



Управління IT інфраструктурними проєктами

Робоча програма навчальної дисципліни (Силабус)

– Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	12 Інформаційні технології
Спеціальність	123 Комп'ютерна інженерія
Освітня програма	Комп'ютерна інженерія
Статус дисципліни	Вибіркова
Форма навчання	очна(денна) / заочна
Рік підготовки, семестр	3 курс, весняний
Обсяг дисципліни	4 кредитів/120 год. Денна форма: лекцій 36 годин, Лаб. роб. 18 год., СРС 66 год. Заочна форма: лекцій 36 год., лаб. роб. 18 год., СРС 104 год.
Семестровий контроль/ контрольні заходи	Залік
Розклад занять	Згідно розкладу на весняний семестр поточного навчального року за адресою http://rozklad.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектори: професор кафедри ОТ, д.т.н., доцент Клименко Ірина Анатоліївна ikliryna@gmail.com , доцент кафедри ОТ, д.т.н., доцент Ткаченко Валентина Василівна, tkavalivas@gmail.com Лабораторні: Каплунов Артем Володимирович, art.kaplunov@gmail.com , Гайдай Анатолій Русланович tolya.hei@gmail.com
Розміщення курсу	Навчальний курс розміщений на платформі дистанційного навчання «Сікорський» в середовищі Google Workspace for Education: https://classroom.google.com/c/NTgwNDU1NTQ3ODgz?cjc=qzdv3d4

Програма навчальної дисципліни

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

Мета дисципліни «Управління IT інфраструктурними проєктами» – набуття теоретичних знань і практичних навичок управління IT проєктами в галузі розроблення вбудованих систем, а також навичок командної роботи: набуття Soft skills шляхом застосування своїх Hard skills для розв'язку задач практики. Ціллю вивчення дисципліни є освоєння сучасних практик провідних IT-підприємств згідно зі світовими стандартами.

Вивчення дисципліни реалізується у формі організації і реалізації стартап-проєкту в сфері розроблення вбудованих систем. Результатом опрацювання завдань дисципліни є завершений практичний проєкт. Зміст дисципліни враховує і передбачає залучення менторів – практиків провідних IT компаній.

Отриманні знання і навички під час вивчення дисципліни забезпечують досконале розуміння сучасних процесів командної розробки складних програмно-апаратних систем і комплексів на рівні передових національних і світових практик, а також забезпечують необхідний досвід та навички для участі в таких проектах і їх реалізації в майбутньому.

Вивчення дисципліни підсилює наступні загальні та фахові компетенції:

- ЗК2. Здатність вчитися і оволодівати сучасними знаннями
- ФК1. Здатність застосовувати законодавчу та нормативно правову базу, а також державні та міжнародні вимоги, практики і стандарти з метою здійснення професійної діяльності в галузі комп'ютерної інженерії.
- ФК2 Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.
- ФК5. Здатність використовувати засоби і системи автоматизації проектування до розроблення компонентів комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем тощо.
- ФК9 Здатність системно адмініструвати, використовувати, адаптувати та експлуатувати наявні інформаційні технології та системи.
- ФК13. Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій.
- ФК14. Здатність проектувати системи та їхні компоненти з урахуванням усіх аспектів їх життєвого циклу та поставленої задачі, включаючи створення, налаштування, експлуатацію, технічне обслуговування та утилізацію.
- ФК16. Здатність проектувати, впроваджувати та обслуговувати високопродуктивні паралельні та розподілені комп'ютерні системи та їх складові з використання ПЛІС модулів і систем автоматизованого проектування

У відповідності до вищезазначеного, підсилені загальні та фахові компетенції дадуть наступні результати навчання:

- отримають досвід командного виконання проектів з використанням усіх фаз життєвого циклу (від збору вимог до впровадження);
- навчаться проводити допроектний R&D етап проектування;
- навчаться розробляти технічну пропозицію і технічне завдання;
- навчаться проводити інженерія вимог (бізнес-аналіз);
- отримають практичний досвід DevOps;
- отримають практичний досвід супроводження; організація процесів (моделей життєвого циклу) за класичними та Agile моделями;
- отримають досвід виконання проектів з використанням різних методологій розробки (класичних та Agile) та на різних ролях (розробник, тестувальник, аналітик, DevOps, керівник тощо);
- отримають базовий практичний досвід оцінювання якості (верифікація і тестування);
- отримають практичний досвід знаходження помилок в програмному та апаратному забезпеченні убудованих систем;
- отримають досвід використання автоматизованих засобів командної розробки проектів таких;
- отримають практичний досвід звітування і здачі проекту;
- отримають досвід бізнес-комунікації (наради, презентації, активне слухання, письмова комунікація, проходження співбесіди тощо);
- отримають досвід роботи в команді (надавання зворотного зв'язку, ділення досвідом, вирішування конфліктних ситуацій тощо);
- отримають досвід управління часом;
- отримають досвід практичного вирішення проблемних ситуацій;

- отримають досвід відстоювання власних рішень у професійній дискусії та представлення результатів власних розробок;
- отримають досвід командної роботи над складними технічними проектами відповідно до сучасних вимог до процесів і технологій їх реалізації;
- навчаться складати дієве резюме, представляти свої особисті якості та проходити співбесіди.

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

При вивченні дисципліни «Управління інфраструктурними ІТ проектами» доцільно використовувати знання, отримані при вивченні попередніх дисциплін: «Вступ до операційної системи Linux», «Архітектура комп'ютерів. Частина 1. Управляючі та арифметичні пристрої», «Архітектура комп'ютерів. Частина 2. Процесори»; дисципліни із Ф-каталога вибіркового дисциплін, що входять до складу сертифікатної програми: «Технології програмування на С для вбудованих систем», «Технології програмування на ПЛІС (FPGA)».

Дисципліна є базовою для курсів із Ф-каталога вибіркового дисциплін, що входять до складу сертифікатної програми: «Технології тестування (QA) вбудованих систем», «Технології проектування інтелектуальних систем» (вибіркова), , а також є корисною для вивчення нормативних дисциплін «Архітектура комп'ютерів. Частина 3. Мікропроцесорні засоби», «Архітектура комп'ютерів. Курсова робота», «Системне програмування», «Комп'ютерні мережі», «Комп'ютерні системи».

3. Структура кредитного модуля

РОЗДІЛ І. ОРГАНІЗАЦІЯ ПРОЦЕСІВ УПРАВЛІННЯ ІТ ПРОЄКТАМИ.

Лекція 1. Вступ до управління ІТ проектами в галузі розроблення вбудованих систем

Структура та задачі дисципліни.

Вимоги ІТ індустрії до гнучких навичок (soft skills).

Основні поняття управління ІТ проектами.

Лекція 2. Моделі управління ІТ проектами.

Моделі життєвого циклу складних програмно-апаратних комплексів.

Гнучкі – Agile моделі життєвого циклу: Scrum; Kanban.

Інструментарій управління ІТ проектами.

Лекція 3. Процеси управління ІТ проектами на R&D етапі

Цілі, суб'єкти, об'єкти R&D етапу.

Артефакти R&D етапу.

Лекція 4. Організація команди та командна робота в процесі управління ІТ проектами.

Колектив та ролі на ІТ проектах.

Підбір колективу, розробка та вивчення резюме.

Колективна робота на ІТ підприємствах.

Лекція 5. Управління ІТ проектами на етапі інженерії вимог

Теоретичні основи інженерії вимог.

Методи формального аналізу вимог та техніки уточнення вимог.

РОЗДІЛ ІІ. РЕАЛІЗАЦІЯ ПРОЦЕСІВ УПРАВЛІННЯ ІТ ПРОЄКТАМИ.

Лекція 6. Створення ефективних архітектурних рішень програмних систем.

Теоретичні основи проектування архітектури

складних програмно-апаратних комплексів (IoT).

Технології створення ефективних архітектурних рішень.

Лекція 7. Командна робота колективу на ІТ проектах

Роль єдності в колективі.

Ефект команди та її роль в управлінні ІТ проектами.

Лекція 8. Проектування та реалізація процесів людино-машиної взаємодії (UX/UI дизайн)

Принципи людино-машиної взаємодії в складних програмно-апаратних комплексах (IoT)

Практичні аспекти та артефакти UX/UI проектування

Лекція 9. Колективна розробка складних програмно-апаратних комплексів (IoT).

Колективна розробка програмних систем з позицій системного аналізу.

Збіркове програмування.

«Чистий» код в колективній роботі.

Лекція 10. Прийняття рішень в групі – ІТ команді.

Теоретичні відомості групового прийняття рішень: теорія прийняття управлінських рішень; групове прийняття рішень.

Практика прийняття індивідуальних та групових рішень.

Лекція 11. Оцінювання та забезпечення якості (QA) в процесах управління ІТ проектами.

Якість, ефективність, верифікація, тестування – процеси та артефакти.

Якість програмного забезпечення та програмно-апаратних систем.

Лекція 12. Внутрішні процеси командного управління на ІТ проектах. Практика провідних ІТ компаній.

Лідерство.

Конфліктологія: компроміс та консенсус.

Тайм-менеджмент: колектив та персоналії.

Лекція 13. Підготовка та проведення ефективних колективних заходів.

Наради та перемовини.

Ефективна презентація.

Лекція 14. Впровадження та супроводження складних програмно-апаратних комплексів (IoT).

Практика провідних ІТ компаній.

Процеси впровадження та супроводження.

Практика впровадження та супроводження.

Лекція 15. Напрямки розвитку технологій управління ІТ проектами в області вбудованих систем в умовах 4-ї індустріальної революції (Industry 4.0/IoT)

ІТ проекти сучасності та управління ними.

ІТ проекти перспективи та управління ними.

4. Навчальні ресурси та матеріали

4.1. Базова література

1. Кузьмініх В.О., Коваль О.В., Воронько М.П. Оцінка часу виконання типових задач проектів на підприємствах з функціональною організаційною структурою// Реєстрація, зберігання і обробка даних, ISSN 1560-9189, 2012 т. 14, № 3, с.77-82
2. Кузьмініх В.О., Хаустов Д.В., Коростельова Є.Ю. Аналіз ризиків у корпоративній системі управління проектами// Реєстрація, зберігання і обробка даних, 2010 . – т. 12, № 3, с. 99–107
3. Кузьмініх В.О., Коваль О.В. Реалізація сценарного підходу в управлінні проектами на основі типових задач // Реєстрація, зберігання і обробка даних, ISSN 1560-9189, 2015.т. 17, №1,с.77-87
4. Кузьмініх В.О. Трирівнева корпоративна система управління проектами//Реєстрація, зберігання і обробка даних, 2009, т.11, №3, с. 75–82

4.2. Додаткова література

5. Лавріщева К.М. Програмна інженерія. – К.: 2008. – 319 с.

6. Managing Successful Projects with PRINCE2 (2009 Edition) - Office of Government Commerce, 2009 4. 8. Eric S. Norman. Work Breakdown Structure. ISBN 9780470177129; 2008 г. – 304 с.

4.3. Інформаційні ресурси

1. Дистанційний курс – на платформі дистанційного навчання «Сікорський» в середовищі Google Workspace for Education: <https://classroom.google.com/c/NTgwNDU1NTQ3ODgz?cjc=qzdv3d4>

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компоненту)

Розподіл навчального часу за видами занять і завдань з дисципліни згідно з робочим навчальним планом. На кредитний модуль виділено 120 годин та 4 кредитів – 18 лекцій (36 годин), 9 лабораторних робіт (18 годин)

Для досягнення мети навчальної дисципліни слід зосередитись в лекційному матеріалі на управлінні ІТ проектами в галузі розроблення вбудованих систем, а також навичок командної роботи: набуття Soft skills шляхом застосування своїх Hard skills для розв'язку задач практики

Мета лабораторних робіт – придбання практичних навичок управління ІТ проектами в галузі розроблення вбудованих систем, а також навичок командної роботи в рамках таких проектів. Ціллю виконання лабораторних робіт є досконале розуміння сучасних процесів командної розробки складної системи на практиці, отримання необхідного досвіду та навичок для участі в таких проектах на ролі розробника, а також отриманні базових знань для організації та реалізації проектів в ІТ компаніях на різних ролях, в тому числі, як DevOps.

Лабораторні роботи організовані, як виконання і реалізація стартап-проекту в сфері розроблення вбудованих систем. Результатом опрацювання завдань дисципліни є завершений практичний проект.

Тематика лабораторних робіт

Лабораторна робота 1. Встановлення робочого проекту. Ознайомлення з вимогами та додавання нових вимог

Лабораторна робота 2. Створення спринту та його оцінка.

Лабораторна робота 3. Ci/CD Jenkins

Лабораторна робота 4. Написання коду в команді.

Лабораторна робота 5. Рев'ю коду в команді

Лабораторна робота 6. Написання юніт тестів

Лабораторна робота 7. Запуск автоматичної збірки коду в команді

Лабораторна робота 8. Проведення ретроспективи

6. Самостійна робота здобувача вищої освіти денної форми навчання

Види самостійної роботи (66 годин):

- підготовка до аудиторних занять, (0,5 годин x 18 лекцій = 9 годин);
- підготовка та оброблення проведення розрахунків за первинними даними, отриманими на лабораторних заняттях, виконання лабораторних робіт, розв'язок задач, розміщення результатів на GitLab (рекомендовано 1,5 годин x 8 лабораторних робіт = 12 годин);
- виконання модульної контрольної роботи (2 МКР x 4 години = 8 годин);
- підготовка до експрес-тестів (4 години);
- підготовка до заліку (4 години);

- опрацювання тем на самостійну роботу, завантаження та збирання програмного забезпечення для виконання лабораторних робіт (29 годин).

Тем на самостійне опрацювання (денна форма навчання)

- Тема 1. Вимоги ІТ індустрії до гнучких навичок (soft skills).
- Тема 2. Інструментарій управління ІТ проектами.
- Тема 3. Підбір колективу, розробка та вивчення резюме.
- Тема 4. Колективна робота на ІТ підприємствах.
- Тема 5. Технології створення ефективних архітектурних рішень.
- Тема 6. Конфліктологія: компроміс та консенсус.
- Тема 7. Наради та перемовини.
- Тема 8. ІТ проекти сучасності та управління ними.
- Тема 9. ІТ проекти перспективи та управління ними.

5. Методика викладання дисципліни на заочній формі навчання

Зміст лекцій та самостійної роботи

Лекція 1.

РОЗДІЛ І. ОРГАНІЗАЦІЯ ПРОЦЕСІВ УПРАВЛІННЯ ІТ ПРОЄКТАМИ.

Вступ до управління ІТ проектами в галузі розроблення вбудованих систем

Структура та задачі дисципліни.

Основні поняття управління ІТ проектами.

Моделі управління ІТ проектами.

Моделі життєвого циклу складних програмно-апаратних комплексів.

Гнучкі – Agile моделі життєвого циклу: Scrum; Kanban.

Тем на самостійне опрацювання

Вимоги ІТ індустрії до гнучких навичок (soft skills).

Інструментарій управління ІТ проектами.

Лекція 2.

Процеси управління ІТ проектами на R&D етапі

Цілі, суб'єкти, об'єкти R&D етапу.

Артефакти R&D етапу.

Організація команди та командна робота в процесі управління ІТ проектами.

Колектив та ролі на ІТ проектах.

Управління ІТ проектами на етапі інженерії вимог

Теоретичні основи інженерії вимог.

Тем на самостійне опрацювання

Підбір колективу, розробка та вивчення резюме.

Колективна робота на ІТ підприємствах.

Методи формального аналізу вимог та техніки уточнення вимог.

Лекція 3.

РОЗДІЛ ІІ. РЕАЛІЗАЦІЯ ПРОЦЕСІВ УПРАВЛІННЯ ІТ ПРОЄКТАМИ.

Створення ефективних архітектурних рішень програмних систем.

Теоретичні основи проектування архітектури складних програмно-апаратних комплексів (IoT).

Командна робота колективу на IT проектах

Роль єдності в колективі.

Проектування та реалізація процесів людино-машиної взаємодії (UX/UI дизайн)

Практичні аспекти та артефакти UX/UI проектування

Теми на самостійне опрацювання

Технології створення ефективних архітектурних рішень.

Ефект команди та її роль в управлінні IT проектами.

Принципи людино-машиної взаємодії в складних програмно-апаратних комплексів (IoT)

Лекція 4

Колективна розробка складних програмно-апаратних комплексів (IoT).

«Чистий» код в колективній роботі.

Прийняття рішень в групі – IT команді.

Практика прийняття індивідуальних та групових рішень.

Оцінювання та забезпечення якості (QA) в процесах управління IT проектами.

Якість, ефективність, верифікація, тестування – процеси та артефакти.

Якість програмного забезпечення та програмно-апаратних систем.

Внутрішні процеси командного управління на IT проектах. Практика провідних IT компаній.

Тайм-менеджмент: колектив та персоналії.

Впровадження та супроводження складних програмно-апаратних комплексів (IoT). Практика провідних IT компаній.

Процеси впровадження та супроводження.

Практика впровадження та супроводження.

Теми на самостійне опрацювання

Колективна розробка програмних систем з позицій системного аналізу.

Збіркове програмування.

Теоретичні відомості групового прийняття рішень: теорія прийняття управлінських рішень; групове прийняття рішень.

Лідерство.

Конфліктологія: компроміс та консенсус.

Підготовка та проведення ефективних колективних заходів.

Наради та перемовини.

Ефективна презентація.

Напрямки розвитку технологій управління IT проектами в області вбудованих систем в умовах 4-ї індустріальної революції (Industry 4.0/IoT)

IT проекти сучасності та управління ними.

IT проекти перспективи та управління ними.

Тематика лабораторних робіт для самостійного виконання

Лабораторна робота 1. Встановлення робочого проекту. Ознайомлення з вимогами та додавання нових вимог

Лабораторна робота 2. Створення спринту та його оцінка.

Лабораторна робота 3. Ci/CD Jenkins

Лабораторна робота 4. Написання коду в команді.

Лабораторна робота 5. Рев'ю коду в команді

Тематика лабораторних робіт для аудиторного виконання

Аудиторне заняття 1:

Захист лабораторних робіт 1-5. Тестування.

Аудиторне заняття 2:

Лабораторна робота 6. Написання юніт тестів

Аудиторне заняття 3:

Лабораторна робота 7. Запуск автоматичної збірки коду в команді

Аудиторне заняття 4:

Лабораторна робота 8. Проведення ретроспективи

Види самостійної роботи для здобувачів заочної форми навчання (104 години):

- підготовка до аудиторних занять, (1,5 години x 4 лекцій = 6 годин);
- підготовка до експрес-тестів (4 години);
- самостійне виконання лабораторних робіт 1-5 (3,5 години x 5 лаб. робіт = 17,5 годин);
- підготовка до виконання лабораторних робіт (6 – 8), оформлення протоколів, розміщення протоколів на GitLab (1,5 години x 3 лабораторних робіт = 4,5 годин)
- підготовка до МКР1 (4 годин);
- виконання МКР2 (4 години);
- підготовка до заліку (4 години);
- самостійне опрацювання теоретичного матеріалу, завантаження та збирання програмного забезпечення для виконання лабораторних робіт (60 годин).

Політика та контроль

5. Політика навчальної дисципліни (освітнього компонента)

Для виконання лабораторних робіт та модульних контрольних робіт встановлюються дедлайни.

Виконання лабораторних робіт поза встановлених термінів супроводжуються штрафними балами, які вираховуються із оцінки за протокол. МКР не приймається поза встановлені терміни.

Штрафні бали виставляються за: невчасну здачу лабораторної роботи. Кількість штрафних балів не більше 10. Штрафні бали та жорсткі дедлайни не запроваджуються у період військового стану

Заохочувальні бали виставляються за: активну участь на лекціях; виконання поточних домашніх завдань, ведення конспекту, підготовка повідомлення з презентацією по одній із тем СРС дисципліни тощо. Кількість заохочуваних балів не більше 10.

Окремі теми лекційних занять супроводжуються короткими експрес-тестами (на 15 хвилин), які включають матеріал вивченої теми та питання, які задані на самостійне вивчення. Бали отримані за тест входять в семестрову рейтингову оцінку. Поточні тести не перескладаються.

Виконанню кожної лабораторної роботи передуює виконання індивідуального завдання і оформлення його у вигляді протоколу. Студент, який прийшов на заняття без оформленого протоколу до лабораторної роботи не допускається. Першим етапом студент захищає результати отримані під час виконання індивідуального завдання до лабораторної роботи, на другому етапі –

захищає теорію шляхом усного опитування або тестування. Більшість лабораторних робіт супроводжуються тестами для оцінки вивченого теоретичного та практичного матеріалу до лабораторної роботи. Бали отримані за виконання лабораторної роботи, за тест та за протокол входять в оцінку за лабораторну роботу. Тестування проводиться на лабораторному занятті після перевірки результатів виконання лабораторних робіт. Студент, який не виконав індивідуальне завдання до лабораторної роботи й до тесту не допускається.

Виконання лабораторних робіт є обов'язковими для допуску до семестрового контролю. Умовою допуску до семестрового контролю є зарахування всіх лабораторних робіт та стартовий рейтинг не менше 30 балів.

Модульна контрольна робота пишеться на лекційному занятті без застосування допоміжних засобів (мобільні телефони, планшети та ін.); результат пересилається до відповідної директорії Google -диску через Google форму.

Модульна контрольна робота не переписується за умови негативної оцінки, негативна оцінка за МКР (менше ніж 9 балів (<60%)) прирівнюється до 0 балів, в цьому випадку МКР не зараховується.

Оцінка, яку студент може отримати за виконання кожної лабораторної роботи та за кожну модульну контрольну роботу наведені в таблиці 1 оцінювання семестрових робіт, розділ 8 силабуса.

Таким чином мінімальна оцінка, яку повинен отримати студент для допуску до семестрового контролю дорівнює 60 балів, максимальна – 100 балів за виконання всіх поточних робіт за семестр.

Здобувачі, які виконали всі умови допуску до заліку (виконали всі лабораторні роботи) та мають рейтингову оцінку менше 60 балів, а також ті здобувачі, хто бажає підвищити свою рейтингову оцінку, на останньому за розкладом занятті мають можливість пройти семестровий контроль у вигляді залікової контрольної роботи.

У разі виконання залікової контрольної роботи рейтингова оцінка визначається як сума балів за залікову контрольну роботу та балів за індивідуальні семестрові завдання.

До індивідуальних семестрових завдань вноситься індивідуальна робота студента що стосується виконання лабораторних робіт. Максимальна кількість балів за індивідуальну роботу за семестр дорівнює 60 балів. Максимальна оцінка за залікову контрольну роботу дорівнює 40 балів. Таким чином здобувач має можливість підвищити свою рейтингову оцінку написавши залікову контрольну роботу і додавши додаткові бали до кількості балів, які отримані протягом семестру за індивідуальну семестрову роботу.

Після виконання залікової контрольної роботи, якщо оцінка за залікову контрольну роботу більша ніж за рейтингом, здобувач отримує оцінку за результатами залікової контрольної роботи. Якщо оцінка за залікову контрольну роботу менша ніж за рейтингом, попередній рейтинг здобувача (за винятком балів за семестрове індивідуальне завдання) скасовується і він отримує оцінку з урахуванням результатів залікової контрольної роботи. Цей варіант формує відповідальне ставлення здобувача до прийняття рішення про виконання залікової контрольної роботи, змушує його критично оцінити рівень своєї підготовки та ретельно готуватися до заліку.

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

Семестровий рейтинг студента з кредитного модуля розраховується, виходячи із 100-бальної шкали. Рейтинг складається з балів, що студент отримує за виконання 8 лабораторних робіт R_L , двох модульних контрольних робіт $R_{МКР}$ та експерт-тестів $R_{ЕТ}$.

Максимальна кількість балів за лабораторні роботи складає 60 балів, тобто $R_L = 60$.

Узагальнені критерії оцінювання лабораторних робіт наступні:

- своєчасність підготовки протоколу до лабораторного заняття, повнота виконання теоретичного або практичного завдання в протоколі, протокол вчасно викладений на GitLab;
- коректність функціонування розроблених моделей на програмному або апаратному забезпеченні, демонстрація власного репозиторію на GitLab з матеріалами лабораторної роботи та, наявність комітів;
- опитування за тематикою лабораторної роботи для зарахування практичної частини роботи, захист одержаних в роботі результатів, відповіді на додаткові теоретичні запитання викладача, повнота оформлення звіту/протоколу по роботі на GitLab.

Детальний підхід до оцінювання кожної лабораторної роботи наведений в таблиці 1.

Таблиця 1. Деталізація оцінювання кожної лабораторної роботи

Назва заняття	Форма контролю	Кількість балів	Допуск до семестрового контролю	Всього балів
Лабораторна робота 1.	Вступний тест	4	2	4
Лабораторна робота 2	Виконання завдання	3	5	8
	Опитування	3		
	Протокол на GitLab	2		
Лабораторна робота 3	Виконання завдання	3	5	8
	Опитування по QA	3		
	Протокол на GitLab	2		
Лабораторна робота 4	Виконання завдання	3	5	8
	Опитування по QA	3		
	Протокол на GitLab	2		
Лабораторна робота 5	Виконання завдання	3	5	8
	Опитування	3		
	Протокол на GitLab	2		
Лабораторна робота 6	Виконання завдання	3	5	8
	Опитування	3		
	Протокол на GitLab / демонстрація	2		
Лабораторна робота 7	Виконання завдання	3	5	8
	Опитування	3		
	Протокол на GitLab / демонстрація	2		
Лабораторна робота 8	Виконання завдання	3	5	8
	Опитування	3		
	Протокол на GitLab / демонстрація	2		
Кількість балів за індивідуальну роботу				60
Експрес-тести на лекціях	2 x 5	10	5	10

МКР	МКР1 (Тест)	15	9	15
	МКР2	15	9	15
Всього балів		100	60	100

Максимальна кількість балів за МКР $R_{МКР} = 2 \times 15 = 30$ балів.

МКР1 проводиться у вигляді автоматизованого тестування на платформі Google Workspace for Education. Тест складається із 60 питань $R_{МКР_2} = 0,25 \times 60 = 15$ балів

Модульна контрольна робота МКР2 виконується самостійно за індивідуальним завданням. Критерії оцінювання МКР2 за чотирма рівнями:

- правильна та змістовна відповідь з поясненнями в термінах предметної області : 13 – 15 балів;
- правильна відповідь, неповні пояснення: 11 – 12 балів;
- відповідь містить помилки: 9 – 10 балів;
- відповідь містить суттєві помилки, немає пояснень: 4-8 балів;
- немає відповіді: 0 балів.

Оцінка за МКР2 знижується за:

- некоректне оформлення;
- відсутність коментарів в змістовних термінах;
- відсутність пояснень під час розрахунків.

Максимальна кількість балів за експрес-тести – 10 балів, тести проводяться на лекціях у вигляді автоматизованого тестування на платформі Google Workspace for Education.

Максимальна кількість балів за залікову контрольну роботу дорівнює $R_3 = 40$ балів.

Залікова контрольна робота проводиться у вигляді автоматизованого тестування на платформі Google Workspace for Education / moodle, складається з вибраних питань, які були протягом семестру в МКР, експрес тестах, на захистах лабораторних робіт. Максимальна оцінка за залікову контрольну роботу $R_3 = 40$ балів.

Календарна атестація студентів (на 8 та 14 тижнях семестрів) з дисципліни проводиться за значенням поточного рейтингу студента на час атестації. Якщо значення цього рейтингу не менше 50 % від максимально можливого на час атестації, студент вважається атестованим. В іншому випадку в атестаційній відомості виставляється «неатестовано».

Кількість балів, що отримує студент за семестр визначається за формулою

$$R = R_{л} + R_{МКР} + R_{E-тести}$$

Максимальна кількість балів за семестр не перевищує $R_C = 100$.

З урахуванням одержаної суми балів кінцева оцінка визначається за таблицею 3.

Якщо студент пише залікову роботу, кількість балів, що отримує студент за семестр визначається формулою

$$R = R_{лр} + R_3$$

де, $R_{лр} = R_{л}$.

Максимальна кількість балів за семестр не перевищує $R = 100$.

З урахуванням одержаної суми балів кінцева оцінка визначається таблицею 3.

<i>Кількість балів</i>	<i>Оцінка</i>
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, д.т.н, доцент, професор кафедри ОТ Клименко Ірина
Анатоліївна, асистент кафедри ОТ, Таранюк Вікторія
Анатоліївна.

Ухвалено кафедрою обчислювальної техніки (протокол № 10 від 25.05.2022 р.)

Погоджено методичною комісією ФІОТ (протокол №10 від 09.06.2022 р.)