

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій
Кафедра оптоелектроніки та інформаційних технологій

Затверджено

на засіданні кафедри оптоелектроніки та інформаційних технологій
факультету електроніки та комп'ютерних технологій
Львівського національного університету імені Івана Франка
(протокол №6 від 24 червня 2025 р.)

Завідувач кафедри _____

Силабус з навчальної дисципліни
«Проект з розробки системи опрацювання даних»,
що викладається в межах ОПІ «Комп'ютерні науки»
першого (бакалаврського) рівня вищої освіти для здобувачів
зі спеціальності 122 Комп'ютерні науки

Львів 2025

Назва дисципліни	Проект з розробки системи опрацювання даних
Адреса викладання дисципліни	м. Львів, вул. Тарнавського, 107
Факультет та кафедра, за якою закріплена дисципліна	Факультет електроніки та комп'ютерних технологій, кафедра оптоелектроніки та інформаційних технологій
Галузь знань, шифр та назва спеціальності	12 Інформаційні технології, 122 Комп'ютерні науки
Викладачі дисципліни	Васюта Василь Михайлович, канд. фіз.-мат. наук, асистент Горон Богдан Ігорович, докт. філос., доцент Кушнір Олег Степанович, докт. фіз.-мат. наук, проф., проф.
Контактна інформація викладачів	vasyl.vasiuta@lnu.edu.ua bohdan.horon@lnu.edu.ua oleh.kushnir@lnu.edu.ua https://electronics.lnu.edu.ua/employee/vasiuta-vasyl-mykhaylovykh/ https://electronics.lnu.edu.ua/employee/horon-bohdan-ihorovykh/ https://electronics.lnu.edu.ua/employee/kushnir-o-s
Консультації з питань навчання по дисципліні відбуваються	Консультації в день проведення семінарських занять (за попередньою домовленістю). Також можливі онлайн-консультації через Zoom. Для погодження часу онлайн-консультацій слід писати на електронну пошту викладача.
Сторінка дисципліни	https://e-learning.lnu.edu.ua/course/view.php?id=6957
Інформація про дисципліну	Дисципліна «Проект з розробки системи опрацювання даних» є вибірковою дисципліною для студентів, що навчаються за спеціальністю 122 Комп'ютерні науки в рамках освітньої програми «Комп'ютерні науки». Її викладають в 6 семестрі в обсязі 4,0 кредитів (за Європейською Кредитно-Трансферною Системою – ECTS).
Коротка анотація дисципліни	Навчальну дисципліну розроблено для ознайомлення студентів із сучасними практичними підходами до планування та створення комп'ютерних систем опрацювання даних групою виконавців із різними спеціалізаціями. Під час виконання практичної частини курсу студент виконує завдання у відповідності до обраної спеціалізації: розробник програмного забезпечення, інженер контролю якості, проєктний менеджер, DevOps, архітектор програмного забезпечення, бізнес-аналітик або дизайнер UI/UX.
Мета та цілі дисципліни	<p>Метою вивчення дисципліни «Проект з розробки системи опрацювання даних» є ознайомлення студентів з теоретичними основами підходів до організації роботи та безпосередньої розробки комп'ютерної системи для опрацювання даних, а також формування в студентів практичних навичок, які б дали змогу ефективно виконувати задачі з розробки системи у відповідності до обраної спеціалізації.</p> <p>Спеціальні компетентності, які здобуваються під час вивчення цієї навчальної дисципліни, такі:</p> <p>СК 8. Здатність проєктувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.</p> <p>СК 9. Здатність реалізувати багаторівневу обчислювальну модель на основі архітектури клієнт-сервер, включаючи бази даних, знань і сховища даних, виконувати розподілену обробку великих наборів даних на кластерах стандартних серверів для забезпечення обчис-</p>

	<p>лювальних потреб користувачів, у тому числі на хмарних сервісах.</p> <p>СК 10. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника.</p> <p>СК 11. Здатність до інтелектуального аналізу даних на основі методів обчислювального інтелекту включно з великими та погано структурованими даними, їхньої оперативної обробки та візуалізації результатів аналізу в процесі розв'язування прикладних задач.</p> <p>СК 12. Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення.</p>
<p>Література для вивчення дисципліни</p>	<p>Основна література:</p> <ol style="list-style-type: none"> 1. Томас Д. Програміст-прагматик: друге видання / Томас Девід, Хант Ендрю. – 2024. – 368 с. 2. Мартін Р. Чиста архітектура. Мистецтво розробки програмного забезпечення / Мартін Роберт. – Фабула, Серія #PROSystem, 2019. – 368 с. 3. Graham D. Foundations of Software Testing. ISTQB Certification / Graham Dorothy, Black Rex, van Veenendaal Erik. – CENGAGE, 2018. – 290 р. 4. Піхлер Р. Agile продукт-менеджмент за допомогою Scrum / Піхлер Роман. – Фабула, Серія #PROSystem, 2019. – 128 с. 5. Wiegers K. Software Requirements (Developer Best Practices) 3rd Edition / Wiegers Karl, Beatty Joy. – Print2print, 2013. – 672 р. 6. Готельф Дж. Learn UX. Створення класних продуктів з командами Agile / Готельф Джеф, Сейден Джош. – ArtHuss, 2024. – 206 с. 7. Кім Дж. DevOps. Посібник / Кім Джин, Хамбл Джек, Дебуа Патрік, Вілліс Джон. – Фабула, Серія #PROSystem, 2023. – 384 с. 8. Kleppmann M. Designing Data-Intensive Applications / Kleppmann Martin. – O`Reilly Media, 2021. – 614 р. 9. Методичні рекомендації до виконання практикуму з навчальної дисципліни «Проект з розробки системи опрацювання даних» для студентів спеціальності 122 Комп'ютерні науки, які навчаються за освітньою програмою «Комп'ютерні науки» / В. Васюта, Б. Горон, О. Кушнір. – Львів, ЛНУ, 2025. – 24 с. <p>Додаткова література:</p> <ol style="list-style-type: none"> 10. Фрімен Е. Head First. Паттерни проектування / Е. Фрімен, Е. Робсон, К. Сієрра, Б. Бейтс. – Фабула, Серія #PROSystem, 2020. – 672 с. 11. Петтон Дж. Мапа історій користувача. Відкрий правдиву історію, створи саме той продукт / Петтон Дж., Ікономі П. – ArtHuss, Серія PRJCTR-teka, 2022. – 276 с. 12. Рейнвотер Г. Дж. Як пасти котів. Посібник для програмістів, які мають керувати іншими програмістами / Рейнвотер Дж. Генк. – Фабула, Серія #PROSystem, 2020. – 320 с. 13. Посібник зі Скраму
<p>Обсяг курсу</p>	<p>Сумарно 120 годин. Із них 48 годин практичних занять і 72 години самостійної роботи.</p>
<p>Очікувані результати навчання</p>	<p>Після завершення цього курсу студент буде:</p> <ul style="list-style-type: none"> - знати основні спеціалізації ІТ-спеціалістів та їхнє коло відповідальності, етапи життєвого циклу продукту від етапу пресеюлу до підт-

	<p>римки, базові засади скраму та Agile, методи оцінювання часу виконання задач, підходи до розв'язування проблем, пов'язаних з обробкою даних;</p> <ul style="list-style-type: none"> - вміти виконувати завдання у відповідності до обраної спеціалізації, вміти готувати технічну документацію та звіти у відповідності до обраної спеціалізації, організувати роботу у відповідності до скраму.
Ключові слова	Розробка програмного забезпечення, життєвий цикл програмного забезпечення, скрам, планування та організація розробки проєкту
Формат курсу	Очний
	Проведення практичних занять (включно з семінарами та лабораторними проєктами) та консультації для поглибленого розуміння тем
Теми	Див. СХЕМА КУРСУ
Підсумковий контроль, форма	Залік в кінці семестру
Пререквізити	Для вивчення курсу студенти потребують базових знань з дисциплін «Алгоритмізація та програмування», «Теорія обчислень, алгоритми і структури даних», «Об'єктно-орієнтоване програмування», «Організація баз даних та знань», «Розробка та проєктування інформаційних систем»
Навчальні методи та техніки, які будуть використовуватися під час викладання курсу	Практичні заняття, що включають презентації, семінари, лабораторні завдання, індивідуальні практичні завдання, обговорення та дискусії.
Необхідне обладнання	Мультимедіа, платформи Microsoft Teams, Moodle і Zoom, комп'ютерне програмне забезпечення
Критерії оцінювання (окремо для кожного виду навчальної діяльності)	<p>Оцінювання проводиться упродовж семестру та під час залікової сесії за 100-бальною шкалою. Бали нараховуються за такими видами робіт із таким співвідношенням:</p> <ul style="list-style-type: none"> • практичні завдання (лабораторні роботи): 45% оцінки; максимальна кількість балів 45; • усне представлення фінального результату роботи: 15% оцінки; максимальна кількість балів 15; • контрольні заміри знань (2 модулі): 40% оцінки; максимальна кількість балів 40. <p>Загалом 100 балів.</p> <hr/> <p>Контрольні заміри знань проводять у формі стандартних практичних завдань і теоретичних питань.</p> <p>Академічна добросесність: Очікується, що лабораторні та контрольні роботи студентів будуть їхніми оригінальними дослідженнями або міркуваннями. Відсутність посилань на використані джерела, фабрикавання джерел, списування, втручання в роботу інших студентів становлять, але не обмежують, приклади можливої академічної недобросесності. Виявлення ознак академічної недобросесності в роботі студента є підставою для її незарахування викладачем, незалежно від масштабів плагіату або спроб обману.</p> <p>Відвідування занять є важливою складовою навчання. Очікується, що всі студенти відвідають усі заняття курсу. Студенти мають інформувати викладача про неможливість відвідати заняття. Студенти зобов'язані дотримуватися всіх термінів, визначених для виконання видів робіт, передбачених курсом.</p> <p>Література. Уся література, яку студенти не зможуть знайти самостійно, буде надана викладачем виключно в освітніх цілях без права її</p>

	<p>передачі третім особам. Студенти також заохочуються до використання іншої літератури та джерел, зокрема наукової літератури, яка відсутня серед обов'язкової та рекомендованої.</p> <p>Політика виставлення балів. Враховуються бали, набрані на поточному опитуванні, самостійній роботі та бали підсумкового контролю знань. Обов'язково враховуються присутність на заняттях та активність студента під час занять; наголошується на неприпустимості користування мобільним телефоном, планшетом або іншими мобільними пристроями під час занять з метою, не пов'язаною з навчанням, списування та плагіату, несвоєчасного виконання поставлених завдань і т. ін. Жодні форми порушення академічної доброчесності не толеруються.</p>
<p>Питання до контрольних робіт</p>	<p>Перелік питань і завдань для проведення підсумкової оцінки знань усіх тем курсу до контрольних робіт розміщено на веб-сторінці:</p> <ol style="list-style-type: none"> 1. Відповідальності розробника ПЗ. 2. Відповідальності інженера контролю якості (тестувальника). 3. Відповідальності проєктного менеджера. 4. Відповідальності DevOps. 5. Відповідальності архітектора ПЗ. 6. Відповідальності бізнес-аналітика. 7. Відповідальності UI/UX дизайнера. 8. Рефакторинг. 9. Юніт-тести. 10. Інтеграційні тести. 11. Тести продуктивності. 12. Smoke-тести. 13. Регресійне тестування. 14. Тест-стратегія. 15. Тест-план. 16. Тест-сценарій. 17. Пенетрейшн-тестування. 18. Менеджмент ризиків. 19. CI/CD-пайплайни. 20. Матриця відповідальностей RACI. 21. Діаграма Ганта. 22. Дорожня карта проєкту. 23. Задача фази пресейлу. 24. Складові компоненти контракту розробки ПЗ. 25. Типи контрактів: фіксована ціна, час-матеріали, окрема команда. 26. Задача фази дискавері. 27. Цілі та обсяг проєкту (Project Scope). 28. Поняття про PoC (proof of concept). 29. Складові і ціль документу BRD (Business Requirements Definition). 30. Складові і ціль документу FRD (Functional Requirements Definition). 31. Функціональні та нефункціональні вимоги. 32. Різновиди нефункціональних вимог. 33. Користувацький сценарій (Use Case). 34. Ієрархія задач: епіки, фічі та юзер-сторі. 37. Складові і ціль документу WBS (Work Breakdown Structure). 38. Універсальна мова UML. 39. Класова діаграма UML. 40. Діаграма послідовностей UML. 41. Діаграма сценаріїв UML. 42. Архітектурна діаграма. 43. Принципи побудови складу команди для проєкту. 44. Життєвий цикл програмного забезпечення (SDLC). 45. Каскадна методологія розробки (Waterfall). 46. Маніфест Agile. 12 принципів маніфесту Agile. 47. Поняття про скрам і канбан. Скрамбан. 48. Дошка канбану та її складові компоненти. 49. Тест-керована розробка (TDD). 50. Парне та екстремальне програмування. 51. Масштабування методології розробки: скрам скрамів і модель Spotify. 52. Мінімальний життєздатний продукт (MVP).

	<p>53. Задачі і функціонал систем проектного трекінгу: Jira, Azure DevOps, Trello.</p> <p>54. Метрики продуктивності команди.</p> <p>55. Документ аналізу конкурентів.</p> <p>56. SWOT-аналіз.</p> <p>57. Складові і ціль документу SRS (Service Requirement Specification).</p> <p>58. Складові і ціль посібника користувача (User Guide).</p> <p>59. Поняття про Test Item, Test Feature, Test Case і відмінності між ними.</p> <p>60. Складові і ціль документу Test Summary Report.</p> <p>61. Поняття про дизайн-систему та її складові.</p> <p>62. Мокапи, ваєрфрейми та прототипи дизайну.</p> <p>63. Шкала Лікерта.</p> <p>64. Шкала NPS (Net Promoter Score).</p> <p>65. Менеджмент ризиків та супутні документи. Матриця ризиків. Ймовірність та імпакт.</p> <p>66. Технічна документація: OpenAPI, StoryBook та інші.</p> <p>67. Складові і ціль документу Run Book.</p> <p>68. Складові і ціль деплоймент (ролбек) плану.</p> <p>69. Дорожня карта міграції.</p> <p>70. Зустрічі в скрамі: щоденні стендапи, планування спринта, грумінг беклогу, ревію спринта, ретроспектива.</p> <p>71. Беклог спринта і беклог продукту.</p> <p>72. Метод аналогій оцінки тривалості проекту.</p> <p>73. Метод оцінки тривалості проекту способом розбиття на частини.</p> <p>74. Нисхідний метод оцінки тривалості проекту.</p> <p>75. Метод експертної оцінки. Модель Кука.</p> <p>76. Дельфійський метод.</p> <p>77. Пленнінг-покер.</p> <p>78. Три- і двочкова оцінки.</p> <p>79. Оцінки в годинах і сторі-пойнтах.</p>
Опитування	Анкету-оцінку з метою оцінювання якості курсу буде надано по завершенню курсу.

СХЕМА КУРСУ

Тиж.	Тема, план, короткі тези	Форма діяльності (заняття)	Література, ресурси в Інтернеті	Завдання (год.)	Термін виконання
1	Вступне заняття. Академічна доброчесність. Необхідність опрацювання великої кількості даних при експоненційному зростанні збереженої інформації. Виклики і проблеми обробки даних терабайтних масштабів. Дані і машинне навчання. Спеціалізації IT-спеціалістів. Відповідальності кожної зі сторін. RACI-матриця. Результати розробки. Діаграма Ганта та дорожня карта проекту.	Практ.	[4], середовище Moodle	3 год.	Кінець поточного тижня
2	Фаза пресеєлу та фаза дискавері. Типи контрактів. Функціональні і нефункціональні вимоги. BRD vs FRD. WBS.	Практ.	[5], середовище Moodle	1 год.	Кінець поточного тижня
3	Старт розробки. Архітектурна діаграма системи. Універсальна мова моделювання UML. Правильний вибір технологій та складу команди.	Практ.	[1], [2], [4], середовище Moodle	1 год.	Кінець поточного тижня
4	Життєвий цикл програмного забезпечення SDLC. Методології розробки ПЗ: каскадна модель, Agile та інші. Мінімальний життєвий продукт MVP.	Практ.	[4], середовище Moodle	1 год.	Кінець поточного тижня
1-4	Одна з лабораторних, залежно від ролі:	Практ. (Лаб.)	середовище Moodle	6 год.	Кінець тижня 5

	<p>Лаб. №1.1. Створення кістяків проєктів. Налаштування кросс-катінгу в аспектно-орієнтованому підході..</p> <p>Лаб. №1.2. Розробка тест-плану, написання тест кейсів до функціональних вимог.</p> <p>Лаб. №1.3. Старт проєкту. Створення базової проєктної документації.</p> <p>Лаб. №1.4. Налаштування гітхаб-політик для гілок. Створення docker-файлу і налаштування локального запуску.</p> <p>Лаб. №1.5. Побудова архітектурної діаграми. Вибір і обґрунтування обраних технологій</p> <p>Лаб. №1.6. Виділення функціональних особливостей. Розбиття задачі по функціональних частинах.</p> <p>Лаб. №1.7. Створення палітри кольорів, схеми розмірів та типографії.</p>				
5	Захисне заняття 1. Кінець фази дискавері.	Практ.	[1]-[13], середовище Moodle	3 год.	Кінець поточного тижня
6	Системи проєктного трекінгу та їхні особливості. Трекінг прогресу задач. Канбан-дошка та її елементи. Дашборди з проєктними метриками. Бурндаун-чарт і командна швидкість. Звітування робочого часу.	Практ.	[4], середовище Moodle	1 год.	Кінець поточного тижня
7	Демо. Тестування №1.	Практ.	[1]-[13], середовище Moodle	3 год.	Кінець поточного тижня
8	Документація в розробці ПЗ I. Найкращі практики створення документації. Складові частини звіту про аналіз конкурентів. SWOT-аналіз. SRS. Test Items vs Test Features vs Test Cases. Test Summary Report. Дизайн-система. Ваєрфрейми, мокапи і прототипи.	Практ.	[3], [4], [6], середовище Moodle	1 год.	Кінець поточного тижня
9	Документація в розробці ПЗ II. Документація проєктного менеджера. Менеджмент ризиків. Технічна документація: OpenAPI, StoryBook, RunBook. Зовнішні звіти і звіти ассесменту. CI і CD.	Практ.	[1], [2], [4], [7], середовище Moodle	1 год.	Кінець поточного тижня
6–9	Одна з лабораторних, залежно від ролі: Лаб. №2.1. Написання логічної частини аплікації. Лаб. №2.2. Написання тест кейсів і проведення нефункціонального тестування. Лаб. №2.3. Налаштування системи проєктного трекінгу. Лаб. №2.4. Налаштування хостингу. Лаб. №2.5. Створення WBS. Лаб. №2.6. Написання звіту про аналіз конкурентів. Лаб. №2.7. Створення дизайн-системи окремих компонент.	Практ. (Лаб.)	середовище Moodle	6 год.	Кінець тижня 10
10	Захисте заняття 2. MVP.	Практ.	[1]-[13], середовище Moodle	3 год.	Кінець поточного тижня
11	Скрам. Спринт. Початок спринта. Планування спринта. Щоденні стедапи. Груммінг беклогу. Рев'ю	Практ.	[3], [4], середовище Moodle	1 год.	Кінець поточного тижня

	спринта. Ретроспектива. 1-1.				
12	Оцінка задачі. Оцінка в годинах і сторі-пойнтах. Методи оцінювання. Метод аналогій. Метод розбиття на частини. Нисхідний метод. Метод експертної оцінки. Дельфійський метод. Триточкова оцінка.	Практ.	[1], [3], середовище Moodle	1 год.	Кінець поточного тижня
13	Робота з беклогом. Беклог спринта і беклог продукту. Пленнінг-покер. Аналіз стратегії розвитку продукту.	Практ.	[4], середовище Moodle	1 год.	Кінець поточного тижня
14	Реліз продукту. Представлення результатів.	Практ.	[1]-[13], середовище Moodle	3 год.	Кінець поточного тижня
11–14	Одна з лабораторних, залежно від ролі: Лаб. №3.1. Написання автоматизованих тестів. Лаб. №3.2. Створення звіту про результати тестування (Test Summary Report) Лаб. №3.3. Розробка проєктної документації і організація роботи команди. Лаб. №3.4. Створення CI/CD-пайплайнів. Лаб. №3.5. Створення Run Book. Лаб. №3.6. Розробка SRS. Лаб. №3.7. Створення мокапів базових сценаріїв.	Практ. (Лаб.)	середовище Moodle	6 год.	Кінець тижня 15
15	Захисне заняття 3. Реліз продукту.	Практ.	[1]-[13], середовище Moodle	3 год.	Кінець поточного тижня
16	Заключне заняття. Тестування №2. Анкетування.	Практ.	[1]-[13], середовище Moodle	3 год.	Кінець поточного тижня