



КОМПОНЕНТИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ. ЧАСТИНА 2. МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ. АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>121 Інженерія програмного забезпечення</i>
Освітня програма	<i>Інженерія програмного забезпечення мультимедійних та інформаційно-пошукових систем</i>
Статус дисципліни	<i>основна</i>
Форма навчання	<i>Очна (денна)</i>
Рік підготовки, семестр	<i>2 рік підготовки, 3 семестр</i>
Обсяг дисципліни	<i>Лекції: 36 год., комп'ютерний практикум: 18 год., самостійна робота: 66 год.</i>
Семестровий контроль/ контрольні заходи	<i>Залік, модульна контрольна робота, календарний контроль</i>
Розклад занять	<i>Згідно розкладу на осінній семестр поточного навчального року (http://roz.kpi.ua/)</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: к.т.н., доцент, Люшенко Л.А., LyushenkoL@gmail.com Комп'ютерний практикум: к.т.н., доцент, Люшенко Л.А., LyushenkoL@gmail.com</i>
Розміщення курсу	<i>MS Teams. Доступ зареєстрованим користувачам.</i>

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

«Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення» є складовою комплексної дисципліни "Компоненти програмної інженерії". Вивчення цієї складової дозволяє сформуванню у здобувачів освіти компетенції, необхідні для розв'язання практичних задач професійної діяльності, пов'язаної із розробленням програмного забезпечення в частині виявлення, формування, аналізування вимог до програмного забезпечення та моделювання програмних систем на цьому етапі розробки з застосування міжнародних стандартів та best practices.

Метою вивчення дисципліни *«Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення» є формування у здобувачів освіти здатності самостійно розроблювати програмне забезпечення та здійснювати наступне: виявляти та формувати вимоги та аналізувати вимоги до програмної системи; моделювати програмну систему відповідно вимогам; обирати технологію розробки*

відповідно до визначених вимог; виконувати документування вимог та моделей для подальшого розроблення програмної системи.

Предметом дисципліни «Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення» є методи, технології та процеси моделювання, виявлення, формування та аналізування вимог для створення програмного забезпечення.

Вивчення дисципліни «Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення» формує фахові компетенції відповідно до ОП, необхідні для розв'язання практичних задач професійної діяльності, пов'язаної з розробленням, впровадженням, вдосконаленням та експлуатацією програмних інформаційних систем, а саме:

ФК 01 Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.

ФК 02 Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

ФК 03 Здатність розробляти архітектури, модулі та компоненти програмних систем.

ФК 04 Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

ФК 05 Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.

ФК07 Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

ФК08 Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.

ФК10 Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя

ФК11 Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розроблення програмного забезпечення.

ФК12 Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.

ФК13 Здатність обґрунтовано обирати та освоювати інструментарій з розроблення та супроводження програмного забезпечення.

ФК 21 Здатність визначати, аналізувати та документувати вимоги до програмного забезпечення мультимедійних та інформаційно-пошукових систем

ФК 22 Здатність створювати інноваційні стартап-проекти, обчислювати основні техніко-економічні показники та розробляти бізнес-моделі інноваційних стартап-проектів програмного забезпечення мультимедійних та інформаційно-пошукових систем, які мають комерційний потенціал для інвестування.

Вивчення дисципліни «Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення» сприяє формуванню у студентів наступних **програмних результатів навчання (ПРН)** за освітньою програмою:

ПРН01 Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

ПРН02 Знати кодекс професійної етики, розуміти соціальну значимість та культурні аспекти інженерії програмного забезпечення і дотримуватись їх в професійній діяльності.

ПРН03 Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

ПРН04 Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.

ПРН06 Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

ПРН07 Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

ПРН08 Знати та вміти розробляти людино-машинний інтерфейс.

ПРН09 Вміти використовувати методи та засоби збору, формулювання та аналізу вимог до програмного забезпечення.

ПРН10 Проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування.

ПРН11 Вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання.

ПРН13 Знати і застосовувати методи розроблення алгоритмів, конструювання програмного забезпечення та структур даних і знань.

ПРН14 Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

ПРН15 Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

ПРН16 Мати навички програмного розроблення, погодження оформлення і випуску всіх видів програмної документації.

ПРН17 Вміти застосовувати методи компонентної розробки програмного забезпечення.

ПРН18 Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.

ПРН19 Знати та вміти застосовувати методи верифікації та валідації програмного забезпечення.

ПРН20 Знати підходи щодо оцінювання та забезпечення якості програмного забезпечення.

ПРН23 Вміти документувати та презентувати результати розроблення програмного забезпечення.

ПРН31 Вміти визначати, аналізувати та документувати вимоги до програмного забезпечення мультимедійних та інформаційно-пошукових систем

ПРН32 Вміти розробляти та аналізувати моделі повного циклу створення програмного забезпечення мультимедійних та інформаційно-пошукових систем.

ПРН33 Вміти організовувати повний цикл управління програмним продуктом.

ПРН34 Вміти створювати інноваційні стартап проекти проектування програмного забезпечення мультимедійних та інформаційно-пошукових систем, які мають комерційний потенціал для інвестування.

ПРН35 Вміти розробляти та аналізувати бізнес-моделі інноваційних стартап проектів розроблення програмного забезпечення мультимедійних та інформаційно-пошукових систем, які мають комерційний потенціал для інвестування.

ПРН36 Вміти управляти проектами створення та впровадження програмного забезпечення згідно міжнародних стандартів.

ПРН38 Вміти застосовувати технології програмування для розроблення програмного забезпечення мультимедійних та інформаційно-пошукових систем.

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Успішному вивченню дисципліни «Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення» передують вивчення дисциплін «Компоненти програмної інженерії. Частина 1. Вступ до програмної інженерії», «Основи програмування» навчального плану підготовки бакалаврів за спеціальністю 121 Інженерія програмного забезпечення.

Отримані при засвоєнні дисципліни «Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення» теоретичні знання та практичні уміння забезпечують успішне вивчення дисциплін «Компоненти програмної інженерії. Частина 3. Архітектура програмного забезпечення», «Компоненти програмної інженерії. Частина 4. Якість та тестування програмного забезпечення», «Стандартизація та технології розроблення мультимедійних та інформаційно-пошукових програмних продуктів», «Програмне забезпечення мультимедійних систем», виконання дипломного проектування за спеціальністю 121 Інженерія програмного забезпечення.

3. Зміст навчальної дисципліни

Дисципліна «Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення» передбачає вивчення таких тем:

Тема 1. Управління вимогами до програмного забезпечення.

Тема 2. Основи моделювання програмних системи

Модульна контрольна роботи

Залік

Базова література:

1. Розробка та аналіз вимог до програмного забезпечення. Компоненти програмної інженерії. Курсове проектування [Електронний ресурс] : навчальний посібник для студентів з спеціальності 121 «Інженерія програмного забезпечення», освітньої програми «Інженерія програмного забезпечення мультимедійних та інформаційно-пошукових систем» / Л. А. Люшенко, Я. В. Хіцко ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1,87 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2020. – 64 с.
2. Конспект лекцій з дисципліни «Аналіз вимог до програмного забезпечення» для студентів напряму підготовки 6.050103 «Програмна інженерія». Укладач: канд. фіз.-мат. наук, доцент Божуха Л.М.– Дніпродзержинськ, ДДТУ: 2015. – 94 с.
3. С. О. Цибульник, К. С. Барандич Технології розроблення програмного забезпечення частина 1. Життєвий цикл програмного забезпечення. Електронне мережне навчальне видання Київ КПІ ім. Ігоря Сікорського, 2022. – 270 с.
4. Петрик М.Р. Моделювання програмного забезпечення : науковометодичний посібник / М.Р. Петрик, О.Ю. Петрик – Тернопіль : Вид-во ТНТУ імені Івана Пулюя, 2015. – 200 с.
5. Стеценко, І.В. Моделювання систем: навч. посіб. [Електронний ресурс, текст] / І.В. Стеценко; М-во освіти і науки України, Черкас. держ. технол. ун-т. – Черкаси : ЧДТУ, 2010. – 399 с. ISBN 978-966-402-073-9

Додаткова література:

6. Введення в програму інженерію та управління життєвим циклом програмного забезпечення. Програмна інженерія. Програмні вимоги. "Software Requirements. IEEE Guide to the Software Engineering Body of Knowledge(1) - SWEBOOK®, 2004. -21 с
7. Wiegers K. Software Requirements (Developer Best Practices) / K. Wiegers. – London: Pearson, 2013. – 672 р.
8. ISO 9126 – «Інформаційна технологія. Оцінка програмного продукту. Характеристики якості керівництво по їх застосуванню». <https://studfiles.net/preview/5300003/page:4/>
9. ISO/IEC 12207:2008 System and software engineering Software life cycle processes (IDT)
10. UML специфікації. www.omg.com.
11. Ноздріна Л. В. Бізнес-аналіз як драйвер успіху ІТ-проєкту / Л. В. Ноздріна, В. М. Лисак // Матеріали III Всеукраїнської науково-практичної інтернет-конференції студентів, аспірантів та молодих вчених «Сучасні інформаційні системи та технології». Херсон : Херсонський національний технічний університет, 2020. С. 190-192.
12. В.В.Нетепчук. Управління бізнес-процесами: Навч. посібник. – Рівне: НУВГП, 2014. – 158 с.

Використати для опанування практичних умінь дисципліни. Матеріали знаходяться у вільному доступі в Інтернеті.

Навчальний контент

13. Методика опанування навчальної дисципліни (освітнього компонента)

№	Тип навчального заняття	Опис навчального заняття
Тема 1. Управління вимогами до програмного забезпечення.		
1	Лекція №1 Вступ до управління вимог.	Типи програмних інформаційних систем. Місце управління вимоги в життєвому циклі розроблення програмного забезпечення. Цикл управління вимогами. Принципи управління вимоги. Стандарти управління вимогами. ДСТУ 34.601–90 і ДСТУ 34.201–89)
2	Лекція №2 Структура вимог до програмного забезпечення	Структура вимог до програмного забезпечення. Методи виявлення вимог до програмного забезпечення. Вимоги до продукту і процесу Системні вимоги та вимоги до програмного забезпечення. Функціональні, нефункціональні вимоги і характеристики програмного продукту.
3	Комп'ютерний практикум №1	Вибір кейсу для розроблення програмного забезпечення. Опис ідеї стартапу
4	Лекція №3. Бізнес вимоги. Частина 1 Лекція №3. Бізнес вимоги. Частина 2	Визначення зацікавлених сторін. Ідентифікація зацікавлених сторін. Аналіз зацікавлених сторін. Стратегія управління. План комунікацій з зацікавленими сторонами. Вимоги до програмного забезпечення зацікавлених сторін.
5	Комп'ютерний практикум №2	Логіко-структурний аналіз. Дерево проблем

6	<p><i>Лекція №4. Бізнес вимоги. Бізнес-модель програмного стартапу. Частина 1.</i></p> <p><i>Лекція №4. Бізнес вимоги. Бізнес-модель програмного стартапу. Частина 2.</i></p>	<p><i>Модель використання програмного забезпечення (стартапу). Поняття ІТ стартапу. Формування канви стартапу. Вимоги до комерціалізації програмного стартап проекту, як основна бізнес-вимога. Бачення проекту, мета проекту, продукт проекту, результати проекту. Стратегічні альтернативи реалізації проекту.</i></p>
7	<p><i>Комп'ютерний практикум №3</i></p>	<p><i>Зацікавлені сторони програмного проекту</i></p>
8	<p><i>Лекція №6. Виявлення вимог. Методи, технології виявлення вимог</i></p>	<p><i>Методи технології виявлення вимог. Джерела вимог. Опитування. Організація. Вивчення документації. Співпраця з зацікавленими сторонами.</i></p>
9	<p><i>Лекція №7. Формування і опис бізнес-вимог в програмному стартап проекті.</i></p>	<p><i>Процес документування, аналіз бізнес вимог. Погодження вимог. Усунення протиріч. Затвердження вимог. Вимоги в технічному завданні.</i></p>
10	<p><i>Комп'ютерний практикум №4</i></p>	<p><i>Побудова бізнес моделі програмного стартап проекту</i></p>
11	<p><i>Лекція №8 Вимоги зацікавлених сторін (користувача) до програмного забезпечення</i></p>	<p><i>Вимоги зацікавлених сторін, як правило, не формалізовані, тому потребують додаткової формалізації зі сторони розробників та відповідного документування. Технології та методи формалізації, документування.</i></p>
12	<p><i>Лекція №9 Функціональні та нефункціональні вимоги</i></p>	<p><i>Визначення функціональних та нефункціональних вимог. Функціональні вимоги перелік функцій або сервісів, які повинна надавати система, а також обмежень на дані і поведінку. Нефункціональні вимоги характеризують принципи взаємодії із середовищами або іншими системами, а також визначають показники часу роботи, захисту даних і досягнення якості з урахуванням рекомендацій використовуваного стандарту. Різниця між функціональними, нефункціональними та бізнес вимогами.</i></p>
13	<p><i>Комп'ютерний практикум №5</i></p>	<p><i>Розробка та проведення опитувань зацікавлених сторін (користувачів)</i></p>
14	<p><i>Лекція № 10 Системні вимоги, атрибути якості, зовнішні системи та інтерфейси, обмеження</i></p>	<p><i>Визначення: системні вимоги, атрибути якості, вимоги до зовнішніх систем та інтерфейсів, обмеження. Особливості виявлення вимог, різниця у визначенні та аналізі.</i></p>
15	<p><i>Лекція №11 Аналізування вимог до програмного забезпечення</i></p>	<p><i>Інструменти аналізування вимог. Експертиза вимог. Тестування вимог. Критерії прийнятності. Прототипування вимог.</i></p>
16	<p><i>Комп'ютерний практикум № 6</i></p>	<p><i>Користувацькі історії. Реєстри вимог.</i></p>

17	Лекція № 12 Основи моделювання програмного забезпечення з використання UML	UML може бути застосовано на всіх етапах життєвого циклу аналізу бізнес-систем і розробки програмного забезпечення. Доречно використовувати UML діаграми для наглядного опису вимог.
18	Лекція №13. Основи моделювання поведінки програмного забезпечення (прецедент) UML	Різні види діаграм які підтримуються UML, і найбагатший набір можливостей представлення певних аспектів системи робить UML універсальним засобом опису як програмних, так і ділових систем. Для опису функціольності програмної системи доречно використовувати діаграми послідовностей.
19	Комп'ютерний практикум № 7	Специфікації вимог. Документообіг
Тема 2. Основи моделювання програмних системи		
20	Лекція 14. Основи моделювання послідовностей (сценаріїв) UML	Діаграми послідовностей UML відображає взаємодії об'єктів впорядкованих за часом. Такі діаграми доречно використовувати при описі процесів, сценаріїв тощо.
21	Лекція 15. Документування вимог	Реєстр вимог. Специфікація вимог. Склад специфікації вимог Рекомендації до формулювання вимог. Стандартні шаблони специфікації. Засоби документування: структурована природня мова, візуальна мова UML
22	Комп'ютерний практикум № 8	Моделювання вимог. Діаграми прецедентів та послідовностей
23	Лекція 16. Вимоги та тестування програмного забезпечення	Вимоги до якості. Зв'язок вимог та тестування. Доказова база реалізації вимог. Ризики вимог та тестування. Внесення змін в вимоги до програмного забезпечення
Модульна контрольна робота		
Екзамен		

5. Самостійна робота студента/аспіранта

Дисципліна «Компоненти програмної інженерії. Частина 2. Моделювання програмного забезпечення. Аналіз вимог до програмного забезпечення» ґрунтується на самостійних підготовці до аудиторних занять на теоретичні та практичні теми.

№	Назва теми, що виноситься на самостійне опрацювання	Кількість годин	Літературні джерела
Тема 1. Управління вимогами до програмного забезпечення.			
1	Лекція №1 Вступ до управління вимог.	1	1,8,9
2	Лекція №2 Структура вимог до програмного забезпечення	1	2,3,7
3	Комп'ютерний практикум №1 .Вибір кейсу для розробляння програмного забезпечення. Опис ідеї стартапу	3	1,2,3,11

4	Лекція №3. Бізнес вимоги. Частина 1	1	2,3,8
	Лекція №3. Бізнес вимоги. Частина 2	1	2,3,8
5	Комп'ютерний практикум №2. Логіко-структурний аналіз. Дерево проблем	3	1,3,7
6	Лекція №4. Бізнес вимоги. Бізнес-модель програмного стартапу. Частина 1.	1	1,3,6
	Лекція №4. Бізнес вимоги. Бізнес-модель програмного стартапу. Частина 2.	1	1,2,3
7	Комп'ютерний практикум №3. Зацікавлені сторони програмного проєкту	3	3,6
8	Лекція №6. Виявлення вимог. Методи, технології виявлення вимог	1	1,7
9	Лекція №7. Формування і опис бізнес-вимог в програмному стартап проєкті.	1	1,6,7
<i>Тема 2. Основи моделювання програмних системи</i>			
10	Комп'ютерний практикум №4. Побудова бізнес моделі програмного стартап проєкту	3	1,7
11	Лекція №8 Вимоги зацікавлених сторін (користувача) до програмного забезпечення	1	1,9
12	Лекція №9 Функціональні та нефункціональні вимоги	1	2,3
13	Комп'ютерний практикум №5. Розробка та проведення опитувань зацікавлених сторін (користувачів)	3	1,7
14	Лекція № 10 Системні вимоги, атрибути якості, зовнішні системи та інтерфейси, обмеження	1	1,7,8
15	Лекція №11 Аналізування вимог до програмного забезпечення	1	1, 3,7
16	Комп'ютерний практикум № 6. Користувацькі історії. Реєстри вимог.	3	1,7
17	Лекція № 12 Основи моделюва програмного забезпечення з використання UML	1	4,5
18	Лекція №13. Основи моделювання поведінки програмного забезпечення (прецедент) UML	1	4,12
19	Комп'ютерний практикум № 7. Специфікації вимог. Документообіг вимог	3	1,7
20	Лекція 14. Основи моделювання послідовностей (сценаріїв) UML	1	5,10,12
21	Лекція 15. Документування вимог	1	1, 7,10,11
22	Комп'ютерний практикум № 8. Моделювання вимог. Діаграми прецедентів та послідовностей	3	4,5,10,12
23	Лекція 16. Вимоги та тестування програмного забезпечення	1	1,7,8

Модульна контрольна робота	10	1-12
Залік	15	1-12

Політика та контроль

6. Політика навчальної дисципліни (освітнього компонента)

Відвідування лекційних занять є обов'язковим.

Відвідування занять комп'ютерного практикуму може бути епізодичним та за потреби консультації/захисту робіт комп'ютерного практикуму.

Правила поведінки на заняттях: активність, повага до присутніх, відключення телефонів.

Дотримання політики академічної доброчесності.

Правила захисту робіт комп'ютерного практикуму: роботи повинні бути зроблені відповідно до поставлених задач та згідно з варіантом.

7. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Протягом семестру студенти виконують 8 комп'ютерних практикумів. Максимальна кількість балів за кожний комп'ютерний практикум: 10 балів. Виконання всіх комп'ютерних практикумів є обов'язковими до виконання для отримання заліку.

Бали нараховуються за:

- якість виконання комп'ютерного практикуму: 0-5 бали;
- відповідь під час захисту комп'ютерного практикуму: 0-3 бали;
- своєчасне представлення роботи до захисту: 0-2 бали.

Критерії оцінювання якості виконання:

- 5 бали – робота виконана якісно, в повному обсязі;
- 4 бали – робота виконана якісно, в повному обсязі, але має недоліки;
- 3 бали – робота виконана в повному обсязі, але містить незначні помилки;
- 2 бали – робота виконана в повному обсязі, але містить суттєві помилки;
- 0 балів – робота виконана не в повному обсязі.

Критерії оцінювання відповіді:

- 3 бали – відповідь повна, добре аргументована;
- 2 бали – відповідь вірна, але має недоліки або незначні помилки;
- 1 бал – у відповіді є суттєві помилки;
- 0 балів – немає відповіді або відповідь невірна.

Критерії оцінювання своєчасності представлення роботи до захисту:

- 2 бали – робота представлена до захисту не пізніше вказаного терміну;
- 0 балів – робота представлена до захисту пізніше вказаного терміну.

Максимальна кількість балів за виконання та захист комп'ютерних практикумів:

10 балів × 8 комп. практ. = 80 балів.

Протягом семестру на лекціях відбуваються **опитування за темою поточного заняття**.

Максимальна кількість балів за всі опитування: 3 бали. Кількість **опитування за темою поточного заняття** для одного студента є необмеженою.

Завдання на **модульну контрольну роботу** складається з 1 теоретичне та 1 практичне запитань. Відповідь на кожне запитання оцінюється 10 балами.

Критерії оцінювання кожного запитання контрольної роботи:

- 9-10 балів – відповідь вірна, повна, добре аргументована;
- 7-8 балів – відповідь вірна, розгорнута, але не дуже добре аргументована;

5-6 балів – в цілому відповідь вірна, але має недоліки;
3-4 балів – у відповіді є незначні помилки;
1-2 бали – у відповіді є суттєві помилки;
0 балів – немає відповіді або відповідь невірна.

Максимальна кількість балів за модульну контрольну роботу:
10 балів × 2 запитань = 20 балів.

Рейтингова шкала з дисципліни дорівнює:
 $R = R_c = 80 \text{ балів} + 20 \text{ балів} = 100 \text{ балів}$.

Календарний контроль: проводиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.

На першій атестації (8-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 20 балів (50 % від максимальної кількості балів, яку може отримати студент до першої атестації).

На другій атестації (14-й тиждень) студент отримує «зараховано», якщо його поточний рейтинг не менше 40 балів (50 % від максимальної кількості балів, яку може отримати студент до другої атестації).

Семестровий контроль: залік

Умови допуску до семестрового контролю:

При семестровому рейтингу (R_c) не менше 60 балів та зарахуванні усіх робіт комп'ютерного практикуму, студент отримує залік «автоматом» відповідно до таблиці (Таблиця відповідності рейтингових балів оцінкам за університетською шкалою). В іншому разі він має виконувати залікову контрольну роботу.

Необхідною умовою допуску до залікової контрольної роботи є виконання і захист комп'ютерного практикуму.

Якщо студент не погоджується з оцінкою «автоматом», то може спробувати підвищити свою оцінку шляхом написання залікової контрольної роботи, при цьому його бали, отримані за семестр, зберігаються, а з двох отриманих студентом оцінок виставляється краща («м'яка» система оцінювання).

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

8. Додаткова інформація з дисципліни (освітнього компонента)

Перелік питань, які виносяться на семестровий контроль, наведено у Додатку 1.

Робочу програму навчальної дисципліни (силабус):

Складено к.т.н., доцент, Люшенко Л.А.

Ухвалено кафедрою ПЗКС (протокол № 12 від 26.04.2023 р.)

Погоджено Методичною комісією факультету прикладної математики (протокол № 10 від 26.05.2023 р.)

Додаток 1. Перелік питань, які виносяться на семестровий контроль

1. Вплив зовнішнього середовища на створення сучасних інформаційних систем
2. Особливості інтерпретації вимог . Рівні вимог
3. Бізнес-модель програмного продукту
4. Сегменти споживачів. Найважливіші клієнти. На які сегменти їх можна поділити.
5. Ціннісні пропозиції.
6. Канали.
7. Взаємовідносини з клієнтами
8. Ключові ресурси
9. Ключові дії.
10. Ключові партнери. Зацікавлені сторони.
11. Розробка та управління вимогами. «Розростання» вимог користувачів
12. Двозначність вимог
13. Мінімальна специфікація
14. Вигоди від високоякісного процесу розробки вимог
15. Характеристики вимог
16. Характеристики окремих положень специфікації вимог
17. Характеристики специфікації вимог
18. Співпраця клієнтів і розробників
19. Виявлення вимог. Аналіз вимог
20. Специфікації вимог. Перевірка вимог.
21. Процес створення вимог
22. Визначення способу продукту аж до бізнес-вимог
23. Конфліктуючі бізнес-вимоги
24. Бізнес-вимоги і варіанти використання
25. Масштаби і обмеження проекту
26. Контекстна діаграма
27. Основні джерела отримання інформації про потреби клієнтів
28. Виявлення вимог. Як зрозуміти, що збір вимог завершено
29. Підхід із застосуванням варіанти використання продукту
30. Варіанти використання і сценарії використання
31. Визначення варіантів використання
32. Документування варіантів використання
33. Варіанти використання продукту і функціональні вимоги
34. Переваги способу із застосуванням варіантів використання
35. Яких пасток слід побоюватися при способі із застосуванням
36. Визначення понять модель, моделювання , адекватність моделі
37. Життєвий цикл ІС. Формування життєвого циклу ІС
38. Визначення вимог до ІС. Етап аналізу вимог
39. Етап проектування. Особливості проектування інформаційних систем .

40. Стадія реалізації ІС передбачає розробку і тестування компонентів і комплексне тестування системи
41. Місце моделювання в процесі створення інформаційної системи . Стадії побудови моделі інформаційної системи.