

Чернівецький національний університет імені Юрія Федьковича

(повне найменування закладу вищої освіти)

Факультет математики та інформатики

(назва факультету/навчально-наукового інституту)

Кафедра математичного моделювання

(назва кафедри, що забезпечує викладання)

“ЗАТВЕРДЖУЮ”

**Декан факультету
математики та інформатики**

Ольга МАРТИНЮК



“28”

серпня

2026 року

РОБОЧА ПРОГРАМА

навчальної дисципліни

Управління IT-проектами

(назва навчальної дисципліни)

обов'язкова

(вказати: обов'язкова)

Освітньо-професійна програма **Інформаційні технології та управління проектами**

(назва програми)

Спеціальність **122 Комп'ютерні науки**

(вказати: код, назва)

Галузь знань **12 Інформаційні технології**

(вказати: шифр, назва)

Рівень вищої освіти **перший (бакалаврський)**

(вказати: перший (бакалаврський) / другий (магістерський) / третій (освітньо-науковий))

Факультет **математики та інформатики**

(назва факультету/ навчально-наукового інституту, на якому здійснюється підготовка фахівців за вказаною освітньою програмою)

Мова навчання **українська**

(вказати: на якій мові читається дисципліна)

Чернівці 2026 рік

Робоча програма навчальної дисципліни «Управління ІТ-проектами» складена відповідно до освітньо-професійної програми «Інформаційні технології та управління проектами»

Розробники:

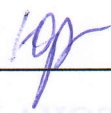
Піддубна Лариса Андріївна, доцент кафедри математичного моделювання, кандидат фіз.-мат. наук, доцент;

Перцов Андрій Сергійович, доцент кафедри математичного моделювання, кандидат фіз.-мат. наук, доцент

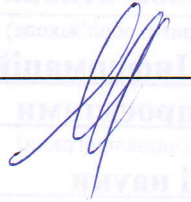
Викладачі, що забезпечують читання даної навчальної дисципліни:

Піддубна Лариса Андріївна, доцент кафедри математичного моделювання, кандидат фіз.-мат. наук, доцент;

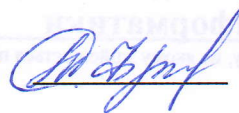
Перцов Андрій Сергійович, доцент кафедри математичного моделювання, кандидат фіз.-мат. наук, доцент

Погоджено з гарантом ОП  **Ігор ЮРЧЕНКО**

Затверджено на засіданні кафедри математичного моделювання
Протокол № 1 від «27» серпня 2026 року

Завідувач кафедри  **Ігор ЧЕРЕВКО**

Схвалено методичною радою факультету математики та інформатики
Протокол № 1 від «28» серпня 2026 року

Голова методичної ради  **Тоня ФРАТАВЧАН**

Мета навчальної дисципліни: Дисципліна “Управління ІТ проєктами” націлена на оволодіння студентами основами теоретичних знань та практичних навичок щодо основних підходів та засад управління ІТ-проєктами, використання практичних інструментів управління ІТ-проєктами в ролі членів команд з управління ІТ-проєктами, усвідомлення взаємозв’язку між теоретичним підґрунтям управління проєктами та його прикладним застосуванням у конкретній галузі. Надання майбутнім фахівцям сучасні фундаментальні знання з основних аспектів управління ІТ-проєктами, а також набуті навичок адаптації і впровадження проєктних рішень у практичну діяльність.

Пререквізити. Навчальні дисципліни: “Програмування”, “Бази даних та інформаційні системи”, «Проектування програмних систем».

Постреквізити. Навчальні дисципліни: Курсова робота, Виробнича практика.

Результати навчання. У результаті вивчення навчальної дисципліни студенти повинні

знати основні теоретичні та практичні питання організації та виконання проєктів, пов’язаних із розробкою програмних систем довільної складності та архітектури,

вміти формувати структуру програмного засобу відповідно до обраної архітектури, оформляти документацію, коди відповідно до визнаних стандартів, організовувати роботу команду за обраною методикою, оцінювати ризики виконання проєкту згідно зі стандартом вищої освіти з урахуванням таких загальних і фахових компетентностей, а також програмних результатів навчання:

ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК8. Здатність генерувати нові ідеї (креативність).

ЗК9. Здатність працювати в команді.

ФК3. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проєктування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв’язності та нерозв’язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.

ФК7. Здатність застосовувати теоретичні та практичні основи методології та технології моделювання для дослідження характеристик і поведінки складних об’єктів і систем, проводити обчислювальні експерименти з обробкою й аналізом результатів.

ФК8. 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.

ФК9. 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 обчислювальних потреб користувачів, у тому числі на хмарних сервісах.

ФК10. 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.

ПРН8. Використовувати методологію системного аналізу об'єктів, процесів і систем для задач аналізу, прогнозування, управління та проєктування динамічних процесів в макроекономічних, технічних, технологічних і фінансових об'єктах.

ПРН9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

ПРН10. Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проєктувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування.

ПРН11. Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проєктну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).

Опис навчальної дисципліни Загальна інформація

Форма навчання	Рік підготовки	Семестр	Кількість		Кількість годин						Вид підсумкового контролю
			кредитів	годин	лекції	практичні	семінарські	лабораторні	самостійна робота	індивідуальні завдання	
Денна	4	7	4	120	30	—	—	15	75	—	залік

Структурний зміст навчальної дисципліни

Назви змістових модулів і тем	Кількість годин											
	денна форма						Заочна форма					
	усього	у тому числі					усього	у тому числі				
		л	п	лаб	інд	с.р.		л	п	лаб	інд	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Змістовий модуль 1.												
Тема 1.1. Огляд історії розвитку УП.	6	2				4	—	—	—	—	—	—
Тема 1.2. Загальна характеристика сучасних методологій УП ІТ проєктами. Класичні моделі розробки ПЗ:	15	2		1		12	—	—	—	—	—	—

каскадно-водоспадна, ітераційна, інкрементна. Перехід до Agile. Фреймворк Synefin												
Разом за змістовим модулем 1.	21	4		1		16	–	–	–	–	–	–
Змістовий модуль 2.												
Тема 2.1. Agile – методологія управління ІТ проектами.	15	4		3		8	–	–	–	–	–	–
Тема 2.2. SCRUM методологія управління ІТ проектами. Поняття Kanban. Поняття WIP limits, Lead Time та Cycle Time. Scrum vs Kanban	13	4		2		7						
Тема 2.3. Методики оцінки параметрів ІТ проекту. Техніки відносної оцінки: T-Shirt Sizing (XS, S, M, L, XL) для високорівне-вого планува-ння. Декомпо-зиція задач (Task Breakdown) та їх оцінка в годи-нах . Основні метрики. Поняття Velocity (швид-кість команди) та Capacity.	15	4		3		8						
Тема 2.4. Особливості командної роботи. Ролі та обов'язки у програмній команді, переваги командної роботи, ризики та складність такої співпраці Моделі керування командною роботою. Матриця відповідальності RACI . Етапи розвитку команди за Такманом (Forming, Storming, Norming, Performing)	17	4		3		10						

Тема 2.5. Методика підбору членів команди у виконанні ІТ проєкту. Промислові технології розробки. Середовища розробки (Dev, Stage, Prod). Поняття CI/CD очима ПМа. Управління Технічним боргом (Technical Debt).	10	4				6						
Тема 2.6. Оцінка ризиків ІТ проєктів	13	4		3		10						
Тема 2.7. Поняття PRD (Product Requirements Document). Організація бази знань проєкту (Confluence / Notion).	4	2				10	–	–	–	–	–	–
Разом за змістовим модулем 2	99	26		14		59	–	–	–	–	–	–
Всього годин	120	30		15		75	–	–	–	–	–	–
Підсумковий модуль	залік											

Тематика лекційних занять з переліком питань

№	Назва теми з основними питаннями
1	Тема 1.1. Огляд історії розвитку УП.
2	Тема 1.2. Загальна характеристика сучасних методологій УП ІТ проєктами. Класичні моделі розробки ПЗ: каскадна, ітераційна, інкрементна. Перехід до Agile. Фреймворк Synefin.
3	Тема 2.1. Agile – методологія управління ІТ проєктами.
4	Тема 2.2. SCRUM методологія управління ІТ проєктами. Поняття Kanban, WIP limits, Lead Time та Cycle Time. Scrum vs Kanban
5	Тема 2.3. Методики оцінки параметрів ІТ проєкту. Техніки відносної оцінки: T-Shirt Sizing . Velocity та Capacity.
6	Тема 2.4. Особливості командної роботи. Ролі, матриця відповідальності RACI . Етапи розвитку команди за Такманом.
7	Тема 2.5. Методика підбору членів команди у виконанні ІТ проєкту. Середовища розробки (Dev, Stage, Prod). Поняття CI/CD очима ПМа. Управління Технічним боргом (Technical Debt).
8	Тема 2.6. Оцінка ризиків ІТ проєктів
9	Тема 2.7. Поняття PRD (Product Requirements Document). Організація бази знань проєкту.

Детальні презентації до кожної лекції наведено на сайті електронного навчання на сторінці курсу <https://moodle.chnu.edu.ua/course/view.php?id=4861>

Тематика лабораторних занять з переліком питань

№	Назва теми (завдання)
1	<i>Лабораторна робота №1. Розробка технічного завдання</i> Завдання. 1. Заповнити Lean Canvas (4 блоки: Проблема, Рішення із зазначенням реального текстста, Цільова аудиторія, Унікальна цінність). 2. Скласти Project Charter (Мета, Score In, Score Out, Стейкхолдери). 3. Розширити зміст Lean Canvas (додати ще 5 блоків: канали залучення, реальні статті витрат типу хостингу чи API, потоки доходів, ключові метрики, прихована перевага). 4. У Project Charter додати аналіз 1-2 реальних конкурентів та описати жорсткі технічні обмеження (Constraints) обраного текстста. Скласти паспорт IT проекту за обраною ПО.
2	<i>Лабораторна робота №2.</i> Завдання. 1. Написати 10 User Stories. - Asserptance Criteria(AC) мають містити не лише бізнес-логіку, а й технічні деталі реалізації (назви полів у БД, вимоги до API, конкретні повідомлення про помилки). - Пріоритезація MoSCoW. 2. Доповнити AC у професійному форматі BDD (Given / When / Then). 3. Замінити метод пріоритезації з MoSCoW на кількісний WSJF, обов'язково додавши коротке письмове обґрунтування (чому виставлені саме такі бали Value та Job Size) для кожної функціональності.
3	<i>Лабораторна робота №3</i> Планування роботи над проектом засобами Jira (Trello) Завдання 1. Створити дошку Scrum/Kanban. Завести Епіки, прив'язати до них User Stories. 2. Оцінити весь беклог методом T-Shirt Sizing (відносна оцінка). 3. Декомпозувати 1-2 сторі на технічні Sub-tasks (фронт, бек, база) та оцінити їх у реальних годинах. 4. Наповнити перший Спринт задачами та натиснути "Start Sprint". 5. Розрахувати реальний Capacity команди та пояснити свій Focus Factor (наприклад, 70% на код, 30% на мітинги). Наповнити спринт задачами суворо в межах розрахованого Capacity. 6. Налаштувати кастомний Workflow (наприклад: To Do, In Progress, Code Review, Done). Проставити залежності між задачами (Issue Links: <i>blocks / is blocked by</i>).
4	<i>Лабораторна робота № 4. Управління ризиками та комунікаціями</i> Завдання. 1. За допомогою інструментів Confluence, Notion створити простір проекту. 2. Скласти Risk Register на 5 ризиків (Ймовірність, Вплив, Стратегія). 3. Побудувати матрицю RACI з реальними прізвищами студентів команди. 4. Для Risk Register прорахувати кількісний Risk Score та написати детальний Action Plan (Mitigation) для топ-2 ризиків.

	5. Провести тестовий командний мітинг (скрін екрану чи фото додати) та написати Meeting Notes із реальними Action items. 6. Створити технічну Onboarding-сторінку із інструкцією по розгортанню локального середовища і лінками на репозиторій.
--	---

Детальні завдання до кожної лабораторної роботи наведено на сайті електронного навчання на сторінці курсу <https://moodle.chnu.edu.ua/course/view.php?id=4861>

Контрольні питання до лабораторних робіт

Лабораторна робота №1

1. Що таке проєкт і які його ключові характеристики?
2. Назвіть та поясніть основні обмеження проєкту (трикутник проєкту: зміст, терміни, вартість).
3. Опишіть п'ять фаз життєвого циклу проєкту згідно з PMBOK.
4. Чим проєкт відрізняється від операційної діяльності? Наведіть приклади.
5. Хто такі "зацікавлені сторони" (stakeholders) і чому важливо керувати їхніми очікуваннями?

Лабораторна робота №2

1. Назвіть ключові відмінності між каскадною (Waterfall) та гнучкою (Agile) методологіями.
2. Опишіть ролі в методології Scrum: Product Owner, Scrum Master, Development Team.
3. Які основні артефакти Scrum? Поясніть: Product Backlog, Sprint Backlog, Increment.
4. Що таке "спринт" і скільки часу він зазвичай триває?

Лабораторна робота №3

1. Опишіть процедуру управління запитами на зміни в проєкті.
2. Які документи створюються на етапі управління змістом?
3. Поясніть, що таке "діаграма Ганта" і які дані вона відображає.
4. Які переваги та недоліки використання діаграми Ганта?
5. Що таке Критичний шлях (Critical Path) проєкту? Як його визначити і навіщо?
6. Поясніть, що таке "вільний резерв часу" (Free Slack/Float) і "загальний резерв часу" (Total Slack/Float).

Лабораторна робота № 4

1. Яка основна мета створення окремого простору (Space) для проєкту в Confluence або Notion, а не просто зберігання файлів на диску?
2. У чому різниця між стратегіями "Mitigation" (пом'якшення) та "Acceptance" (прийняття) при роботі з ризиками? Наведіть приклад ризику, який краще прийняти, а не пом'якшувати.
3. Розшифруйте аббревіатуру RACI. Яка роль відповідає за виконання завдання, а хто несе кінцеву відповідальність за його результат (навіть якщо не робить його руками)?
4. Як розраховується кількісний показник Risk Score (пріоритет ризику)? Як цей показник допомагає визначити топ-2 ризики для створення детального Action Plan?
5. Що таке Meeting Notes (нотатки зустрічі) і чому вони вважаються важливішими за протокол? Який обов'язковий елемент має бути в нотатках наприкінці, щоб зустріч вважалася ефективною?
6. Чому інструкція з розгортання локального середовища (onboarding-сторінка) має містити не тільки текстові кроки, а й посилання на репозиторій та список залежностей (наприклад, версії Node.js, Python, Docker)?
7. Уявіть, що під час командного мітингу ви виявили новий ризик, якого не було в початковому Risk Register. Ваші дії згідно з процедурами, описаними в лабораторній роботі?

Завдання для самостійної роботи студентів

Самостійна робота складається з повторення матеріалу, засвоєного на лекціях, самостійного опанування частини теоретичного матеріалу, роботи з контрольними запитаннями та завданнями.

Студенти можуть отримувати до 1 бала в рахунок виконання завдань СРС під час кожного лекційного заняття за правильні відповіді на запитання лектора, активне обговорення багатоваріантних підходів до рішення представленої лектором проблеми (для активізації пошукової та дослідної діяльності студентів).

№	Назва теми	Завдання для самостійної роботи	К-сть год.
1	Теми 1.1-2.7	Підготовка до лекційних занять	20
2	Теми 1.1-2.7	Підготовка до лабораторних занять	35
3	Теми 1.1-2.7	Підготовка до підсумкового модуль-контролю	20

Зміст завдань для самостійної роботи

№	Навчальний елемент	Зміст	Год.
1	Тема 1.1. Огляд історії розвитку УП.	Вивчити поняття командної роботи, причини такого підходу.	4
2	Тема 1.2. Загальна характеристики-ка сучасних методологій УП ІТ проектами. Класичні моделі роз-робки ПЗ: каскадно-водоспадна, ітераційна, інкрементна. Перехід до Agile. Фреймворк Synefin.	Вивчити класичні моделі розробки ПЗ: каскадно-водоспадна, ітераційна, інкрементна. Скласти таблицю порівняння цих моделей. Перехід до Agile. Фреймворк Synefin.	12
3	Тема 2.1. Agile – методологія управління ІТ проектами.	Вивчити основні ідеї цієї методики, її переваги і недоліки, умови застосування.	8
4	Тема 2.2. SCRUM методологія управління ІТ проектами. Поняття Kanban. Поняття WIP limits, Lead Time та Cycle Time. Scrum vs Kanban	Вивчити основні ідеї цієї методики, її переваги і недоліки, умови застосування. Поняття Kanban. Поняття WIP limits, Lead Time та Cycle Time. Scrum vs Kanban	7
5	Тема 2.3. Методики оцінки параметрів ІТ проекту. Техніки відносної оцінки: T-Shirt Sizing (XS, S, M, L, XL) для високорівневого планування. Декомпозиція задач (Task Breakdown) та їх оцінка в	Класифікація та призначення параметрів проекту. Складіть порівняльну таблицю параметрів методик. Техніки відносної оцінки: T-Shirt Sizing (XS, S, M, L, XL) для високорівневого планування. Декомпозиція задач (Task Breakdown) та їх оцінка в годинах . Основні метрики. Поняття Velocity (швид-кість команди) та Capacity.	8

	годинах. Основні метрики. Поняття Velocity (швидкість команди) та Capacity.		
6	Тема 2.4. Особливості командної роботи. Ролі та обов'язки у програмній команді, переваги командної роботи, ризики та складність такої співпраці. Моделі керування командною роботою. Матриця відповідальності RACI . Етапи розвитку команди за Такманом (Forming, Storming, Norming, Performing)	Скласти таблицю важливості параметрів складання команди для виконання ІТ проекту. Проаналізувати моделі керування командною роботою. Розглянути, проаналізувати, вивчити матрицю відповідальності RACI . Етапи розвитку команди за Такманом (Forming, Storming, Norming, Performing)	10
7	Тема 2.5. Методика підбору членів команди у виконанні ІТ проекту. Промислові технології розробки. Середовища розробки (Dev, Stage, Prod). Поняття CI/CD очима ПМа. Управління Технічним боргом (Technical Debt).	Вивчити середовища розробки (Dev, Stage, Prod). Поняття CI/CD очима ПМа. Управління Технічним боргом (Technical Debt).	6
8	Тема 2.6. Оцінка ризиків ІТ проєктів	Вивчити методи визначення ризиків, їх класифікацію.	10
9	Тема 2.7. Поняття PRD (Product Requirements Document). Організація бази знань проєкту (Confluence / Notion).	Розглянути, вивчити інструмент Confluence / Notion	10

Самостійна робота складається з повторення матеріалу, засвоєного на лекціях, самостійного опанування частини теоретичного матеріалу, роботи з контрольними запитаннями та завданнями.

Методи навчання

Методи навчання та викладання: лекції, лабораторні заняття, електронне навчання з використанням системи Moodle, тестування.

Система контролю та оцінювання

Критерієм підсумкового оцінювання є досягнення студентом мінімальних порогових рівнів оцінок (балів) за кожним передбаченим результатом навчання. Мінімальний пороговий рівень оцінки варто визначати за допомогою якісних критеріїв і трансформувати його в мінімальну позитивну оцінку використовуваної числової (рейтингової) шкали.

Система оцінювання рівня навчальних досягнень ґрунтується на принципах ECTS та є накопичувальною. Протягом семестру студенти виконують 3 лабораторних робіт. Кожна лабораторна робота оцінюється по 20 балів.

Виконуючи завдання лабораторної роботи, студент повинен оформити і завантажити для подальшої перевірки на сайт електронного навчання звіт разом із працездатними файлами програмної реалізації завдань ЛР (правила оформлення наведені на сторінці навчальної дисципліни на сайті).

50% балів, відведених на оцінювання ЛР, студент отримує за працюючий програмний продукт, в якому реалізовано всі завдання ЛР та оформлений звіт. Решта 50% балів виставляється після захисту студентом виконаного звіту. На захисті звіту з ЛР студент має відповісти на питання щодо власної реалізації. При відповіді на теоретичні питання та питання щодо своєї розробки БД у випадку неістотної помилки знімається 10-20% балів, а у випадку істотної 20-40% балів, якщо ж студент не опанував теоретичний матеріал, плутається в означеннях, наводить логічно невірні твердження, то знімається до 50% балів від усієї суми балів за ЛР.

Максимальна кількість, яку можна набрати на підсумковому модулі (тестування) – 40 балів.

Підсумкова оцінка виставляється за результатами суми балів, набраних на змістових модулях під час семестру та підсумковому модулі згідно з нижченаведеною таблицею.

Методи формування професійної компетентності: розповідь, пояснення, бесіда, демонстрація, візуалізація, дискусія тощо. Методи формування практичних умінь та навичок: розв'язування задач лабораторних робіт, виконання завдань, розробка та аналіз алгоритмів і програмного коду, захист звітів з лабораторних робіт.

Засобами оцінювання та демонстрування результатів навчання є: стандартизовані тести; аналітичні звіти з лабораторних робіт; презентації результатів виконаних завдань та досліджень ІНДЗ, усний контроль у вигляді індивідуального та фронтального опитування на лекціях та лабораторних заняттях.

Формами поточного контролю є усна чи письмова (тестування, лабораторна робота, ІНДЗ) відповідь студента.

Формою підсумкового контролю є залік.

Критерії оцінювання поточного та підсумкового контролю

Розподіл балів, які отримують студенти

Поточне тестування та самостійна робота						Модуль контроль	Сума
Змістовий модуль №1		Змістовий модуль № 2					
Тема 1.1	Тема 1.2	Тема 2.3	Тема 2.4	Тема 2.5	Тема 2.6		
15		15		15	15	40	100

Шкала оцінювання: національна та ЄКТС

Оцінка за національною шкалою	Оцінка за шкалою ECTS	
	Оцінка (бали)	Пояснення за розширеною шкалою
Відмінно	A (90-100)	відмінно
Добре	B (80-89)	дуже добре
	C (70-79)	добре
Задовільно	D (60-69)	задовільно
	E (50-59)	достатньо
Незадовільно	FX (35-49)	(незадовільно) з можливістю повторного складання
	F (1-34)	(незадовільно) з обов'язковим самостійним повторним опрацюванням освітнього компонента до перескладання

Перелік питань для самоконтролю та підсумкового контролю навчальних досягнень студентів

ЗРАЗКИ ТЕОРЕТИЧНИХ ПИТАНЬ

Рівень 1. Базовий (Знання та термінологія)

1. Дайте визначення поняттю «проект».
2. Що таке «трикутник обмежень» проекту?
3. Чим проект відрізняється від операційної діяльності?
4. Назвіть 5 фаз життєвого циклу проекту згідно з PMBOK.
5. Хто такі стейкхолдери (зацікавлені сторони)?
6. Яку роль відіграє Статут проекту (Project Charter)?
7. Що таке Офіс управління проектами (PMO)?
8. У чому сутність каскадної (Waterfall) моделі?
9. Сформулюйте основні цінності Agile-маніфесту.
10. Опишіть роль Product Owner у Scrum.
11. Які обов'язки виконує Scrum Master?
12. Хто входить до складу Development Team?
13. Що таке Product Backlog?
14. Поясніть поняття «спринт».
15. Що таке Sprint Backlog?
16. Що таке щоденний Scrum (Daily)?
17. Дайте визначення Інкременту продукту.
18. Опишіть філософію Kanban.
19. Що таке WIP-ліміти?
20. Що таке Зміст проекту (Scope)?
21. Поясніть призначення WBS (ієрархічна структура робіт).
22. Опишіть призначення діаграми Ганта.
23. Розшифруйте матрицю RACI.
24. Що таке MVP (Minimum Viable Product)?
25. Що таке «Технічний борг»?
26. Для чого використовується Jira або Trello?
27. Що таке Meeting Notes?
28. Що таке User Stories?
29. Чим відрізняються середовища Dev, Stage та Prod?

Рівень 2. Середній (Розуміння процесів та метрик)

30. Порівняйте ітераційну та інкрементну моделі розробки.
31. Опишіть мету ретроспективи спринту.
32. Поясніть різницю між метриками Lead Time та Cycle Time.
33. У чому ключові відмінності між Scrum та Kanban?
34. Що таке «повзучість змісту» (Scope Creep) та як їй запобігти?
35. Опишіть техніку відносної оцінки T-Shirt Sizing.
36. Поясніть техніку Planning Poker.
37. Що таке Velocity (швидкість команди) і як вона вимірюється?
38. Що таке Capacity (ємність) команди?
39. Поясніть різницю між вільним та загальним резервом часу.
40. Назвіть 5 етапів розвитку команди за Такманом.
41. Яку інформацію містить Реєстр ризиків (Risk Register)?
42. Опишіть 4 стратегії реагування на ризики (Mitigation, Acceptance тощо).
43. Як розраховується показник Risk Score?
44. Що таке PRD (Product Requirements Document)?
45. Поясніть роль бази знань (Confluence/Notion) у проєкті.
46. Опишіть процедуру управління змінами (Change Management).
47. Поясніть різницю між функціональними та нефункціональними вимогами.
48. Що таке Acceptance Criteria (AC)?
49. Поясніть метод пріоритезації MoSCoW.
50. Як формат «Given / When / Then» допомагає в описі задач?
51. Що таке Focus Factor команди?
52. Які документи створюються на етапі закриття проєкту?

Рівень 3. Високий (Аналіз та практичне застосування)

53. Як фреймворк Synefin допомагає менеджеру обрати між Waterfall та Agile?
54. Порівняйте функціональну, проєктну та матричну структури: яку обрати для стартапу?
55. Що таке бізнес-кейс (Business Case) і які фінансові показники він має містити?
56. Що таке Критичний шлях (Critical Path) і як його зміна впливає на дедлайн?
57. Як розраховується пріоритет за методом WSJF (Weighted Shortest Job First)?
58. Опишіть структуру Lean Canvas: чому він кращий за традиційний бізнес-план для ІТ?
59. Опишіть концепцію CI/CD з точки зору управління ризиками релізу.
60. Як менеджеру ефективно вирішувати конфлікти на етапі Storming (за Такманом)?
61. Які метрики KPI найкраще відображають успішність проєкту (крім бюджету та термінів)?
62. Опишіть техніку декомпозиції задач (Task Breakdown) для великої фічі.
63. Як правильно провести Onboarding нового розробника в команду?
64. Аналіз ризиків: у чому різниця між «пом'якшенням» (Mitigation) та «прийняттям» (Acceptance) у критичних ситуаціях?
65. Роль «уроків, отриманих» (Lessons Learned): як впровадити їх у наступний проєкт?
66. Оцініть вплив «технічного боргу» на Velocity команди в довгостроковій перспективі.
67. Який обов'язковий елемент (Action Items) робить Meeting Notes інструментом контролю?
68. Чому Product Owner не може бути Scrum Master-ом в одній команді? (Конфлікт інтересів).
69. Як обґрунтувати замовнику необхідність написання документації, якщо він хоче «тільки код»?
70. Опишіть дії менеджера, якщо команда не встигає виконати Sprint Backlog до кінця спринту.

Зарахування результатів неформальної освіти

Зарахування результатів неформальної освіти проводиться у відповідності до "Порядку визнання у Чернівецькому національному університеті імені Юрія Федьковича результатів навчання, здобутих шляхом неформальної та/або інформальної освіти" (затверджено Вченою радою ЧНУ протокол № 16 від 25

листопада 2024 р.) <https://www.chnu.edu.ua/media/4g5fzssb/poriadok-vyznannia-rezultativ-navchannia-zdobutykh-shliakhom-neformalnoi-ta-abo-informalnoi-osvity.pdf>.

Рекомендована література

1. Посібник зі зведення знань з управління проектами (посібник PMBOK®): Project Management Institut, 2017. 6-е изд. 726 с.
2. Cynefin framework / Модель «Кеневін» / <https://www.maxzosim.com/cynefin-framework-model-kenevin/>
3. Що таке Канбан // <https://brainrain.com.ua/uk/shcho-take-kanban/>
4. Посібник з Канбану // <https://kanbanguides.org/the-kanban-guide/2025.5/>
5. Що таке сторі поінти // <https://hire1.com.ua/glossary/story-points>
6. Planning (SCRUM) poker // <https://training.qatestlab.com/blog/technical-articles/planning-poker/>
7. Проектні метрики // <https://pmtips.com.ua/post/shcho-take-metryky>
8. Матриця RACI // <https://wizeclub.education/blog/matritsya-raci-shho-tse-yak-stvoriti-ta-vikoristovuvati/>
9. 5 етапів розвитку команди // <https://peoplefirst.club/media/5-etapiv-rozvytku-komandy-i-do-chogo-tut-konflikty>
10. Технічний борг // <https://dou.ua/forums/topic/35707/>
11. Що таке CI/CD, як він працює та коли знадобиться на проєкті. // <https://highload.tech/uk/blogs/shho-take-ci-cd-yak-vin-pratsyuye-ta-koli-znadobitsya-na-proyekti-lajfhaki-ta-bad-practices/>

Інформаційні ресурси

1. Project Management Institute. (2019). A Guide to the Project Management Body of Knowledge (PMBOK(R) Guide-Sixth Edition / Agile Practice Guide Bundle (HINDI). [N.p.]: Project Management Institute. Retrieved from <http://search.ebscohost.com/login.aspx?direct=true&site=eds-live&db=edsebk&AN=2240462>
2. The Scrum Guide™ [Електронний ресурс]. – 2017. - Режим доступу до ресурсу: <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf#zoom=100>
3. Azure DevOps documentation [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: <https://docs.microsoft.com/en-us/azure/devops/?view=azure-devops>.

Політика академічної доброчесності

Дотримання політики щодо академічної доброчесності учасниками освітнього процесу при вивченні освітнього компонента регламентовано чинними документами ЧНУ:

- "Кодекс академічної доброчесності та етики Чернівецького національного університету імені Юрія Федьковича" <https://www.chnu.edu.ua/universitytet/normatyvni-dokumenty/kodeks-akademichnoi-dobrochesnosti-ta-etyky-chernivetskoho-natsionalnoho-universitytetu-imeni-yuriiia-fedkovycha/>
- "Положення про виявлення та запобігання академічного плагіату у Чернівецькому національному університеті імені Юрія Федьковича" <https://www.chnu.edu.ua/universitytet/normatyvni-dokumenty/polozhennia-pro-vyavlennia-ta-zapobihannia-akademichnomu-plahiatu-u-chernivetskomu-natsionalnomu-universityteti-imeni-yuriiia-fedkovycha/>

- "Політика використання штучного інтелекту в Чернівецькому національному університеті імені Юрія Федьковича"
<https://www.chnu.edu.ua/universytet/normatyvni-dokumenty/polityka-vykorystannia-shtuchnoho-intelektu-v-chernivetskomu-natsionalnomu-universyteti-imeni-yuriia-fedkovycha-zi-zminamy-zatverdzhenyi-vchenoiu-radoiu-chnu-imeni-yuriia-fedkovycha/>

Використання здобувачами освіти інструментів штучного інтелекту (ШІ) під час виконання навчальних завдань:

- допускається за умови дотримання принципів академічної доброчесності (здобувач освіти несе відповідальність за зміст, достовірність і результати поданої роботи, а у разі використання ШІ зобов'язаний зазначити факт і спосіб його застосування);
- не допускається використання ШІ як заміни самостійного виконання завдань або з метою порушення вимог академічної доброчесності;
- додаткові обмеження чи вимоги щодо використання ШІ можуть визначатися викладачем в освітньому процесі для конкретних видів навчальної діяльності відповідно до особливостей дисципліни.