



МЕТОДОЛОГІЯ ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни	
Рівень вищої освіти	Другий (магістерський)
Галузь знань	12 Інформаційні технології
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці
Статус дисципліни	Обов'язкова (нормативна)
Форма навчання	Очна (денна)
Рік підготовки, семестр	I курс, другий (весінній) семестр
Обсяг дисципліни	120 годин / 4 кредити ЄКТС, з яких 54 години аудиторних (36 годин лекції, 18 годин лабораторні) та 66 годин становить самостійна робота
Семестровий контроль / контрольні заходи	Залік
Розклад занять	http://rozklad.kpi.ua/
Мова викладання	Українська
Інформація про керівника курсу / викладачів	<u>Лектор:</u> к.т.н., доцент Шпурик Вадим Вадимович <u>Лабораторні роботи:</u> к.т.н., доцент Шпурик Вадим Вадимович
Розміщення курсу	http://route.ho.ua/kpi/se/

Програма навчальної дисципліни

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

Студентам спеціальності Інженерія програмного забезпечення варто вивчати дисципліну «Методологія інженерії програмного забезпечення» тому, що вона дозволяє набути навичок і отримати вміння з розробки програмного забезпечення, що дозволять покращити та розширити знання, набуті студентами за попередні роки навчання, вдосконалити навички та здібності розробки, а також, напрацювати навик і досягти умінь в розробці програмного забезпечення.

Метою опанування дисципліни є отримання студентами ґрунтовної підготовки з теоретичних і практичних основ у методологія інженерії програмного забезпечення: вмінь проектувати, розробляти та тестувати програмне забезпечення; отримання знань об'єктно-орієнтованого підходу до розробки, набуття здібностей із застосування каркасів, прикладів та шаблонів проектування; отримання навичок командної роботи та вмінь аналізувати вимоги у вирішеннях прикладних і наукових завдань в області інформаційних систем і технологій.

Предметом дисципліни є методи та підходи, що використовуються при командній роботі, аналізі вимог, проектуванні, реалізації, тестуванні, впровадженні та експлуатації інформаційних систем та

технологій, систем обробки інформації на базі сучасних технологій обробки. Розглядаються основні етапи процесу розробки програмного продукту, включаючи його опис, проектування, власне розробку, інтеграцію і тестування частин, тестування програми в цілому і супровід. Кожен з етапів докладно ілюструється на конкретних прикладах. Окремі розділи присвячені визначенню типу процесу розробки та управління проектом. В тому числі методологія DevOps: принципи побудови та захисту DevOps-конвеєра. Знайомство з DevOps, його впливом на створення хмарних сервісів. Застосування безперервної інтеграції, безперервного постачання та інфраструктури як сервісу. Оцінка ролі та цілей безпеки у культурі DevOps. Визначення компонентів безпеки DevOps. Матеріал супроводжується практичними завданнями для закріплення теоретичних знань.

Програмні результати навчання

Дисципліна сприяє формуванню у студентів певних компетентностей, тобто здатностей до виконання певних завдань самостійно. Компетентності формуються як фахові, так і загальні.

Загальні компетентності:

- здатність до абстрактного мислення, аналізу та синтезу (ЗК-1);
- здатність проводити дослідження на відповідному рівні (ЗК-3);
- здатність спілкуватися з представниками інших професійних груп різного рівня (з експертами інших галузей знань/видів економічної діяльності) (ЗК-4);
- здатність генерувати нові ідеї (креативність) (ЗК-5).

Фахові компетентності:

- здатність ефективно керувати фінансовими, людськими, технічними та іншими проектними ресурсами у сфері інженерії програмного забезпечення (ФК-6);
- здатність забезпечувати якість програмного забезпечення (ФК-9);
- здатність планувати і виконувати наукові дослідження з інженерії програмного забезпечення (ФК-10);
- здатність застосовувати і розвивати фундаментальні і міждисциплінарні знання для успішного розв'язання наукових проблем інженерії програмного забезпечення (ФК-11).

Згідно з вимогами програми навчальної дисципліни після засвоєння навчальної дисципліни студенти мають продемонструвати певні **програмні результати навчання**, тобто набуті ними знання й уміння описані далі:

- 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 (ПРН-6);
- конфігурувати програмне забезпечення, керувати його змінами та розробленням програмної документації на всіх етапах життєвого циклу (ПРН-13);
- прогнозувати розвиток програмних систем та інформаційних технологій (ПРН-14);
- здійснювати реінжиніринг програмного забезпечення відповідно до вимог замовника (ПРН-15);
- збирати, аналізувати, оцінювати необхідну для розв'язання наукових і прикладних задач інформацію, використовуючи науково-технічну літературу, бази даних та інші джерела (ПРН-17);
- розробляти математичне і програмне забезпечення для наукових досліджень в галузі інженерії програмного забезпечення (ПРН-18);
- планувати і виконувати наукові дослідження в сфері інженерії програмного забезпечення, обирати методики та інструменти, аналізувати результати, обґрунтовувати висновки (ПРН-20).

2. Пререквізити та постреквізити дисципліни

Дисципліна «Методологія інженерії програмного забезпечення» розміщується в структурно-логічній схемі після таких нормативних освітніх компонентів циклу загальної та професійної підготовки, як: «Розробка застосунку Інтернету речей та сенсорних мереж в енергетиці», «Інтелектуальний аналіз даних для задач енергетики», «Хмарні GRID-технології».

Навчальна дисципліна забезпечує вивчення та засвоєння освітнього компонента «Методологія інженерії програмного забезпечення. Курсовий проект», і вивчення дисципліни «Бізнес-аналіз в ІТ». Також, засвоєні знання та навички можуть бути використані для виконання магістерської дисертації та при проходженні науково-дослідної практики.

3. Зміст навчальної дисципліни

Навчальна дисципліна «Методологія інженерії програмного забезпечення» складається з одного кредитного модуля, який поділяється за розділами та темами, що описані у таблиці далі.

Розділ 1. Введення в методологію інженерії програмного забезпечення	Тема – 1. Етапи розробки програмного забезпечення
	Тема – 2. Процес розробки програмного забезпечення
Розділ 2. Методологія DevOps: DevOps підхід, як основа сучасної розробки програмного забезпечення	Тема – 1. Неперервна інтеграція (Continuous integration)
	Тема – 2. Інфраструктура (Infrastructure)
	Тема – 3. Неперервна доставка (Continuous delivery)
Розділ 3. Неперервна інтеграція	Тема – 1. Тести програмного забезпечення (Software tests)
	Тема – 2. Автоматичне виконання тестів (Automatic execution of tests)
Розділ 4. Масштабування програмних систем	Тема – 1. Балансування навантаження. Кешування. CDN
	Тема – 2. Stateful/stateless architecture. Sharding. Черга повідомлень

Структура кредитного модуля з розподілом годин по видах навантаження виглядає наступним чином

Назви розділів і тем	Кількість годин			
	Всього	Лекції	Лабораторні	СРС
Розділ 1. Введення в методологію інженерії програмного забезпечення				
Тема – 1. Етапи розробки програмного забезпечення	10	4	2	4
Тема – 2. Процес розробки програмного забезпечення	14	4	2	8
Разом за розділом 1	24	8	4	12
Розділ 2. Методологія DevOps: DevOps підхід, як основа сучасної розробки програмного забезпечення				
Тема – 1. Неперервна інтеграція (Continuous integration)	12	4	2	6
Тема – 2. Інфраструктура (Infrastructure)	14	4	2	8
Тема – 3. Неперервна доставка (Continuous delivery)	14	4	2	8
Разом за розділом 2	40	12	6	22
Розділ 3. Неперервна інтеграція				
Тема – 1. Тести програмного забезпечення (Software tests)	10	4	-	6
Тема – 2. Автоматичне виконання тестів (Automatic execution of tests)	10	4	2	4
Разом за розділом 3	20	8	2	10
Розділ 4. Масштабування програмних систем				
Тема – 1. Балансування навантаження. Кешування. CDN	16	4	2	10
Тема – 2. Stateful/stateless architecture. Sharding. Черга повідомлень	14	4	2	8
Разом за розділом 4	30	8	4	18
МКР	6	-	2	4
Всього годин	120	36	18	66

4. Навчальні матеріали та ресурси

Для кращого засвоєння дисципліни «Методологія інженерії програмного забезпечення» та виконання робіт на лабораторних роботах і самостійної роботи студентам надаються список рекомендованої літератури та навчальні матеріали, розміщені на університетських ресурсах.

Базові ресурси та література:

1. Software Engineering: Architecture-driven Software Development by Richard F Schmidt.
2. Clean Code - A Handbook of Agile Software Craftsmanship by Robert C. Martin.
3. Design Patterns: Elements of Reusable Object-Oriented Software by Erich Gamma, Richard Helm, Ralph Johnson, John M. Vlissides.
4. Software Engineering with UML by Bhuvan Unhelkar
5. The Essence of Software Engineering by Volker Gruhn, Rüdiger Striemer
6. Anderson, David: Kanban: Successful Evolutionary Change in Your Technology Business: Blue Hole Press, 2010 — ISBN 0984521402
7. Basili, R.; Turner, J.: Iterative Enhancement: A Practical Technique for Software Development. In: IEEE Transactions on Software Engineering Vol. 1 (1975), pp. 390–396.
8. Beck, Kent; Beedle, Mike; Bennekum, Arie van; Cockburn, Alistair; Cunningham, Ward; Fowler, Martin; Grenning, James; Highsmith, Jim; et al.: Manifesto for Agile Software Development.
9. IEEE: IEEE Standard Glossary of Software Engineering Terminology: IEEE; IEEE, 1990.

Додаткова література:

1. Think Like a Programmer: An Introduction to Creative Problem Solving by V. Anton Spraul
2. Coding Theory - Algorithms, Architectures, and Applications by Andre Neubauer, Jurgen Freudenberger, Volker Kuhn
3. The Art of Computer Programming: Volume 3: Sorting and Searching by Donald E. Knuth
4. The Algorithm Design Manual by Steven S. Skiena
5. Beyer, Betsy; Jones, Chris; Petoff, Jennifer; Murphy, Niall Richard: Site Reliability Engineering: How Google Runs Production Systems
6. Poppendieck, Mary; Poppendieck, Tom: Lean Software Development: An Agile Toolkit. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2003 — ISBN 0321150783

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Навчальна дисципліна складається з одного кредитного модуля, який охоплює 4 кредити або 120 годин. За порядком слідування та розподілом годин кредитний модуль розміщується на першому курсі другого (магістерського) вищого рівня освіти в другому (весінньому) семестрі.

Дисципліна включає в себе проведення лекційних занять – 36 годин в семестрі та лабораторних робіт – 18 годин на семестр. Також в курс дисципліни включено 66 годин протягом семестру, що відводяться на самостійну роботу студентів.

Календарний план запланованих заходів: лекційні заняття

Види робіт за розділами та темами		Кількість ауд.годин
Розділ 1. Введення в методологію інженерії програмного забезпечення		
Тема – 1. Етапи розробки програмного забезпечення	Лекція 1. Розробка програмного забезпечення: процес, способи та засоби	2
	Лекція 2. Життєвий цикл розробки програмного забезпечення: поняття, опис, що включає, як документується, як контролюється	2

Види робіт за розділами та темами			Кількість ауд.годин
Тема – 2. Процес розробки програмного забезпечення	Лекція 3. Процес розробки програмного забезпечення: вимоги, різновиди та якість. Типи ролей розробки програмного забезпечення		2
	Лекція 4. Список моделей розробки програмного забезпечення, які повинні знати всі інженери		2
Розділ 2. Методологія DevOps: DevOps підхід, як основа сучасної розробки програмного забезпечення			
Тема – 1. Неперервна інтеграція (Continuous integration)	Лекція 5. Що таке DevOps, для чого і як застосовують		2
	Лекція 6. Підхід неперервної інтеграції		2
Тема – 2. Інфраструктура (Infrastructure)	Лекція 7. CI/CD як інструменти інфраструктури		2
	Лекція 8. IaaS як інструмент інфраструктури		2
Тема – 3. Неперервна доставка (Continuous delivery)	Лекція 9. Підхід неперервної доставки: поняття, засоби, призначення		2
	Лекція 10. Підхід неперервної доставки: способи реалізації		2
Розділ 3. Неперервна інтеграція			
Тема – 1. Тести програмного забезпечення (Software tests)	Лекція 11. Тести програмного забезпечення: поняття, види, причини застосування		2
	Лекція 12. Тести програмного забезпечення: приклади реалізації		2
Тема – 2. Автоматичне виконання тестів (Automatic execution of tests)	Лекція 13. Автоматизація тестів: цілі, як виконувати, поширені помилки		2
	Лекція 14. Тести програмного забезпечення та їх автоматичне виконання		2
Розділ 4. Масштабування програмних систем			
Тема – 1. Балансування навантаження. Кешування. CDN	Лекція 15. Балансування навантаження. Кешування		2
	Лекція 16. CDN		2
Тема – 2. Stateful/stateless architecture. Sharding. Черга повідомлень	Лекція 17. Stateful/stateless architecture		2
	Лекція 18. Sharding. Черга повідомлень		2

Календарний план запланованих заходів: лабораторні роботи

№	Види робіт за розділами та темами	Кількість ауд.годин
1	Лабораторна робота 1. Створення навчальних проєктів для роботи в командах	4
2	Лабораторна робота 2. Створення план-графіка управління програмним проєктом	4
3	Лабораторна робота 3. DevOps підхід до розробки архітектури програмного забезпечення. Робота над командними проєктами	4
4	Лабораторна робота 4. Застосування CI/CD/IaaS до навчальних проєктів	4
5	Лабораторна робота 5. Застосування CI/CD/IaaS до навчальних проєктів. Продовження	4
6	Лабораторна робота 6. Розробка та реалізація ручних та автоматизованих тестів до командних програмних проєктів	8
7	Лабораторна робота 7. Масштабування готових командних проєктів	4
8	Лабораторна робота 8. Масштабування командних проєктів. Продовження	4

6. Самостійна робота студента

Самостійна робота студентів навчальної дисципліни поділена на завдання відповідно до розділів і тем. Самостійна робота складається з завдань, які студенти мають виконати самостійно в поза аудиторні години навчання. Завдання містять як теоретичні, так і практичні задачі та

розташовані в календарному плані дисципліни так, щоб доповнювати лекційний матеріал або передувати виконанню лабораторних робіт.

У таблиці далі наведений перелік завдань за термінами їх виконання протягом семестру.

Календарний план запланованої самостійної роботи студентів

№	Види робіт за розділами та темами	Кількість ауд.годин
1	СРС 1. Існуючі засоби організації спілкування в команді під час роботи над проектом Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	2
2	СРС 2. Порівняльний аналіз існуючих засобів спілкування при командній роботі, вибір кращого рішення для роботи над командними проектами Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	2
3	СРС 3. Складові частини програмного продукту Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	4
4	СРС 4.Arteфакти програмних продуктів Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	4
5	СРС 5. Огляд застосунків зі змішаною архітектурою Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	4
6	СРС 6. Причини виникнення DevOps підходу Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	2
7	СРС 7. Порівняльний аналіз існуючих готових рішень CI (Continuous integration) Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	4
8	СРС 8. Порівняльний аналіз існуючих готових рішень CD (Continuous delivery) Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	4
9	СРС 9. Continuous deployment: поняття та випадки застосування Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	4
10	СРС 10. Порівняльний аналіз існуючих готових рішень CD Continuous deployment Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	4
11	СРС 11. Швидке прототипування: основні поняття Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	3
12	СРС 12. Дослідження здійсненності: способи, методи, засоби реалізації Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	3
13	СРС 13. Контроль якості вимог до програмного забезпечення Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	2
14	СРС 14. Роль контролю якості в інспектування аналізу вимог Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	2
15	СРС 15. Поняття CDN: переваги та недоліки Рекомендована література: [Л1 с. 1-376], [Л5 с. 1-251]	4
16	СРС 16. Порівняльний аналіз існуючих рішень CDN, їх переваги та недоліки Рекомендована література: [Л1 с. 1-376], [Л2 с. 1-251]	6
17	СРС 17. Існуючі стандарти для реалізації черги повідомлень Рекомендована література: [Л1 с. 1-376], [Л2 с. 1-251]	8
18	Модульна контрольна робота	4

Політика та контроль

7. Політика навчальної дисципліни

У навчальній дисципліні «Методологія інженерії програмного забезпечення» протягом семестру передбачено проведення: лекційних занять і лабораторних робіт.

Відвідування занять

Відвідування усіх видів аудиторних занять, а саме: лекційних занять і лабораторних робіт є обов'язковим.

Виняток становить лише відсутність на заняттях з поважних причин (тобто хвороби, форм-мажорні обставини, участь в олімпіадах, тощо), про що студент має повідомити викладача та надати підтвердження причини. У разі відсутності студента на аудиторному занятті з поважних причин нарахування штрафних балів не здійснюється.

Правила захисту лабораторних робіт

Протягом семестру студенти мають виконувати і захищати лабораторні роботи на заняттях. Для кожної з робіт встановлені граничні терміни захисту (дедлайни).

Для отримання максимальних балів за кожну з передбачених програмою робіт студент повинен виконати роботу в повній мірі, без помилок і вчасно її захистити.

Якщо студент пропустив термін захисту роботи з поважної причини, він має можливість виконати та здати роботу без втрати балів.

Якщо студент пропустив термін захисту роботи без поважних причин – йому надається можливість відпрацювання пропущеної роботи шляхом написання самостійної роботи за відповідною темою. Але в такому випадку максимальний бал за роботу становитиме 75% від найвищої оцінки за завдання.

Причиною, що унеможлиблює захист пропущених робіт з лабораторних робіт є лише закінчення навчального процесу (семестру).

Політика дедлайнів та перескладань

На кожну лабораторну роботу за календарним планом виділяється по два тижні для виконання та захисту роботи. Всі терміни оголошуються студентам на початку кожного семестру.

Перескладання лабораторних робіт не передбачено.

В кінці семестру передбачено перескладання модульної контрольної роботи.

8. Види контролю та рейтингова система оцінювання результатів навчання

У навчальній дисципліні передбачені наступні види контролю:

- проведення поточного контролю;
- проведення календарного контролю;
- проведення семестрового контролю.

Поточний контроль

Протягом семестру проводиться поточний контроль у вигляді експрес-опитування та модульної контрольної роботи.

Поточний контроль має на меті перевірку знань та вмінь студентів, засвоєних та здобутих ними при прослуховуванні лекційних занять, виконанні лабораторних і самостійних робіт.

Експрес-опитування проводяться за матеріалами лекційних занять і складаються з коротких теоретичних запитань.

Модульна контрольна робота проводиться за матеріалами усіх аудиторних видів занять, тобто: лекційних занять і лабораторних робіт. Завдання складаються з теоретичних і практичних запитань. Контрольна робота проводиться в кінці семестру. Питання та опорні теми для повторення оголошуються студентам заздалегідь. Для проведення модульної контрольної роботи виділяються дві учбові години за рахунок лабораторної роботи.

Календарний контроль

Календарний контроль проводиться двічі на семестр як моніторинг поточного стану знань та вмінь студентів і виконання ними умов силабусу (робочої програми) дисципліни.

Умовою отримання атестації (позитивної оцінки) за проходження календарного контролю є виконання вимог за кількістю та якістю виконаних лабораторних робіт.

Для отримання позитивної оцінки за перший календарний контроль студент повинен виконати з першої по третю лабораторні роботи та набрати мінімально 12 балів, для отримання позитивної оцінки за другий календарний контроль студент потрібен виконати з четвертої по сьому лабораторні роботи та набрати загалом не менше 30 балів.

Причинами, що призводять до отримання незадовільного результату атестації під час календарного контролю є отримання незадовільної оцінки за виконання та захист передбачених календарним планом лабораторних робіт. Незадовільною оцінкою вважається набрані бали у кількості менше 60% можливих.

Семестровий контроль

Семестровий контроль здійснюється у вигляді заліку.

Умовами допуску до складання заліку є виконання всіх завдань, передбачених лабораторними роботами, а також, написання модульної контрольної роботи на оцінку не нижче оцінки «достатньо» (60% від максимального балу).

Загальний рейтинг студента за семестр для допуску до заліку має становити не менше 60 балів.

Система оцінювання результатів навчання

Оцінювання результатів навчання здійснюється згідно рейтингової системи оцінювання за 100-бальною шкалою.

Рейтинг студента складається з балів, які він отримав за виконання та захист лабораторних робіт і написання модульної контрольної роботи.

Також, під час роботи на лекційних заняттях і лабораторних робіт кожен студент може отримати заохочувальні бали або додаткові бали за зразкове виконання експрес-опитування. За невиконання умов навчання передбачені штрафні бали.

Кількість заохочувальних або штрафних балів рівна 5 балам на семестр.

Оцінювання лабораторних робіт

Календарним планом з дисципліни передбачено проведення 8 лабораторних робіт. Оцінюються роботи по 6 балів кожна за наступними критеріями:

- бездоганна робота оцінюється в 6 балів – завдання виконано в повній мірі, вирішує всі поставлені задачі, виконано без помилок або є лише незначні недоліки; студент відповідає на усі питання точно та без помилок або з незначними помилками ;
- є певні недоліки в виконанні роботи – 5-4 балів – завдання виконано неповністю, допущені суттєві помилки; студент відповідає на питання, але допускається незначних помилок;
- є суттєві недоліки у виконанні роботи – 3-1 бал – завдання виконано неповністю, не в повній мірі, з суттєвими помилками в програмі; студент не відповідає на всі питання або робить суттєві помилки;
- робота не виконана, не захищена або не зданий звіт – 0 балів – робота виконана менш, ніж на 60% або здана пізніше дедлайну; робота здана лише у вигляді звіту, але не захищена особисто; робота здана особисто, але не має звіту.

Оцінювання модульної контрольної роботи

Контрольна робота складається з чотирьох завдань: двох теоретичних питань і двох практичних завдань, загальною кількістю в 40 балів. Оцінювання роботи здійснюється за таким розподілом балів за завдання: теоретичні питання оцінюються по 5 балів, а практичні завдання по 15 балів кожне.

Оцінюються питання модульної контрольної роботи за такими критеріями:

- «відмінно», повна відповідь, не менше 90% потрібної інформації, відповіді написані точними формулюваннями та розгорнуто, вирішене основне завдання, дотримано стандартів розробки програмного забезпечення;
- «добре», достатньо повна відповідь, не менше 75% потрібної інформації, що виконана згідно з вимогами, є незначні неточності, відповіді короткі, але точні;
- «задовільно», неповна відповідь, не менше 60% потрібної інформації, завдання виконане з певними недоліками, наявні помилки, відповіді неповні;
- «незадовільно», відповідь не відповідає умовам до «достатньо», наявні суттєві помилки, відповідь відсутня або не вирішує поставлені задачі.

Оцінювання результатів навчання з семестровим контролем у вигляді заліку

Рейтингова оцінка складається з балів, отриманих студентом за результатами заходів поточного контролю, заохочувальних і штрафних балів.

Студенти, які виконали всі умови допуску до заліку та мають рейтингову оцінку 60 і більше балів, отримують відповідну до набраного рейтингу оцінку без додаткових випробувань.

Зі студентами, які виконали всі умови допуску до заліку, але мають рейтингову оцінку менше 60 балів, а також з тими студентами, хто бажає підвищити свою поточну рейтингову оцінку проводиться семестровий контроль у вигляді залікової контрольної роботи.

Після виконання залікової роботи, якщо оцінка за залікову роботу більша, ніж за рейтингом, здобувач отримує оцінку за результатами залікової контрольної роботи. Оцінювання залікової роботи здійснюється за критеріями аналогічними оцінюванню модульної контрольної роботи (з пропорційними балами):

– «відмінно», повна відповідь, не менше 90% потрібної інформації, що виконана згідно з вимогами до рівня вмінь;

– «добре», достатньо повна відповідь, не менше 75% потрібної інформації, що виконана згідно з вимогами до рівня вмінь або є незначні неточності;

– «задовільно», неповна відповідь, не менше 60% потрібної інформації, завдання виконане з певними недоліками;

– «незадовільно», відповідь не відповідає умовам до «задовільно» – 0 балів.

Максимальний бал, який може отримати студент за залікову роботу – 100 балів. Оцінка не додається до попереднього рейтингу.

Умовою допуску до заліку є зарахування всіх лабораторних робіт і написання модульної контрольної роботи, з загальним рейтингом не менше 60 балів.

Відповідність рейтингових балів оцінкам за університетською шкалою:

Бали	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

Робочу програму навчальної дисципліни (силабус):

Складено:

доцент, к.т.н., Шпурик Вадим Вадимович,
асистент кафедри ІПЗЕ Оленева Ксенія Миколаївна

Ухвалено кафедрою інженерії програмного забезпечення в енергетиці НН ІАТЕ (протокол № 28 від 15.05.2023 р.)

Погоджено Методичною комісією НН ІАТЕ (протокол № 9 від 26.05.2023 р.)