Scrum Ukraine. Scrum Ukraine. URL: <https://www.scrum.ua/> (дата звернення: 19.12.2024).
5. Schwaber K., Beedle M. Agile Software Development with SCRUM. Pearson Education, Limited, 2008. 158 p.
6. Poppendieck M., Poppendieck T. Lean Software Development: An Agile Toolkit for Software Development Managers. Addison-Wesley Professional, 2003. 240 p.
7. Sommerville I. Software Engineering: Pearson; 10th edition, 2015. 816 p.
8. KentBeck.com. URL: <https://www.kentbeck.com/> (date of access: 19.12.2024).
9. Beck K. Test Driven Development: By Example. Addison-Wesley Professional, 2002. 240 p.
10. Fowler M. Patterns of Enterprise Application Architecture. Addison-Wesley Professional, 2002. 560 p.
11. Гончарук Д. Scrum Master – які в нього посадові обов'язки та навіщо він потрібен великим компаніям. DOU: Спільнота програмістів. URL: <https://dou.ua/lenta/articles/questions-to-scrum-master/> (дата звернення: 19.12.2024).