

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

Факультет інформаційних технологій

Кафедра технологій управління

Спеціальність 122 «Комп'ютерні науки»
Освітньо-наукова програма «Управління проєктами»

Кваліфікаційна робота магістра
на тему:

«Дослідження процесів управління проєктом розробки інтернет порталу
Trade&Earn для торгівлі (керування) криптовалютою»

Студента 2-го курсу групи УП-21

Науковий керівник:

Гончар Максим
(прізвище, ім'я)

Доцент, к.т.н
(науковий ступінь, вчене звання)

Хлевний Андрій
(прізвище, ім'я)

(підпис студента)

(дата)

(підпис)

Попередній захист:

(Висновок: «До захисту в Екзаменаційній комісії»)

Завідувач кафедри
технологій управління

(підпис)

(прізвище, ініціали)
(дата)

Київ – 2026 р.

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
Факультет інформаційних технологій**

Кафедра технологій управління
Освітній рівень Магістр
Спеціальність 122 Комп'ютерні науки
Освітня програма Управління проєктами

ЗАТВЕРДЖУЮ
Завідувач кафедри
професор Морозов В.В.
5 грудня 2025 року

**ЗАВДАННЯ
НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ**

Студент Гончар Максим Юрійович

Група: УП-21

1. Тема кваліфікаційної роботи: Дослідження процесів управління проєктом розробки інтернет порталу Trade&Earn для торгівлі (керування) криптовалютою

Затверджена наказом від 5 грудня 2025 р. № 4

2. Строк подання студентом готової роботи – «12» травня 2026 р.

3. Цільова установка та вихідні дані до роботи: Дослідити та застосувати гібридну модель управління проєктом (Waterfall + Scrum) при організації розробки інтернет-порталу Trade&Earn для торгівлі криптовалютою. Оцінити ефективність обраного підходу та сформулювати рекомендації щодо його застосування в аналогічних ІТ-проєктах.

4. Зміст роботи:

РОЗДІЛ 1 Системний аналіз проблеми управління проєктом розробки інтернет-порталу для торгівлі криптовалютою: аналіз предметної області,

концептуальна модель проєкту Trade&Earn, порівняльний аналіз методологій (Waterfall, Scrum, Kanban, гібридні підходи), маркетингове обґрунтування.

РОЗДІЛ 2 Дослідження характеристик проєкту та розробка моделей управління: аналіз зацікавлених сторін, організаційна структура команди, обґрунтування гібридної методології, розробка WBS та матриці RACI, календарне планування, розрахунок економічної ефективності (NPV, IRR).

РОЗДІЛ 3 Використання інформаційних технологій для планування та управління проєктом: календарно-сітьове планування у MS Project, ресурсне планування, управління ризиками, технічна архітектура порталу, проєктування бази даних, мокапи інтерфейсу, управління спринтами.

РОЗДІЛ 4 Практика та оцінка результатів: реалізація та аналіз спринтів, порівняння планових і фактичних показників, оцінка адекватності гібридної моделі, рекомендації щодо масштабування.

5. Перелік графічного матеріалу (слайдів)

Схема життєвого циклу гібридної методології (Waterfall + Scrum) для проєкту Trade&Earn, Ієрархічна структура робіт (WBS) проєкту Trade&Earn, Матриця «Вплив – Інтерес» стейкхолдерів проєкту Trade&Earn, Організаційна структура проєктної команди Trade&Earn, Діаграма Ганта проєкту Trade&Earn, Архітектура порталу Trade&Earn, ER-діаграма бази даних порталу Trade&Earn, Мокапи інтерфейсу порталу Trade&Earn, Блок-схема процесу реєстрації та KYC-верифікації, Блок-схема процесу створення та виконання торгового ордеру, Блок-схема процесу двофакторної автентифікації (2FA), Дошки Спринтів у Trello.

6. Календарний план виконання роботи:

№ з/п	Назва частин роботи	План виконання роботи
1	Вивчення літературних джерел з теми дослідження	05.09.2025 – 01.11.2025
2	Збір і вивчення матеріалів з теми дослідження	05.12.2025 – 10.12.2025
3	Складання розгорнутого плану кваліфікаційної роботи	11.12.2025 – 20.12.2025
4	Ознайомлення наукового керівника з розгорнутим планом кваліфікаційної роботи. Внесення змін.	20.12.2025
5	Підготовка розділу 1 «Системний аналіз проблеми управління проєктом розробки інтернет-порталу для торгівлі криптовалютою»	10.01.2026 – 25.01.2026
6	Підготовка розділу 2 «Дослідження характеристик проєкту та розробка моделей управління»	25.01.2026 – 20.02.2026
7	Підготовка розділу 3 «Використання інформаційних технологій для планування та управління проєктом»	20.02.2026 – 25.03.2026
8	Підготовка розділу 4 «Практика та оцінка результатів реалізації гібридної моделі управління проєктом»	25.03.2026 – 28.04.2026
9	Оформлення кваліфікаційної роботи	01.05.2026 – 15.05.2026

10	Передача кваліфікаційної роботи рецензенту для рецензування	15.05.2026
11	Попередній захист кваліфікаційної роботи	18.05.2026
12	Передача кваліфікаційної роботи науковому керівнику	15.05.2026
13	Захист роботи	25.05.2026

Дата видачі завдання: 5 грудня 2025 р.

Керівник роботи ____ Доцент, к.т.н Хлевний Андрій____
(посада, прізвище, ім'я)

(підпис)

Завдання прийняв до виконання студент групи УП-21

____ Гончар Максим ____
(прізвище, ім'я)

(підпис)

АНОТАЦІЯ

Кваліфікаційної роботи магістра на тему: «Дослідження процесів управління проєктом розробки інтернет-порталу Trade&Earn для торгівлі (керування) криптовалютою»

Студент: Гончар Максим Юрійович. Науковий керівник: Хлевний Андрій Олександрович. Рік захисту - 2026.

Темою даної роботи обрано «Дослідження процесів управління проєктом розробки інтернет-порталу Trade&Earn для торгівлі криптовалютою», предметною областю якої є управління ІТ-проєктами, розробка програмного забезпечення та криптовалютні фінансові платформи.

Метою підготовки роботи є розробка та апробація гібридної моделі управління проєктом (Waterfall + Scrum) на прикладі інтернет-порталу Trade&Earn для торгівлі криптовалютою, що дозволить менеджерам ІТ-проєктів забезпечити баланс між передбачуваністю планування і гнучкістю розробки в умовах регуляторних вимог.

Ціль проєкту – розробка інтернет-порталу Trade&Earn для торгівлі криптовалютою, орієнтованого на початківців та досвідчених трейдерів, з інтегрованим освітнім модулем, підтримкою гривневих операцій та відповідністю вимогам Закону України «Про віртуальні активи».

Наукова новизна роботи полягає у наступному: удосконалено підхід до застосування гібридної методології (Waterfall + Scrum) для проєктів розробки криптовалютних платформ, що, на відміну від існуючих, передбачає включення регуляторних етапів (KYC/AML) як обов'язкових пакетів робіт у каскадну фазу та визначає механізм переходу до ітеративної фази на основі зрілості архітектурних рішень. Отримало подальший розвиток обґрунтування критеріїв розмежування каскадної та ітеративної фаз у гібридній моделі управління ІТ-проєктом, що дозволяє визначити точку переходу від послідовного планування до ітеративної розробки з урахуванням ступеня визначеності вимог та готовності архітектури.

Кваліфікаційна робота складається з анотації, вступу, основної частини, яка включає чотири розділи, висновків, переліку використаних джерел.

У першому розділі проводиться системний аналіз предметної області ринку криптовалютних платформ, розглядаються тенденції розвитку ринку та регуляторне середовище в Україні. Виконується порівняльний аналіз методологій управління IT-проєктами (Waterfall, Scrum, Kanban, гібридні підходи) та обґрунтовується доцільність гібридного підходу для проєкту Trade&Earn.

Другий розділ присвячено аналізу зацікавлених сторін за моделлю «Вплив – Інтерес», розробці організаційної структури проєктної команди та матриці відповідальності RACI. Обґрунтовується вибір гібридної методології методом зваженої оцінки. Розробляється WBS та календарний план проєкту на 39 тижнів. Проводиться розрахунок економічної ефективності.

У третьому розділі розробляється календарно-сітьова модель проєкту у MS Project (49 завдань, 38 тижнів, 5 контрольних точок), проводиться ресурсне планування та управління ризиками. Проєктується мікросервісна архітектура порталу, ER-діаграма бази даних, мокапи п'яти екранів інтерфейсу та блок-схеми ключових алгоритмів обробки даних.

У четвертому розділі проводиться практична реалізація управління проєктом на стадії виконання: аналіз двох спринтів, порівняння планових і фактичних показників, оцінка адекватності гібридної моделі та рекомендації щодо її масштабування.

За результатами роботи зроблено висновки. Гібридна модель управління проєктом Trade&Earn забезпечила баланс між передбачуваністю для замовника та гнучкістю для команди розробки. Торговий модуль успішно пройшов навантажувальне тестування з результатом 120 ордерів на секунду при цільових 100. Розроблена модель рекомендована до застосування в IT-проєктах, де регуляторні вимоги поєднуються зі змінністю функціонального середовища, зокрема у сферах фінтех та державних цифрових сервісів.

Робота містить 102 сторінки, 30 рисунків та 23 таблиці. Перелік використаних джерел включає 37 найменувань.

Ключові слова: управління проєктом, IT-проєкти, криптовалюта, інтернет-портал, гібридна методологія, Waterfall, Scrum, WBS, RACI, MS Project.

ЗМІСТ

ВСТУП.....	11
РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ПРОБЛЕМИ УПРАВЛІННЯ ПРОЄКТОМ РОЗРОБКИ ІНТЕРНЕТ-ПОРТАЛУ ДЛЯ ТОРГІВЛІ КРИПТОВАЛЮТОЮ ..	15
1.1. Аналіз предметної області: ринок криптовалютних платформ та тенденції розвитку.....	15
1.2. Концептуальна модель проєкту Trade&Earn: цілі, масштаб, обмеження, стейкхолдери.....	17
1.3. Порівняльний аналіз методологій управління IT-проєктами (Waterfall, Scrum, Kanban, гібридні підходи)	21
1.4. Інструменти планування та контролю проєктів: WBS, RACI, беклог, контрольні точки.....	25
1.5. Маркетингове обґрунтування проєкту та аналіз проєктних альтернатив	27
1.6. Постановка задач дослідження	30
РОЗДІЛ 2. ДОСЛІДЖЕННЯ ХАРАКТЕРИСТИК ПРОЄКТУ ТА РОЗРОБКА МОДЕЛЕЙ УПРАВЛІННЯ	31
2.1. Аналіз зацікавлених сторін проєкту (модель «Вплив - Інтерес»).....	31
2.2. Організаційна структура проєктної команди та розподіл ролей	34
2.3. Обґрунтування вибору гібридної методології управління (Waterfall + Scrum).....	38
2.4. Розробка ієрархічної структури робіт (WBS) та матриці відповідальності (RACI)	44
2.5. Календарне планування та часові обмеження проєкту	51
2.6. Розрахунок економічної ефективності проєкту (NPV, IRR).....	53
РОЗДІЛ 3. ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ ПЛАНУВАННЯ ТА УПРАВЛІННЯ ПРОЄКТОМ.....	57

3.1. Календарно-сітьове планування у MS Project: діаграма Ганта та критичний шлях.....	57
3.2. Ресурсне планування та управління завантаженням команди	61
3.3. Управління ризиками проєкту.....	66
3.4. Технічна архітектура порталу Trade&Earn як артефакт проєктного планування.....	72
3.5. Проєктування структури бази даних порталу	75
3.6. Мокапи інтерфейсу як результат етапу UI/UX-проєктування.....	77
3.7. Блок-схеми ключових алгоритмів обробки даних	82
3.8. Управління проєктом на стадії виконання: процес спринтів, standup, ретроспективи.....	86
РОЗДІЛ 4. ПРАКТИКА ТА ОЦІНКА РЕЗУЛЬТАТІВ	90
4.1. Спринт 1: планування, реалізація, проблеми та результати ревію	90
4.2. Спринт 2: розвиток функціоналу та уточнення процесу	91
4.3. Порівняння планових і фактичних показників проєкту	92
4.4. Оцінка адекватності гібридної моделі управління	94
4.5. Рекомендації щодо масштабування гібридної моделі.....	95
ВИСНОВОК	96
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	99

ВСТУП

Сфера фінансових технологій є одним із найбільш динамічних напрямів розвитку глобальної ІТ-індустрії. Особливе місце в ній займають криптовалютні платформи, попит на які стабільно зростає як у світі, так і в Україні. Управління проектами розробки криптовалютних платформ має специфічні особливості, що відрізняють їх від типових веб-проектів. Такі платформи обробляють фінансові транзакції та зберігають конфіденційну інформацію користувачів, тому вимоги до безпеки є критичними. Водночас функціональні вимоги до продукту часто змінюються під впливом ринкової динаміки та нових регуляторних норм. Необхідність дотримання регуляторних етапів додає до проекту обов'язкові пакети робіт із жорсткою послідовністю, а координація різнопрофільної команди (бекенд, фронтенд, дизайн, безпека, бізнес-аналітика) вимагає чіткого розподілу відповідальності.

Ці фактори формують суперечливі вимоги до процесу управління, де одночасно потрібні передбачуваність і документованість з боку регулятора та замовника і гнучкість з боку ринку та команди розробки. Жодна з «класичних» методологій управління, ні каскадна модель, ні суто ітеративні фреймворки, не здатна повністю задовольнити ці суперечливі вимоги, що обґрунтовує необхідність дослідження гібридних моделей управління ІТ-проектами у сфері FinTech.

Актуальність теми обумовлена зростанням складності ІТ-проектів у сфері фінансових технологій, формуванням регуляторного середовища для криптовалютних платформ в Україні та недостатньою дослідженістю практичної ефективності гібридних моделей управління (Waterfall + Scrum) у контексті розробки криптовалютних платформ. Існуючі наукові роботи переважно розглядають гібридні підходи в загальному контексті ІТ-проектів, тоді як специфіка проектів із регуляторними вимогами KYC/AML та високою волатильністю функціональних вимог залишається недостатньо висвітленою.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконана відповідно до навчального плану підготовки магістрів за

спеціальністю 122 «Комп'ютерні науки», освітньо-наукова програма «Управління проєктами» кафедри технологій управління факультету інформаційних технологій Київського національного університету імені Тараса Шевченка.

Мета і завдання дослідження. Метою проєкту є розробка та апробація гібридної моделі управління проєктом (Waterfall + Scrum) на прикладі інтернет-порталу Trade&Earn для торгівлі криптовалютою, що дозволить менеджерам IT-проєктів забезпечити баланс між передбачуваністю планування і гнучкістю розробки в умовах регуляторних вимог, а також сформулювати рекомендації щодо використання цього підходу у подібних проєктах.

Для досягнення поставленої мети визначено такі завдання дослідження. Перше завдання полягає у проведенні системного аналізу предметної області криптовалютних платформ, визначенні факторів впливу на управління проєктами їх розробки та обґрунтуванні вибору гібридної методології. Друге завдання передбачає дослідження характеристик проєкту Trade&Earn, аналіз зацікавлених сторін, розробку організаційної структури команди, WBS, матриці RACI та календарного плану. Третє завдання стосується розрахунку економічної ефективності проєкту з використанням показників NPV та IRR. Четверте завдання спрямоване на застосування інформаційних технологій для планування та управління проєктом, зокрема побудову календарно-сітьової моделі у MS Project, ресурсне планування, розробку реєстру ризиків та проєктування технічної архітектури порталу. П'яте завдання полягає у практичній реалізації управління проєктом на стадії виконання (управління спринтами), порівнянні планових та фактичних показників, оцінці адекватності гібридної моделі та формулюванні рекомендацій.

Об'єкт дослідження. Процеси управління проєктом розробки інтернет-порталу для торгівлі криптовалютою.

Предмет дослідження. Процеси інтеграції каскадної та ітеративної фаз управління IT-проєктом в умовах регуляторних вимог та волатильності функціональних вимог (на прикладі проєкту Trade&Earn)

Методи дослідження. У роботі використано комплекс методів наукового дослідження. Системний аналіз застосовано для дослідження предметної області та визначення факторів впливу на управління проєктом. Порівняльний аналіз використано для оцінки методологій управління IT-проєктами та обґрунтування гібридного підходу. Метод декомпозиції застосовано для побудови ієрархічної структури робіт (WBS). Матричний метод використано для розробки матриці відповідальності RACI та матриці стейкхолдерів «Вплив – Інтерес». Методи інвестиційного аналізу (NPV, IRR) застосовано для розрахунку економічної ефективності проєкту. Метод критичного шляху використано для календарно-сітьового планування у MS Project. Якісну оцінку ризиків застосовано для формування реєстру ризиків та стратегій реагування. Аналіз фактичних даних використано для порівняння планових і фактичних показників виконання спринтів.

Наукова новизна отриманих результатів. Удосконалено підхід до застосування гібридної методології (Waterfall + Scrum) для проєктів розробки криптовалютних платформ, що, на відміну від існуючих, передбачає включення регуляторних етапів (KYC/AML) як обов'язкових пакетів робіт у каскадну фазу та визначає механізм переходу до ітеративної фази на основі зрілості архітектурних рішень. Отримало подальший розвиток обґрунтування критеріїв розмежування каскадної та ітеративної фаз у гібридній моделі управління IT-проєктом, що дозволяє визначити точку переходу від послідовного планування до ітеративної розробки з урахуванням ступеня визначеності вимог та готовності архітектури.

Практичне значення отриманих результатів. Результати дослідження апробовані на проєкті розробки інтернет-порталу Trade&Earn. Розроблені інструменти управління проєктом (WBS, RACI, календарний план у MS Project, реєстр ризиків) та рекомендації щодо організації гібридного процесу

мають ступінь готовності, достатній для безпосереднього використання при управлінні аналогічними ІТ-проєктами у сфері FinTech.

Апробація результатів роботи. Основні положення та результати дослідження доповідалися та обговорювалися на XI Міжнародній конференції «Information Technology and Implementation (IT&I-2025)» (21 листопада 2025 р., м. Київ, Київський національний університет імені Тараса Шевченка).

Публікації. Гончар М.Ю., Хлевний А.О. Development of an Internet Portal for Trading Education with Blockchain Technology Integration and Tokenized Reward System // Збірник тез доповідей XI Міжнародної конференції «Information Technology and Implementation (IT&I-2025)». – Київ: Видавничий дім «Каравела», 2025. – С. 232. – ISBN 978-960-801-928-7.

Структура та обсяг роботи. Кваліфікаційна робота складається зі вступу, анотації, чотирьох розділів, висновків, переліку використаних джерел та додатків. Обсяг становить 102 сторінки, робота містить 21 рисуноків, 37 таблиць. Перелік використаних джерел включає 37 найменувань.

РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ПРОБЛЕМИ УПРАВЛІННЯ ПРОЄКТОМ РОЗРОБКИ ІНТЕРНЕТ-ПОРТАЛУ ДЛЯ ТОРГІВЛІ КРИПТОВАЛЮТОЮ

1.1. Аналіз предметної області: ринок криптовалютних платформ та тенденції розвитку

Ринок криптовалютних платформ за останнє десятиліття пройшов шлях від нішевого явища до повноцінного сегмента глобальної фінансової індустрії. За даними аналітичної платформи CoinGecko, станом на початок 2025 року у світі функціонувало понад 750 криптовалютних бірж, а загальна кількість зареєстрованих користувачів перевищила 560 мільйонів осіб [1]. При цьому близько 80% обсягу торгів зосереджено на двадцяти найбільших платформах, що свідчить про високий рівень конкуренції та одночасно про наявність простору для нових учасників у вузьких нішах ринку.

Україна стабільно входить до десятки країн світу за рівнем прийняття криптовалют, якщо вимірювати його відношенням обсягу криптотранзакцій до ВВП. За оцінками Statista, у 2025 році кількість українців, які хоча б раз здійснювали операції з криптовалютою, перевищила 5 мільйонів [2]. Цей показник значно зріс після 2022 року, коли криптовалюти використовувались як інструмент збереження заощаджень в умовах воєнної нестабільності.

Важливим фактором розвитку ринку стало прийняття Закону України «Про віртуальні активи» (№ 2074-IX від 17 лютого 2022 року), який визначив правовий статус віртуальних активів та встановив вимоги до постачальників відповідних послуг [3]. Закон зобов'язує платформи впроваджувати процедури ідентифікації клієнтів (KYC) та протидії відмиванню коштів (AML), забезпечувати зберігання і захист персональних даних, а також проходити реєстрацію у Національній комісії з цінних паперів та фондового ринку (НКЦПФР) [9]. Для проєктів розробки нових криптовалютних платформ це означає, що регуляторні вимоги з самого початку мають бути враховані у плануванні, закладені в обсяг робіт та відповідно оцінені у бюджеті і термінах.

Управління проєктом розробки криптовалютної платформи має свої характерні особливості, що відрізняють його від типових веб-проєктів. Перша з них стосується безпеки. Платформа обробляє фінансові транзакції та зберігає конфіденційну інформацію користувачів, тому навіть одна вразливість може призвести до прямих грошових втрат. На практиці це означає, що процеси тестування, код-рев'ю та аудиту безпеки мають бути не факультативними перевірками, а обов'язковими етапами, вбудованими у календарний план проєкту.

Друга особливість пов'язана з нестабільністю вимог. Крипторинок змінюється значно швидше за більшість інших фінансових ринків. Нові регуляторні норми, зміни у поведінці користувачів, поява конкурентних рішень можуть вимагати коригування функціональних вимог безпосередньо під час розробки. Жорстке планування усього обсягу робіт наперед, без можливості їх перегляду, у таких умовах створює ризик отримати на виході продукт, що вже не відповідає реаліям ринку.

Третя особливість стосується складу проєктної команди. Розробка крипто-порталу потребує спеціалістів з бекенд-розробки, фронтенд-розробки, UI/UX-дизайну, інформаційної безпеки, а також людей, які розуміють вимоги фінансового регулювання. Координація такої різнопрофільної команди потребує чіткого розподілу відповідальності та налагоджених комунікаційних процесів, інакше виникають ситуації, коли задачі «провисають» між спеціалістами або дублюються.

Отже, предметна область формує три групи вимог до того, як потрібно управляти подібним проєктом. Перша група визначається регуляторним середовищем і вимагає документованості та послідовності етапів. Друга група зумовлена ринковою динамікою і вимагає гнучкості та можливості змінювати пріоритети. Третя група впливає зі складу команди і вимагає чіткої оргструктури та регулярної комунікації. Ці вимоги частково суперечать одна одній, що обґрунтовує необхідність гібридного підходу до управління(рис. 1.1).

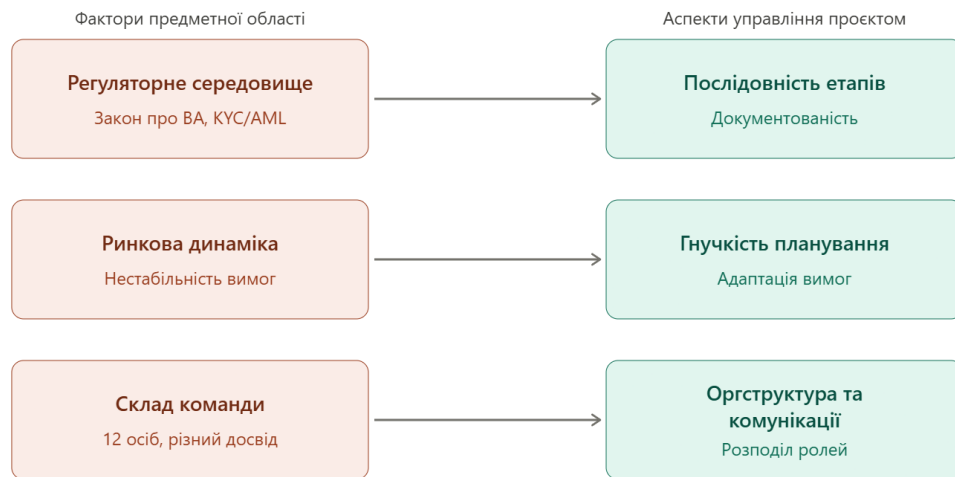


Рис. 1.1. Фактори впливу предметної області на управління проектом розробки інтернет-порталу для торгівлі криптовалютою

Таким чином, предметна область проекту формує три взаємопов'язані групи вимог до системи управління, кожна з яких окремо є керованою, але у сукупності вони створюють суперечливе середовище, де одночасно необхідні документованість і гнучкість. Саме ця суперечність визначає актуальність пошуку збалансованої методологічної основи для управління проектом Trade&Earn, що розглядається у наступних підрозділах.

1.2. Концептуальна модель проекту Trade&Earn: цілі, масштаб, обмеження, стейкхолдери

Об'єктом дослідження у даній роботі є проект Trade&Earn, ініційований ІТ-компанією ТОВ «ТрейдСофт», що спеціалізується на розробці програмних продуктів. Компанія налічує 30 штатних працівників, з яких 18 зайняті у розробці. Результатом проекту є інтернет-портал для торгівлі криптовалютою.

Мета проекту: дослідити та застосувати гібридну модель управління (Waterfall + Scrum) при організації розробки інтернет-порталу Trade&Earn, оцінити її ефективність та сформулювати рекомендації щодо застосування у подібних ІТ-проектах.

Об'єкт дослідження: процеси управління проектом розробки інтернет-порталу для торгівлі криптовалютою.

Предмет дослідження: гібридна методологія управління проектом (Waterfall + Scrum) та її вплив на ефективність реалізації проекту.

Проект має три основні обмеження, які у практиці управління описуються моделлю Triple Constraint (рис. 1.2) [4].

Часове обмеження: тривалість проекту складає 9 місяців від ініціації до запуску першої версії порталу. Вартісне обмеження: бюджет зафіксований на рівні 2,5 мільйони гривень і включає витрати на оплату праці, інфраструктуру, ліцензії та тестування. Обмеження за обсягом: у першу версію (MVP) входить лише базовий набір функцій, розширений функціонал відкладається на наступні ітерації. Ці три параметри взаємопов'язані: зміна одного неминуче впливає на інші. Якщо, наприклад, під час розробки з'являється нова регуляторна вимога, її реалізація збільшує обсяг робіт, що потребує або продовження терміну, або збільшення бюджету, або скорочення іншого функціоналу.

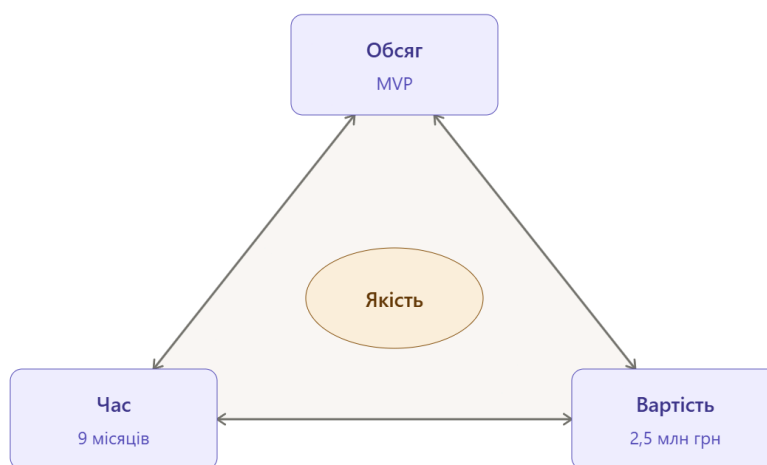


Рис. 1.2. Модель «потрійного обмеження» (Triple Constraint) проекту
Trade&Earn

Команда проекту складається з 12 осіб: менеджер проекту, чотири бекенд-розробники, два фронтенд-розробники, UI/UX-дизайнер, два тестувальники, DevOps-спеціаліст та бізнес-аналітик. Частина команди має досвід роботи за Scrum, частина звикла до каскадного підходу. Ця

неоднорідність досвіду впливає на вибір методології та спосіб організації роботи.

Зацікавлені сторони (стейкхолдери) проєкту розподілені за рівнем впливу та інтересу. Замовник проєкту (CEO компанії «ТрейдСофт») затверджує бюджет, визначає пріоритети і очікує чіткого плану з контрольними точками. Його вплив та інтерес до проєкту високі. Проєктна команда безпосередньо виконує роботу, має високий вплив і високий інтерес, при цьому зацікавлена в гнучкості процесу для самостійного прийняття технічних рішень.

Кінцеві користувачі (трейдери, інвестори-початківці) мають високий інтерес до продукту, але їхній прямий вплив на процес розробки обмежений. Вони не беруть участі у щоденній роботі, проте їхній зворотний зв'язок на демонстраціях (Sprint Review) впливає на пріоритизацію беклогу.

Регулятор (НКЦПФР) має високий вплив, оскільки його вимоги додають до проєкту обов'язкові пакети робіт (впровадження KYC/AML), але інтерес до конкретного проєкту помірний до моменту подання заявки на реєстрацію.

Зовнішні підрядники (провайдер хостингу, постачальник API котирувань) мають обмежений вплив і помірний інтерес, проте від їхньої надійності залежить частина архітектурних рішень.

Аналіз характеристик проєкту показує суперечливість вимог до управління. Замовник очікує передбачуваності: фіксований план, контрольні точки, детальна документація. Команда потребує гнучкості для самостійного прийняття технічних рішень. Регулятор додає обов'язкові етапи з жорсткою послідовністю, а ринок вимагає швидкого реагування на зміни. Ці суперечності визначають необхідність пошуку збалансованого підходу до управління проєктом.

Для формалізації підходу до управління проєктом в умовах визначених обмежень розроблено концептуальну модель гібридної методології управління проєктом Trade&Earn, що відображає повний цикл від вхідних факторів впливу до результатів проєкту (рис. 1.3).

Модель структурована за трьома рівнями. На рівні входів визначено три групи факторів впливу на проєкт, ринкові вимоги (попит, конкуренти), регуляторні вимоги (KYC/AML, Закон України «Про віртуальні активи» №2074-IX) та вимоги стейкхолдерів (замовник і команда). Саме ці фактори породжують суперечливе управлінське середовище, описане вище.

На рівні процесу модель розподіляє управлінські дії на дві послідовні фази. Каскадна фаза (Waterfall) охоплює ініціацію проєкту, планування (WBS, RACI, MS Project) та проєктування (архітектура, база даних, мокапи). Після досягнення milestone KT-2 та затвердження архітектурних рішень проєкт переходить до ітеративної фази (Scrum), яка реалізується у вигляді спринтів 1-10 з регулярними оглядами та коригуванням беклогу.

На рівні виходів модель передбачає три групи результатів. Функціональний MVP порталу Trade&Earn, повний комплект проєктної документації (WBS, реєстр ризиків, календарний план у MS Project) та рекомендації щодо масштабування гібридної моделі.



Рис. 1.3. Концептуальна модель гібридної методології управління проєктом Trade&Earn

Таким чином, концептуальна модель демонструє, що гібридна методологія є не механічним поєднанням двох підходів, а структурованою відповіддю на суперечливі вимоги предметної області. Каскадна фаза забезпечує передбачуваність та документованість для замовника і регулятора, тоді як ітеративна фаза надає команді розробки необхідну гнучкість для адаптації до змін функціональних вимог.

1.3. Порівняльний аналіз методологій управління ІТ-проєктами (Waterfall, Scrum, Kanban, гібридні підходи)

Вибір методології управління є одним з перших рішень, яке приймається при запуску ІТ-проєкту. Від нього залежить, як буде організована робота команди, як відбуватиметься комунікація із замовником і як проєкт реагуватиме на зміни. На практиці найчастіше застосовуються чотири підходи, кожен з яких має свої сильні та слабкі сторони.

Каскадна модель (Waterfall) передбачає послідовне виконання етапів проєкту: спочатку збираються усі вимоги, потім виконується проєктування, далі реалізація, тестування і впровадження. Перехід до наступного етапу відбувається лише після завершення попереднього [5].

Головна перевага Waterfall полягає у тому, що замовник на старті отримує чіткий план з визначеними термінами і бюджетом. Усі вимоги фіксуються на початку, що спрощує оцінку обсягу робіт та контроль прогресу. Проте саме в цьому полягає і обмеження моделі. За різними оцінками, від 35 до 50 відсотків вимог до програмного забезпечення змінюються протягом проєкту [5]. Якщо замовник бачить працюючий продукт лише наприкінці, існує ризик, що результат не відповідатиме його очікуванням, і вносити зміни буде вже дорого.

Scrum є ітеративним фреймворком, що організує роботу у спринтах, тривалість яких зазвичай складає від одного до чотирьох тижнів. Кожен спринт завершується демонстрацією працюючого інкременту продукту [6]. У Scrum визначено три ролі: Product Owner формує та пріоритизує список завдань

(беклог), Scrum Master забезпечує дотримання процесу та допомагає команді усувати перешкоди, а команда розробки самоорганізовується для виконання завдань спринту. Обов'язковими подіями є планування спринту, щоденний стендап тривалістю до 15 хвилин, огляд спринту (Sprint Review) та ретроспектива [6].

Основна перевага Scrum полягає у короткому циклі зворотного зв'язку. Замовник бачить результат кожні 2-4 тижні і може скоригувати напрямок розробки. Обмеження Scrum проявляються у тому, що фреймворк не дає готових інструментів для довгострокового планування бюджету і термінів. Замовнику, який звик до каскадного підходу, буває складно прийняти, що детальний план існує лише на найближчий спринт.

Метод Kanban, на відміну від Scrum, не використовує фіксовані ітерації. Робота організована як безперервний потік завдань, а головним механізмом контролю є обмеження кількості задач, що виконуються одночасно (WIP-ліміти) [7]. Робочий процес візуалізується на дошці зі стовпцями (наприклад, «Заплановано», «У роботі», «Готово»). Kanban не визначає конкретних ролей чи подій, що робить його гнучким, але водночас менш структурованим. Для проєкту розробки нового продукту з фіксованим дедлайном Kanban зазвичай не використовується як самостійний підхід, оскільки він не дає механізмів планування релізу.

Гібридні методології виникають тоді, коли жоден з «чистих» підходів не задовольняє усіх потреб проєкту. Ідея полягає у тому, щоб поєднати сильні сторони різних методологій та компенсувати їхні обмеження [10]. Найпоширенішим варіантом є послідовний гібрид, у якому ранні фази проєкту (ініціація, збір вимог, архітектурне проєктування) виконуються за каскадною моделлю, а фаза реалізації організовується за Scrum. Такий підхід дозволяє замовнику на початку отримати затверджену архітектуру та зафіксований план, а команді на етапі розробки мати гнучкість для ітеративного уточнення деталей.

Для того щоб обґрунтувати вибір методології для конкретного проєкту, потрібно порівняти підходи за критеріями, що є релевантними для цього проєкту. Порівняльна характеристика розглянутих методологій наведена у таблиці 1.1.

Таблиця 1.1.

Порівняльна характеристика методологій управління ІТ-проєктами

Критерій	Waterfall	Scrum	Kanban	Гібридна (Waterfall Scrum) +
1	2	3	4	5
Стабільність вимог	Стабільні, фіксуються на старті	Змінні, уточнюються щоспринту	Будь-які	Стабільні на рівні архітектури, змінні на рівні функцій
Передбачуваність термінів	Висока	Середня	Низька	Висока на рівні фаз, середня на рівні спринтів
Гнучкість до змін	Низька	Висока	Висока	Середня
Зворотний зв'язок від замовника	На початку та наприкінці	Кожні 2–4 тижні	Постійно	На контрольних точках та Sprint Review
Рівень документації	Детальна	Мінімально достатня	Мінімальна	Детальна для ранніх фаз, легка для спринтів
Придатність для проєктів з регуляторними вимогами	Так (послідовність етапів)	Частково	Ні	Так

Аналіз таблиці показує, що для проєкту Trade&Earn жодна з «чистих» методологій не підходить повністю. Каскадна модель забезпечує передбачуваність і дотримання регуляторних вимог, але не дозволяє адаптувати функціональні вимоги під час розробки. Scrum дає гнучкість, але не задовольняє потребу замовника у фіксованому плані на ранніх фазах. Kanban не призначений для проєктів з визначеним обсягом та дедлайном. Гібридна методологія, що поєднує каскадний підхід на етапах ініціації та проєктування зі Scrum на етапі реалізації, виглядає як найбільш збалансоване рішення(рис. 1.4).

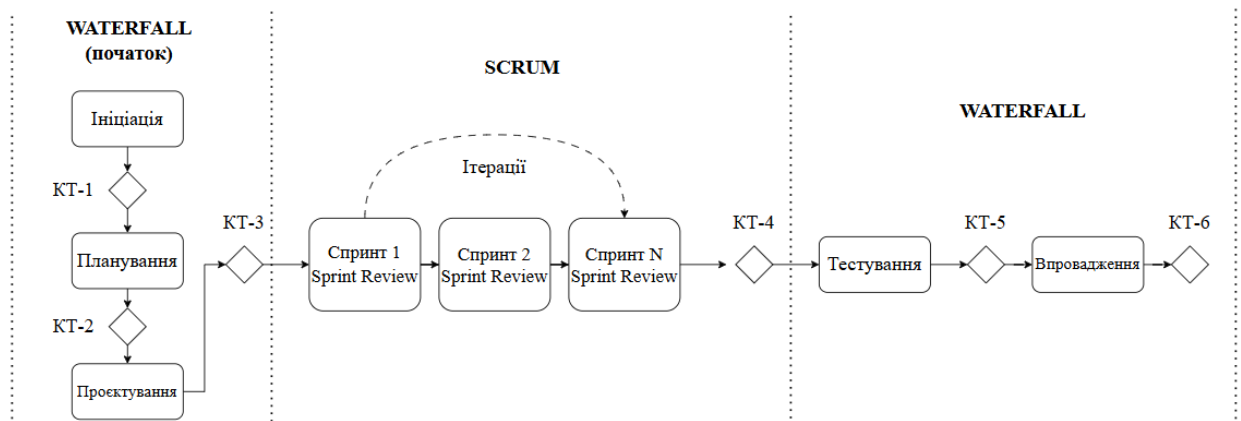


Рис. 1.4. Схема життєвого циклу гібридної методології (Waterfall + Scrum) для проєкту Trade&Earn

Як видно зі схеми, гібридна методологія не є механічним поєднанням двох підходів, а передбачає чітко визначену точку переходу між фазами: момент затвердження архітектури на КТ-2 стає кордоном між послідовним плануванням і ітеративною розробкою. Саме наявність такої точки переходу та формальних критеріїв її досягнення відрізняє запропонований підхід від інтуїтивного «змішування» методологій і визначає його наукову та практичну цінність для проєкту Trade&Earn.

1.4. Інструменти планування та контролю проєктів: WBS, RACI, беклог, контрольні точки

Незалежно від обраної методології, менеджер проєкту потребує конкретних інструментів, щоб розбити проєкт на частини, розподілити відповідальність, визначити пріоритети та відстежувати прогрес. У цьому підрозділі описано чотири інструменти, що будуть використані для управління проєктом Trade&Earn.

Ієрархічна структура робіт (Work Breakdown Structure, або WBS) розбиває увесь обсяг проєкту на менші, більш керовані частини [8]. На верхньому рівні WBS зазвичай відповідає фазам проєкту, на другому рівні фази деталізуються на пакети робіт, а на третьому та четвертому рівнях визначаються конкретні завдання, які можна оцінити за часом і вартістю та призначити відповідальним. WBS одночасно виконує кілька функцій. По-перше, це основа для побудови календарного плану та оцінки трудомісткості. По-друге, WBS допомагає усім учасникам проєкту однаково розуміти обсяг того, що потрібно зробити. По-третє, WBS дозволяє відстежувати прогрес на кожному рівні: якщо завершено 3 з 5 пакетів робіт другого рівня, то прогрес можна оцінити як 60%.

Для проєкту Trade&Earn WBS будуватиметься на 3–4 рівні. Перший рівень включає шість фаз: ініціація, планування, проєктування, розробка, тестування та впровадження. Другий рівень деталізує кожну фазу. Наприклад, фаза «Розробка» розбивається на пакети «Розробка бекенду», «Розробка фронтенду», «Інтеграція API» та «Модуль безпеки» (рис. 1.5).



Рис. 1.5. Верхні рівні ієрархічної структури робіт (WBS) проєкту Trade&Earn

Матриця RACI зв'язує елементи WBS з учасниками проєкту і для кожної комбінації «завдання - учасник» визначає тип залученості [8]. R (Responsible) означає людину, яка безпосередньо виконує роботу. A (Accountable) визначає того, хто відповідає за результат і приймає рішення. C (Consulted) позначає тих, з ким радяться перед прийняттям рішень. I (Informed) визначає тих, кого повідомляють про результат. У проєкті Trade&Earn, де одна й та сама людина може на різних етапах виконувати різні функції, матриця RACI допомагає уникнути двох типових проблем: коли за певне завдання ніхто не відповідає, або коли відповідають одразу кілька осіб і незрозуміло, хто приймає фінальне рішення.

Беклог продукту (Product Backlog) є центральним артефактом Scrum-частини гібридної методології. Це впорядкований список усього, що потрібно реалізувати у продукті [6]. Кожен елемент беклогу зазвичай описується як User Story, тобто формулюється з точки зору кінцевого користувача. Наприклад: «Як трейдер, я хочу бачити графік зміни курсу BTC/UAN за останні 24 години, щоб приймати рішення про купівлю». Беклог не є статичним документом. Product Owner постійно переглядає та переприоритизовує його елементи на основі зворотного зв'язку від замовника, результатів демонстрацій та змін ринкових умов. Для проєкту Trade&Earn беклог формується на етапі планування і використовується як основне джерело завдань для спринтів.

Контрольні точки (milestones) позначають завершення значимих етапів проєкту. У гібридній методології вони виконують роль з'єднувальної ланки між каскадною та ітеративною частинами процесу. На фазах ініціації та проєктування контрольні точки працюють як класичні Waterfall-гейти: замовник перевіряє та затверджує результати перед тим, як команда переходить до наступного етапу. На фазі виконання роль контрольних точок частково виконують Sprint Review, на яких демонструється інкремент продукту.

Для проєкту Trade&Earn визначено п'ять контрольних точок. КТ-1 позначає завершення ініціації та затвердження статуту проєкту. КТ-2 фіксує завершення проєктування та затвердження архітектури. КТ-3 відповідає завершенню першого циклу спринтів та появі працюючого прототипу. КТ-4 означає готовність продукту до фінального тестування (Code Freeze). КТ-5 позначає успішне проходження приймального тестування та готовність до запуску.

Ці чотири інструменти не існують ізольовано. WBS є основою, на якій будується матриця RACI та календарний план. Елементи WBS з фази реалізації трансформуються у беклог для спринтів. Контрольні точки прив'язуються до конкретних рівнів WBS. Разом ці інструменти формують цілісну систему планування та контролю, яка пов'язує каскадну та ітеративну частини процесу.

1.5. Маркетингове обґрунтування проєкту та аналіз проєктних альтернатив

Перш ніж планувати управління проєктом, потрібно обґрунтувати доцільність його реалізації. Рішення про запуск Trade&Earn базується на аналізі трьох ринкових факторів.

Перший фактор полягає у зростанні криптовалютної аудиторії в Україні. Як зазначалось у підрозділі 1.1, кількість українців, що використовують криптовалюту, перевищила 5 мільйонів, і ця цифра продовжує зростати [2]. Зі

зростанням аудиторії зростає і попит на зручні платформи з локалізованим інтерфейсом та підтримкою гривневих операцій.

Другий фактор стосується регуляторної готовності. Прийняття Закону «Про віртуальні активи» [3] створило правове поле для легальних криптовалютних сервісів в Україні. Платформа, яка від початку розробляється з урахуванням вимог цього Закону, має перевагу перед міжнародними конкурентами, що адаптують свої глобальні рішення під українські норми «заднім числом».

Третій фактор пов'язаний з бар'єром входу для нових користувачів. Більшість діючих платформ орієнтовані на досвідчених трейдерів і мають переважаний інтерфейс з великою кількістю графіків, індикаторів та налаштувань. Для людини, яка вперше стикається з криптовалютами, це створює відчуття складності. Trade&Earn позиціонується як платформа, що поєднує торгові функції з освітнім компонентом (навчальні матеріали, демо-рахунок), і орієнтується саме на цей сегмент.

Серед прямих конкурентів на українському ринку працюють платформи Kuna та WhiteBIT. Kuna, заснована у 2017 році, однією з перших в Україні забезпечила гривневі депозити через банківські перекази та підтримку картувань. WhiteBIT, заснована у 2018 році, позиціонує себе як європейську біржу з українським корінням і пропонує ширший набір торгових інструментів, включаючи ф'ючерси та стейкінг. Обидві платформи орієнтовані переважно на користувачів з певним досвідом і не мають розвиненого освітнього модуля. Binance, як глобальний лідер, пропонує найнижчі комісії (0,1% за спотову торгівлю), але її правовий статус в Україні залишається невизначеним, а інтерфейс містить десятки розділів, що ускладнює адаптацію для новачків.

Таблиця 1.2.

**Конкурентний аналіз криптовалютних платформ, доступних
українським користувачам**

Критерій	Kuna	WhiteBIT	Binance	Trade&Earn (плановано)
1	2	3	4	5
Підтримка гривні	Так	Так	Обмежено	Так
Україномовний інтерфейс	Так	Частково	Так	Так
Освітній модуль	Немає	Базовий	Академія (окремо)	Інтегрований
Орієнтація	Досвідчені	Середні та досвідчені	Усі рівні	Початківці та середні
Відповідність Закону №2074-IX	Проходить реєстрацію	Проходить реєстрацію	Не зареєстрована	Закладено від початку

При ініціації проєкту розглядалися три альтернативи реалізації. Перша передбачала придбання та адаптацію існуючого рішення з відкритим кодом. Перевагою є швидкий старт, але обмежена гнучкість та залежність від стороннього коду створюють ризики для безпеки та подальшого розвитку. Друга альтернатива полягала у повній розробці з нуля, що дає максимальний контроль над архітектурою, але потребує суттєво більшого бюджету та часу. Обрано третю альтернативу, що поєднує використання перевірених бібліотек та API для базової торгової інфраструктури із власною розробкою інтерфейсу, освітнього модуля та бізнес-логіки. Цей підхід забезпечує баланс між швидкістю, контролем та безпекою.

1.6. Постановка задач дослідження

Аналіз, проведений у підрозділах 1.1–1.5, дозволив сформулювати проблему дослідження. Управління проєктом розробки інтернет-порталу для торгівлі криптовалютою потребує методології, яка одночасно забезпечує передбачуваність для замовника і гнучкість для команди розробки. Ці вимоги частково суперечать одна одній. Каскадна модель забезпечує передбачуваність, але не дозволяє адаптувати вимоги під час розробки. Scrum надає гнучкість, але не задовольняє потребу замовника у фіксованому плані. Гібридні підходи вже застосовуються на практиці, проте їх використання у проєктах розробки криптовалютних платформ недостатньо досліджене та формалізоване.

Для досягнення мети роботи визначено такі завдання дослідження:

- провести системний аналіз предметної області управління проєктами розробки інтернет-порталів для торгівлі криптовалютою та виявити специфічні вимоги до методології управління;
- дослідити характеристики проєкту Trade&Earn, побудувати ієрархічну структуру робіт та матрицю відповідальності, обґрунтувати вибір гібридної методології (Waterfall + Scrum) на основі формалізованих критеріїв;
- розробити календарно-сітьову модель проєкту у MS Project, провести ресурсне планування та моделювання ризиків, спроектувати технічну архітектуру порталу та структуру бази даних як артефакти проєктного планування;
- змодельовати практичне застосування гібридної моделі на прикладі двох спринтів та оцінити адекватність запропонованої моделі шляхом порівняння планових і фактичних показників проєкту.

РОЗДІЛ 2. ДОСЛІДЖЕННЯ ХАРАКТЕРИСТИК ПРОЄКТУ ТА РОЗРОБКА МОДЕЛЕЙ УПРАВЛІННЯ

2.1. Аналіз зацікавлених сторін проєкту (модель «Вплив - Інтерес»)

Управління зацікавленими сторонами в проєкті Trade&Earn ґрунтується на моделі визначення впливу та інтересу. Вона дозволяє класифікувати стейкхолдерів та сформувані цільові стратегії взаємодії.

Визначення очікувань кожної групи є критичним кроком. Розбіжності у вимогах до управління безпосередньо впливають на вибір методології. Модель оцінює стейкхолдерів за здатністю змінювати проєктні рішення та ступенем зацікавленості у результатах. На основі цього формуються чотири ключові стратегії взаємодії. До них належать тісна співпраця, моніторинг, інформування та мінімальний контроль(рис. 2.1)[11].

Результати аналізу зацікавлених сторін проєкту Trade&Earn наведено у таблиці 2.1.

Таблиця 2.1.

Реєстр зацікавлених сторін проєкту Trade&Earn

Стейкхолдер	Вплив	Інтерес	Очікування	Стратегія управління
1	2	3	4	5
Замовник (СЕО ТОВ «ТрейдСофт»)	Високий	Високий	Чіткий план, дотримання бюджету та термінів, регулярна звітність, контрольні точки	Тісна співпраця, залучення до затвердження контрольних точок, щотижневі статус-звіти, участь у Sprint Review

1	2	3	4	5
Проектна команда (12 осіб)	Високий	Високий	Зрозумілі вимоги, автономія у технічних рішеннях, стабільний процес, відсутність мікроменеджменту	Тісна співпраця, самоорганізація в межах спринтів, участь у плануванні, прозорий беклог
Кінцеві користувачі (трейдери, початківці)	Низький	Високий	Зручний інтерфейс, безпека коштів, освітні матеріали, підтримка гривні	Інформування, залучення до Sprint Review для зворотного зв'язку, бета-тестування
Регулятор (НКЦПФР)	Високий	Низький	Дотримання вимог Закону «Про віртуальні активи», впровадження KYC/AML	Моніторинг, відстеження змін у вимогах, забезпечення відповідності на етапі проєктування
Зовнішні підрядники (хостинг, API)	Низький	Низький	Своєчасна оплата, чіткі технічні вимоги, стабільне навантаження	Мінімальний контроль, фіксація вимог у контрактах, моніторинг SLA

Аналіз таблиці виявляє ключовий управлінський конфлікт між очікуваннями двох стейкхолдерів з високим впливом та високим інтересом.

Замовник вимагає жорсткої передбачуваності. Він очікує побачити затверджений план на весь проєкт, чіткі контрольні точки та щотижневу звітність. Такі вимоги є характерними для каскадної моделі. Натомість проєктна команда потребує гнучкості. Розробники хочуть самостійно декомпонувати завдання, адаптувати технічні рішення в процесі роботи та діяти короткими ітераціями. Цей підхід відповідає принципам Scrum.

Згідно з дослідженнями PMI, подібна розбіжність між методологічними очікуваннями фіксується майже у половині ІТ-проєктів [12]. Зазвичай цю проблему ігнорують і змушують одну зі сторін працювати у некомфортних умовах. Це часто призводить до мікроменеджменту або до повної втрати контролю над процесом. Для проєкту Trade&Earn вирішення цього конфлікту стає головною причиною вибору гібридної моделі управління. Вона здатна задовольнити вимоги обох ключових стейкхолдерів без примусової адаптації жодної зі сторін.

Регулятор в особі НКЦПФР має високий вплив, але низький інтерес. Його вплив проявляється через жорсткий набір обов'язкових законодавчих вимог. Стратегія управління цим стейкхолдером передбачає інтеграцію регуляторних вимог безпосередньо у структуру робіт ще на етапі проєктування. Прикладом таких вимог є впровадження процедур KYC та AML. Кінцеві користувачі мають високий інтерес, але низький вплив. Вони допомагають формувати пріоритети завдань через зворотний зв'язок наприкінці кожного спринту. Взаємодія із зовнішніми підрядниками обмежується виконанням контрактних зобов'язань та моніторингом якості надання послуг.



Рис. 2.1. Матриця «Вплив - Інтерес» стейкхолдерів проєкту Trade&Earn із зонами стратегій управління

Матриця «Вплив - Інтерес» дозволяє визначити стратегію взаємодії з кожним учасником проєкту залежно від його позиції у двовимірному просторі. Учасники з високим рівнем обох параметрів потребують постійної залученості до прийняття рішень, тоді як для решти достатньо регулярного інформування або моніторингу змін у їхньому середовищі. Такий підхід дозволяє керівнику проєкту раціонально розподіляти час і зусилля на комунікацію, зосереджуючи увагу там, де це справді впливає на результат.

2.2. Організаційна структура проєктної команди та розподіл ролей

Організаційна структура команди Trade&Earn побудована за принципом проєктної матриці. Її головна мета полягає в адаптації колективу до гібридної моделі управління. Ця структура забезпечує чітку ієрархію для стратегічного планування на старті та дозволяє перейти до самоорганізації під час розробки. Команда складається з 12 фахівців. Кожен з них поєднує технічну

спеціалізацію з конкретною проєктною відповідальністю. Склад команди наведено у таблиці 2.2.

Таблиця 2.2.

Склад проєктної команди Trade&Earn

Роль	Кількість	Основні функції	Методологічний досвід
1	2	3	4
Менеджер проєкту (PM)	1	Загальне управління, планування, контроль бюджету та термінів, комунікація із замовником, управління ризиками	Waterfall, базовий Scrum
Бекенд-розробник	4	Розробка серверної частини, API, бізнес-логіка, інтеграція з платіжними системами	2 - Scrum, 2 - Waterfall
Фронтенд-розробник	2	Розробка клієнтської частини, інтерфейс користувача, адаптивна верстка	Scrum
UI/UX-дизайнер	1	Проектування інтерфейсу, створення мокапів, прототипування, UX-дослідження	Kanban
Тестувальник (QA)	2	Функціональне та регресійне тестування, тестування безпеки, автоматизація тестів	1 - Scrum, 1 - Waterfall
DevOps-спеціаліст	1	Налаштування CI/CD, інфраструктура, моніторинг, розгортання середовищ	Scrum
Бізнес-аналітик (BA)	1	Збір та формалізація вимог, написання User Stories, комунікація із замовником	Waterfall

Неоднорідний досвід учасників формує складний управлінський виклик. Половина спеціалістів звикла працювати за гнучкими методологіями, тоді як інша половина має досвід роботи виключно з каскадними підходами. Використання єдиного формату роботи неминуче призведе до зниження продуктивності частини колективу. Гібридна модель розв'язує цю проблему через динамічну трансформацію ролей під час переходу між фазами. Це допомагає зберегти звичний стиль роботи для кожної групи спеціалістів.

На етапах ініціації, планування та проєктування встановлюється ієрархічна структура. Менеджер проєкту виконує класичні управлінські функції. Він розподіляє ресурси, будує календарний план та контролює дотримання термінів. Бізнес-аналітик відповідає за формалізацію вимог та підготовку архітектурної документації. З початком фази розробки структура стає горизонтальною. Менеджер проєкту бере на себе функції Scrum Master для допомоги команді в усуненні перешкод. Бізнес-аналітик стає Product Owner, оскільки саме він найкраще розуміє первинні вимоги замовника (рис 2.2). Детальний розподіл ролей наведено у таблиці 2.3.

Таблиця 2.3.

Розподіл ролей за фазами проєкту

Учасник	Каскадні фази (Ініціація - Проектування)	Scrum-фаза (Розробка)	Каскадні фази (Тестування - Впровадження)
1	2	3	4
Менеджер проєкту	РМ, планування, контроль, звітність	Scrum Master, фасилітація, усунення перешкод	РМ, контроль якості, координація запуску
Бізнес- аналітик	Збір вимог, специфікація	Product Owner, пріоритизація беклогу	Приймальне тестування, документація

Продовження табл. 2.3.

1	2	3	4
Бекенд-розробники	Участь у проєктуванні архітектури	Команда розробки, самоорганізація	Виправлення дефектів, оптимізація
Фронтенд-розробники	Участь у проєктуванні UI	Команда розробки, самоорганізація	Виправлення дефектів
UI/UX-дизайнер	Створення мокапів, прототипів	Підтримка, уточнення дизайну для User Stories	Фінальне полірування UI
Тестувальники	Розробка тест-плану	Тестування інкрементів наприкінці кожного спринту	Регресійне та приймальне тестування
DevOps	Налаштування середовищ розробки	Підтримка CI/CD, автоматичне розгортання	Налаштування продакшн-середовища

Наведений розподіл ролей підтверджує, що гібридна модель вимагає від кожного учасника команди методологічної гнучкості: спеціалісти, які звикли до каскадного підходу, залучаються у повному обсязі на фазах планування та проєктування, а з початком ітеративної розробки природно інтегруються у Scrum-процес без примусової перекваліфікації. Організаційну структуру команди з розподілом відповідальності за фазами наведено на рисунку 2.2.

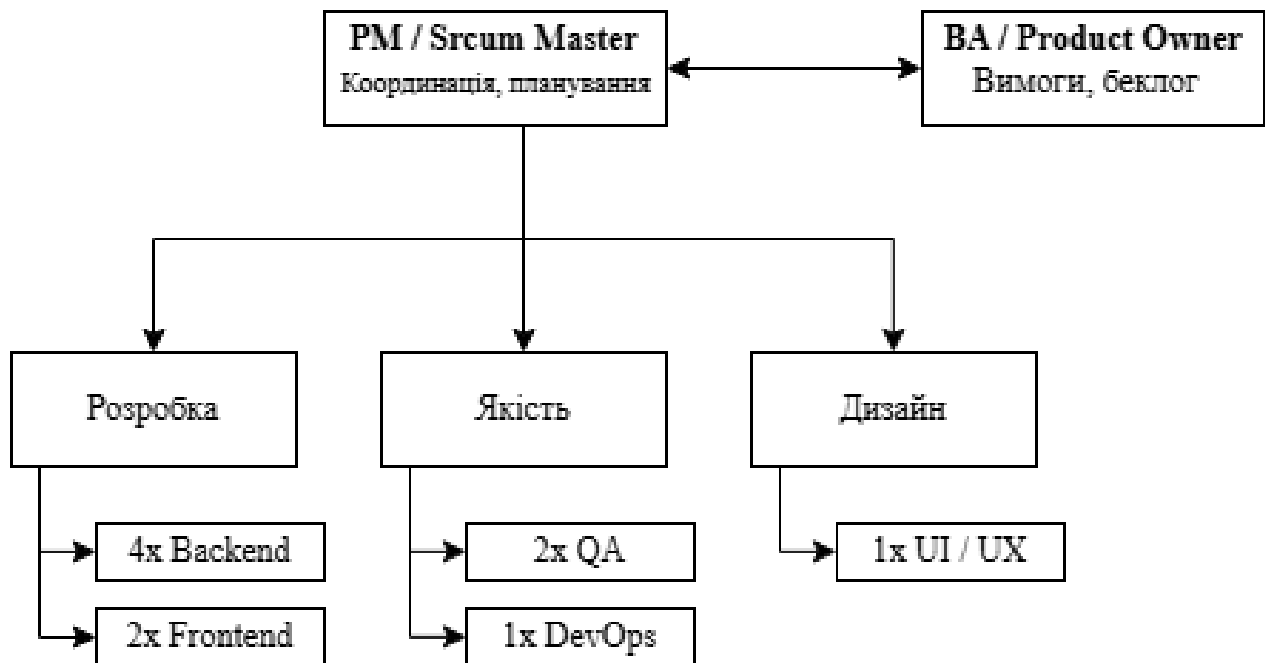


Рис. 2.2. Організаційна структура проєктної команди Trade&Earn з розподілом за фазами

Комунікаційна модель колективу повністю залежить від поточної фази робіт. На етапі проєктування спілкування централізується через менеджера. Він самостійно координує наради та звітує керівництву. Під час фази розробки підхід змінюється. Команда починає проводити щоденні короткі зустрічі для синхронізації. Перед початком кожної ітерації відбувається детальне планування завдань. Наприкінці спринту результати демонструються замовнику, після чого розробники аналізують свою ефективність на ретроспективі. Такий формат забезпечує тісну співпрацю із замовником без його надмірного втручання в операційну діяльність.

2.3. Обґрунтування вибору гібридної методології управління (Waterfall + Scrum)

Вибір гібридної методології для проєкту Trade&Earn базується на методі зваженої оцінки (weighted scoring), який дозволяє формалізувати управлінське рішення через вимірювані показники [13]. Цей підхід усуває суб'єктивність, оцінюючи кожну потенційну методологію за набором критеріїв, що відображають специфіку предметної області, регуляторного середовища та очікування стейкхолдерів.

Кожному з семи критеріїв присвоєно вагу методом попарного порівняння за участю менеджера проєкту та замовника. Оцінювання здійснювалося за шкалою від 1 (мінімальна відповідність) до 5 (максимальна відповідність). Визначальним фактором стала здатність методології вирішити конфлікт між прагненням замовника до фіксованого плану та потребою команди у гнучкості розробки(рис. 2.3). Результати наведено у таблиці 2.4.

Таблиця 2.4.

Зважена оцінка методологій управління для проєкту Trade&Earn

Критерій	Вага	Waterfall	Scrum	Kanban	Гібридна (W+S)
1	2	3	4	5	6
Передбачуваність термінів і бюджету для замовника	0,20	5	2	1	4
Гнучкість до змін функціональних вимог	0,15	1	5	5	4
Відповідність регуляторним вимогам (послідовність, документованість)	0,15	5	2	1	4
Короткий цикл зворотного зв'язку від замовника та користувачів	0,15	1	5	4	4
Можливість довгострокового планування (календарний план)	0,15	5	2	1	4

Врахування методологічного конфлікту замовник та команда	0,10	2	2	2	5
Зважений бал	1,00	3,20	3,00	2,20	4,20

Модель багатокритеріального вибору методології.

Задачу вибору методології управління формалізуємо як задачу багатокритеріальної оптимізації. Нехай задано:

- $M = \{m_1, m_2, m_3, m_4\}$ - множина альтернативних методологій: Waterfall, Scrum, Kanban, гібридна (W+S);
- $C = \{c_1, c_2, \dots, c_n\}$ - множина критеріїв оцінювання, $n = 6$;
- $w = (w_1, w_2, \dots, w_n)$ - вектор вагових коефіцієнтів критеріїв, де:

$$\sum_{i=1}^n w_i = 1, \quad w_i > 0$$

- $A = \|a_{ij}\|$ - матриця оцінок розміром $n \times |M|$, де $a_{ij} \in [1; 5]$ - оцінка методології m_j за критерієм c_i .

Інтегральний показник відповідності методології m_j умовам проєкту визначається як зважена сума оцінок за всіма критеріями:

$$S(m_j) = \sum_{i=1}^n w_i \cdot a_{ij}$$

Оптимальна методологія визначається як така, що максимізує інтегральний показник:

$$m^* = \arg \max_{m_j \in M} S(m_j)$$

Умова прийнятності методології - перевищення порогового значення $S_{\min}$, що встановлюється керівником проєкту та замовником:

$$S(m^*) \geq S_{\min}$$

Для проєкту Trade&Earn $S_{\min} = 3,5$ (за п'ятибальною шкалою). За результатами розрахунку:

$$S(m^*) = S(\text{Hybrid}) = 4,20 > 3,50 = S_{\min}$$

Єдина методологія, що задовольняє умову прийнятності це гібридна (W+S), що математично підтверджує обґрунтованість вибору.

Модель точки переходу між фазами (Waterfall, Scrum).

Ключовою характеристикою гібридної моделі є момент переходу T^* від каскадної фази до ітеративної. Запізнілий перехід збільшує вартість змін, передчасний це означає початок розробки на нестабільній архітектурі.

Визначимо вектор готовності проєкту $R(t)$ у момент часу t :

$$\sum_{i=1}^n w_i = 1, \quad w_i > 0$$

де:

- $R_{\text{arch}}(t) \in \{0, 1\}$ - бінарний індикатор затвердження архітектурного рішення (0 - не затверджено, 1 - затверджено);
- $R_{\text{req}}(t) \in [0, 1]$ - коефіцієнт стабільності вимог, що визначається як частка зафіксованих вимог від загальної кількості виявлених:

$$R_{\text{req}}(t) = \frac{N_{\text{fixed}}(t)}{N_{\text{total}}(t)}$$

- $R_{\text{reg}}(t) \in \{0, 1\}$ - бінарний індикатор формалізації регуляторних вимог (KYC/AML включено до WBS та оцінено);
- $R_{\text{team}}(t) \in [0, 1]$ - коефіцієнт готовності команди до ітеративної роботи (частка учасників, що пройшли навчання Scrum-процесу).

Умова переходу до ітеративної фази це виконання системи нерівностей:

$$\Phi(t): \begin{cases} R_{\text{arch}}(t) = 1 \\ R_{\text{req}}(t) \geq \theta_{\text{req}} \\ R_{\text{reg}}(t) = 1 \\ R_{\text{team}}(t) \geq \theta_{\text{team}} \end{cases}$$

де θ_{req} та θ_{team} — порогові значення, що встановлюються на етапі планування.

Оптимальний момент переходу визначається як найперший момент часу, у який усі умови виконані одночасно:

$$T^* = \min\{t: \Phi(t) = TRUE\}$$

Перехід відбувається у найперший момент часу, коли всі умови системи виконані одночасно. Якщо жодне t у плановому горизонті не задовольняє $\Phi(t)$, проєкт продовжується за каскадною моделлю до перегляду плану.

Для проєкту Trade&Earn порогові значення встановлено: $\theta_{req} = 0,85$; $\theta_{team} = 0,75$. Верифікацію умови $\Phi(t)$ проведено на контрольній точці КТ-2 (13 березня 2026 р.), результати наведено у таблиці 2.5.

Таблиця 2.5.

Верифікація умови переходу $\Phi(t)$ на контрольній точці КТ-2

Індикатор готовності	Порогове значення	Фактичне значення на КТ- 2	Умова виконана
1	2	3	4
1	2	3	4
R_{arch} - затвердження архітектури	= 1	1 (архітектуру затверджено)	Так
R_{req} - стабільність вимог	$\geq 0,85$	0,91	Так
R_{reg} - регуляторні вимоги	= 1	1 (KYC/AML включено до WBS)	Так
R_{team} - готовність команди	$\geq 0,75$	0,83	Так

Оскільки $\Phi(КТ-2) = TRUE$, перехід до ітеративної фази відбувся 19 березня 2026 р. (Спринт 1), що точно відповідає затвердженому календарному плану. Сукупність двох моделей формує цілісний механізм прийняття рішень у гібридному процесі: перша модель обґрунтовує вибір методології на підставі вимірюваних критеріїв, друга визначає момент активації ітеративної складової на підставі готовності проєкту. Використання запропонованого механізму усуває залежність від суб'єктивної оцінки та

підвищує відтворюваність результатів при застосуванні гібридної моделі в аналогічних ІТ-проєктах.

Особливе значення в розробленій моделі має критерій врахування методологічного конфлікту. Більшість класичних підходів до вибору методології фокусуються виключно на характеристиках продукту або середовища [14, 15]. Вони часто залишають поза увагою розбіжності в очікуваннях стейкхолдерів. Введення цього критерію дозволило кількісно оцінити здатність моделі управління забезпечити комфортні умови для всіх учасників. Цей аспект становить елемент наукової новизни даного дослідження.

На підставі результатів аналізу проєкт Trade&Earn буде реалізовуватися за гібридною моделлю. Подальше календарне планування, розподіл фінансів та структурування робіт будуть виконуватися з урахуванням поділу проєкту на жорсткі фази планування та гнучкі ітерації розробки.

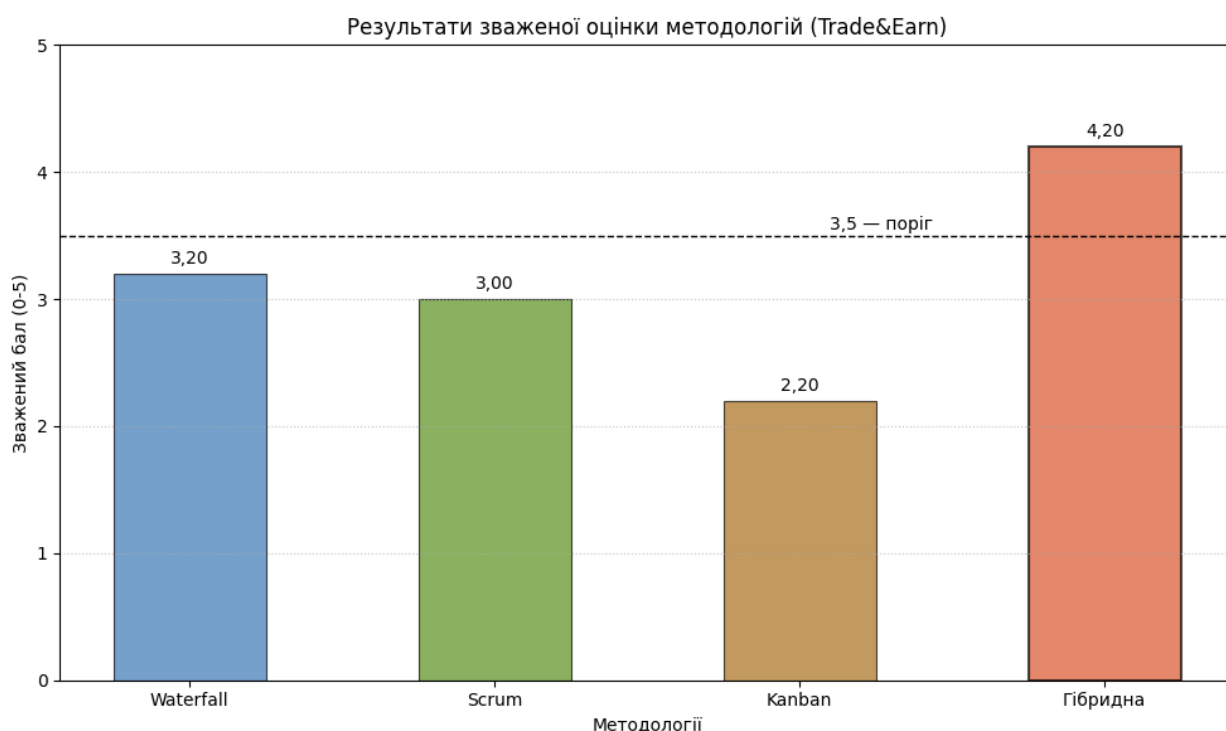


Рис. 2.3. Результати зваженої оцінки методологій управління для проєкту Trade&Earn

За результатами зваженої оцінки гібридна методологія отримала найвищий бал - 4,20, що є єдиним значенням, яке перевищує поріг

прийнятності 3,5. Waterfall і Scrum демонструють прийнятні, але недостатні результати. Kanban із балом 2,20 є найменш придатним варіантом для умов даного проєкту. Таким чином, кількісна оцінка підтверджує обґрунтованість вибору гібридної методології як основи моделі управління проєктом Trade&Earn.

2.4. Розробка ієрархічної структури робіт (WBS) та матриці відповідальності (RACI)

Ієрархічна структура робіт (WBS) проєкту Trade&Earn виступає базовим інструментом для оцінки бюджету, побудови графіка та розподілу відповідальності. Декомпозиція робіт дозволяє перетворити високорівневі цілі порталу на керовані завдання. Структура побудована на трьох рівнях: фази проєкту, пакети робіт та конкретні завдання. Основні елементи WBS представлено у таблиці 2.6.

Таблиця 2.6.

Ієрархічна структура робіт (WBS) проєкту Trade&Earn (рівні 1–3)

Код WBS	Назва елемента	Рівень
1	2	3
1	Ініціація	1
1.1	Розробка статуту проєкту	2
1.2	Ідентифікація стейкхолдерів	2
1.3	Формування команди	2
2	Планування	1
2.1	Збір та аналіз вимог	2
2.1.1	Функціональні вимоги	3
2.1.2	Нефункціональні вимоги (безпека, продуктивність)	3
2.1.3	Регуляторні вимоги (KYC/AML)	3
2.2	Побудова календарного плану	2
2.3	Оцінка бюджету та ресурсів	2
2.4	Планування ризиків	2
3	Проектування	1
3.1	Проектування архітектури порталу	2
3.1.1	Вибір технологічного стеку	3
3.1.2	Проектування серверної архітектури	3

1	2	3
3.1.3	Проектування клієнтської архітектури	3
3.2	Проектування бази даних	2
3.3	UI/UX-проектування	2
3.3.1	Створення wireframes	3
3.3.2	Розробка мокапів інтерфейсу	3
3.3.3	Прототипування та тестування UX	3
3.4	Проектування модуля безпеки	2
4	Розробка	1
4.1	Розробка бекенду	2
4.1.1	Модуль авторизації та автентифікації	3
4.1.2	Торговий модуль (ордери, балансова логіка)	3
4.1.3	Модуль KYC/AML	3
4.1.4	API для зовнішніх інтеграцій	3
4.2	Розробка фронтенду	2
4.2.1	Головна сторінка та навігація	3
4.2.2	Особистий кабінет користувача	3
4.2.3	Торгова панель	3
4.2.4	Освітній модуль	3
4.3	Інтеграція API котирувань	2
4.4	Модуль безпеки (2FA, шифрування)	2
5	Тестування	1
5.1	Функціональне тестування	2
5.2	Тестування безпеки (penetration testing)	2
5.3	Навантажувальне тестування	2
5.4	Приймальне тестування (UAT)	2
6	Впровадження	1
6.1	Розгортання продакшн-середовища	2
6.2	Міграція даних	2
6.3	Навчання користувачів	2
6.4	Запуск та моніторинг	2

Гібридна природа проєкту відображається у способі обробки елементів WBS на різних етапах. Пакети робіт фази «Розробка» (зокрема, модулі авторизації, торгівлі, KYC) трансформуються у беклог продукту. Кожен з 19

елементів третього рівня декомпозується на користувацькі історії (User Stories) і реалізується в межах двотижневих спринтів. Водночас етапи проектування, тестування та впровадження виконуються як класичні послідовні завдання, жорстко прив'язані до контрольних точок графіка.

Для ефективного управління виконанням робіт застосовано матрицю RACI, яка визначає ступінь залученості кожного члена команди до конкретних пакетів WBS. Вона ліквідує розмиття відповідальності, чітко розмежовуючи ролі Accountable (підзвітний), Responsible (виконавець), Consulted (експерт) та Informed (проінформований).

Таблиця 2.7.

Матриця відповідальності RACI проєкту Trade&Earn

Пакет робіт	PM	BA	Бекенд	Фронтенд	UI/UX	QA	DevOps
1	2	3	4	5	6	7	8
1.1 Статут проєкту	A, R	C	I	I	I	I	I
2.1 Збір вимог	A	R	C	C	C	I	I
3.1 Архітектура порталу	A	C	R	C	I	I	C
3.2 Проектування БД	A	C	R	I	I	I	I
3.3 UI/UX- проектування	A	C	I	C	R	I	I
4.1 Розробка бекенду	A	C	R	I	I	C	C
4.2 Розробка фронтенду	A	C	I	R	C	C	I
4.4 Модуль безпеки	A	I	R	I	I	C	C
5.1 Функціональне тестування	A	C	C	C	I	R	I

1	2	3	4	5	6	7	8
5.2 Тестування безпеки	A	I	C	I	I	R	C
6.1 Розгортання продакшн	A	I	C	I	I	I	R

Розподіл відповідальності підтверджує управлінську концепцію гібридного підходу(рис. 2.4). Менеджер проєкту (PM) зберігає підзвітність (роль A) за всі етапи, гарантуючи досягнення цілей. Бізнес-аналітик (BA) виступає основним виконавцем (R) під час збору вимог, а на подальших етапах переходить у статус експерта (C). Під час фази розробки функція експерта реалізується через управління беклогом в ролі Product Owner, що дозволяє контролювати відповідність інкрементів первинним вимогам бізнесу без втручання в технічні деталі.

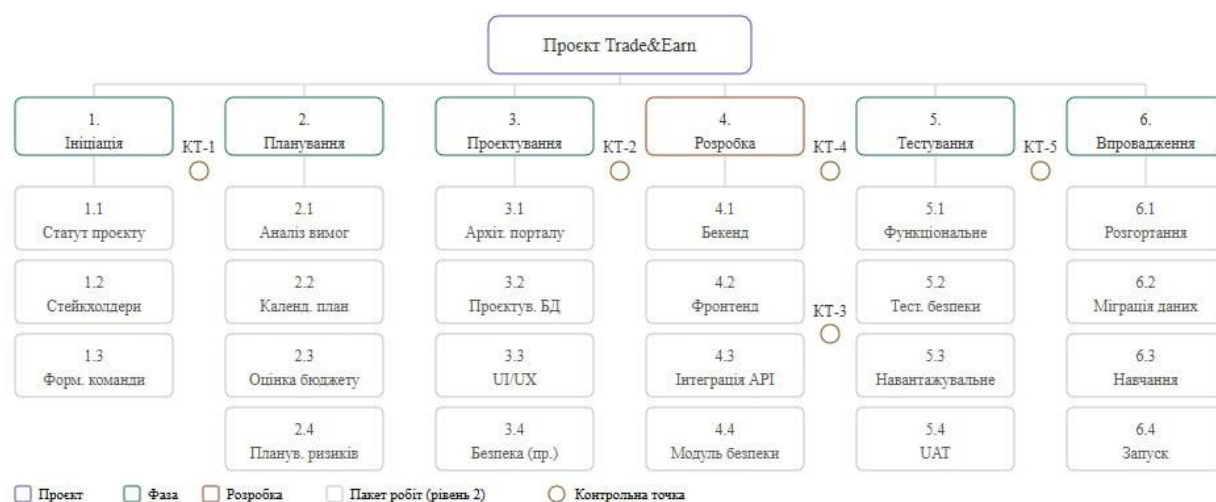


Рис. 2.4. Ієрархічна структура робіт (WBS) проєкту Trade&Earn, рівні 1–2
Кошторисний розрахунок проєкту та оцінка трудомісткості

Бюджет проєкту Trade&Earn розраховано на основі детальної оцінки трудомісткості пакетів робіт WBS та актуальних ринкових ставок ІТ-спеціалістів. При цьому оцінювання виконувалося із використанням підходу bottom-up, що передбачає агрегування вартості окремих задач до рівня всього

проєкту, забезпечуючи вищу точність розрахунків. Фінансове планування враховує три основні категорії витрат: оплату праці проєктної команди (яка традиційно формує до 75% бюджету ІТ-проєктів [17]), витрати на інфраструктуру та обов'язковий управлінський резерв.

Трудовітність кожної фази визначено методом експертної оцінки з огляду на загальне обмеження тривалості проєкту у 39 робочих тижнів (9 місяців). Розподіл трудовитрат підтверджує фокус на ітеративній реалізації: 67% загального часу припадає безпосередньо на фазу розробки. Розподіл трудовітності наведено у таблиці 2.8. Такий розподіл відповідає сучасним підходам до розробки програмного забезпечення, де основний обсяг ресурсів зосереджується на створенні та ітеративному вдосконаленні функціоналу.

Таблиця 2.8.

Трудовітність фаз проєкту Trade&Earn

Фаза WBS	Тривалість (тижні)	Залучені ролі	Кількість залучених осіб	Трудовітність (люд.-тижні)
1	2	3	4	5
1. Ініціація	2	PM, BA	2	4
2. Планування	4	PM, BA, бекенд (2), QA (1)	5	20
3. Проєктування	5	PM, BA, бекенд (4), фронтенд (2), UI/UX, DevOps	10	50
4. Розробка	20	Вся команда (12)	12	240
5. Тестування	5	PM, бекенд (2), QA (2), DevOps	6	30
6. Впровадження	3	PM, DevOps, бекенд (1), QA (1)	4	12
Разом	39			356

Витрати на оплату праці базуються на середніх компенсаціях ІТ-спеціалістів в Україні станом на початок 2025 року за даними профільного

порталу DOU.ua [18]. Базові місячні ставки перераховано в тижневий еквівалент для зручності співставлення з тривалістю залучення кожного фахівця. Розрахунок наведено у таблиці 2.9.

Таблиця 2.9.

Розрахунок витрат на оплату праці проєктної команди					
Роль	К-сть	Середня ставка (грн/міс.)	Ставка (грн/тиж.)	Залучення (тижні)	Витрати (грн)
1	2	3	4	5	6
Менеджер проєкту	1	65 000	16 250	39	633 750
Бекенд-розробник	4	55 000	13 750	32 (серед.)	1 760 000
Фронтенд-розробник	2	50 000	12 500	28 (серед.)	700 000
UI/UX-дизайнер	1	45 000	11 250	26	292 500
Тестувальник (QA)	2	40 000	10 000	30 (серед.)	600 000
DevOps	1	60 000	15 000	30	450 000
Бізнес-аналітик	1	50 000	12 500	28	350 000
Разом оплата праці	12				4 786 250

Сума базових витрат на оплату праці (понад 4,7 млн грн) оптимізується до планових 1 850 000 грн завдяки ефективному розподілу ресурсів. Управлінське рішення полягає у частковому завантаженні (з коефіцієнтом від 0,5 до 0,8) тих фахівців, які паралельно задіяні в інших ініціативах компанії. Це дозволяє вписатися в жорсткі рамки бюджету без втрати якості та технічних компетенцій команди.

Витрати на розгортання хмарної інфраструктури, ліцензування ПЗ та інтеграцію спеціалізованих сервісів (KYC, API котирувань) наведено у таблиці 2.10.

Таблиця 2.10.

Витрати на інфраструктуру та зовнішні сервіси

Стаття витрат	Вартість (грн/міс.)	Кількість місяців	Разом (грн)
1	2	3	4
Хмарний хостинг (AWS/GCP)	25 000	9	225 000
API котирувань (CoinGecko Pro / CoinMarketCap)	5 000	7	35 000
Ліцензія MS Project	2 500	9	22 500
Ліцензії Figma (для дизайнера)	1 500	6	9 000
Інструменти CI/CD (GitHub Actions)	3 000	7	21 000
Сервіс KYC-верифікації (Sumsb / Jumio)	8 000	4	32 000
SSL-сертифікати, домен, DNS	1 000	9	9 000
Разом інфраструктура			353 500

Зведений кошторис формується з урахуванням 10% управлінського резерву на непередбачені ризики. Це обов'язкова складова фінансового планування, що дозволяє менеджеру проєкту оперативно реагувати на зміни вимог без процедури переузгодження бюджету із замовником.

Таблиця 2.11.

Зведений кошторис проєкту Trade&Earn

Стаття	Сума (грн)	Частка (%)
1	2	3
Оплата праці (з урахуванням коефіцієнта завантаження)	1 850 000	74,0
Інфраструктура та зовнішні сервіси	353 500	14,1
Управлінський резерв (10%)	220 350	8,8

Продовження табл. 2.11.

1	2	3
Непрямі витрати (офіс, комунікації, обладнання)	76 150	3,1
Разом бюджет проєкту	2 500 000	100,0

Загальний обсяг фінансування становить 2,5 млн грн, що повністю відповідає початковим інвестиційним обмеженням. Отриманий кошторис є фундаментом для подальшого ресурсного планування у MS Project (Розділ 3) та розрахунку економічної ефективності порталу.

2.5. Календарне планування та часові обмеження проєкту

Календарний план проєкту Trade&Earn побудовано на основі логічних залежностей між фазами WBS та розрахованої трудомісткості. Проєкт триватиме 9 місяців (39 тижнів) з 1 січня 2026 року по 30 вересня 2026 року. Управління часом спирається на комбінацію послідовних етапів і гнучких ітерацій, об'єднаних системою жорстких контрольних точок (Milestones). Календарний розподіл фаз наведено у таблиці 2.12.

Таблиця 2.12.

Календарний план проєкту Trade&Earn (за фазами)

Фаза	Початок	Завершення	Тривалість (тижні)	Контрольна точка
1	2	3	4	5
1. Ініціація	01.01.2026	14.01.2026	2	КТ-1 (14.01)
2. Планування	15.01.2026	11.02.2026	4	
3. Проектування	12.02.2026	18.03.2026	5	КТ-2 (18.03)
4. Розробка (Спринти 1–10)	19.03.2026	05.08.2026	20	КТ-3 (27.05, після Спринту 5)
5. Тестування	06.08.2026	09.09.2026	5	КТ-4 (09.09)
6. Впровадження	10.09.2026	30.09.2026	3	КТ-5 (30.09)

Фаза розробки охоплює 20 тижнів і розділена на 10 двотижневих спринтів. Такий формат рекомендовано Scrum Guide як оптимальний баланс між швидким отриманням зворотного зв'язку від стейкхолдерів та можливістю команди завершити значущий функціональний інкремент [6]. Пріоритети беклогу вибудовані за принципом архітектурної та бізнес-залежності: спочатку реалізується базова інфраструктура, потім торговий модуль, і наприкінці - освітні матеріали та полірування інтерфейсу. Орієнтовний розподіл робіт по спринтах наведено у таблиці 2.13.

Таблиця 2.13.

Розподіл пакетів робіт WBS по спринтах

Спринт	Період	Основні пакети робіт WBS
1	2	3
Спринт 1	19.03 – 01.04.2026	4.1.1 (авторизація, автентифікація), 4.2.1 (головна сторінка)
Спринт 2	02.04 – 15.04.2026	4.1.1 (завершення), 4.2.2 (особистий кабінет)
Спринт 3	16.04 – 29.04.2026	4.1.2 (торговий модуль, початок), 4.3 (інтеграція API котирувань)
Спринт 4	30.04 – 13.05.2026	4.1.2 (торговий модуль, продовження), 4.2.3 (торгова панель, початок)
Спринт 5	14.05 – 27.05.2026	4.1.2 (завершення), 4.2.3 (завершення). КТ-3 (працюючий прототип)
Спринт 6	28.05 – 10.06.2026	4.4 (модуль безпеки, 2FA), 4.1.3 (KYC/AML, початок)
Спринт 7	11.06 – 24.06.2026	4.1.3 (KYC/AML, завершення), 4.1.4 (API для зовнішніх інтеграцій)
Спринт 8	25.06 – 08.07.2026	4.2.4 (освітній модуль), удосконалення UI
Спринт 9	09.07 – 22.07.2026	4.2.4 (завершення), інтеграційне тестування у межах спринту
Спринт 10	23.07 – 05.08.2026	Стабілізація, виправлення дефектів, Code Freeze

Управлінським ядром календарного плану є контрольна точка КТ-3, запланована після п'ятого спринту (середина розробки). У цей момент

менеджер проекту разом із замовником оцінюють працюючий прототип платформи. Це дозволяє вчасно виявити відхилення від графіка та за потреби переглянути беклог (скоротити MVP або змінити пріоритети), не чекаючи завершення всієї фази розробки. Після десятого спринту вводиться період заморозки коду (Code Freeze) - жорстке правило, що забороняє додавання нових функцій для забезпечення стабільності перед приймальним тестуванням.

Важливим часовим обмеженням проекту є процес отримання ліцензії НКЦПФР, який триває 3–6 місяців [9]. З управлінської точки зору цей процес запускається паралельно з етапами тестування. Хоча він не перешкоджає бета-запуску платформи, повноцінне комерційне функціонування неможливе без завершення цієї процедури, що має відображатися у плані реагування на ризики.

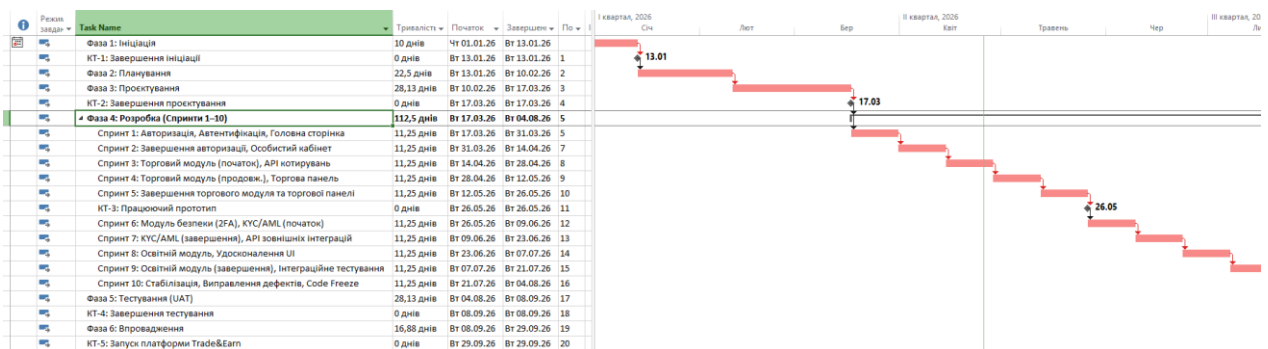


Рис. 2.5. Укрупнений календарний план проекту Trade&Earn

Як видно з рисунку 2.5, кожна фаза проекту має чітко визначені дати початку та завершення, а контрольні точки КТ-1 - КТ-5 фіксують ключові моменти переходу між стадіями. Десять спринтів стадії розробки виконуються послідовно з рівномірним навантаженням по 11,25 дні кожен, що забезпечує передбачуваний ритм роботи команди протягом усього виконавчого етапу.

2.6. Розрахунок економічної ефективності проекту (NPV, IRR)

Оцінка економічної ефективності проекту Trade&Earn визначає доцільність інвестицій з боку ТОВ «ТрейдСофт». Аналіз ґрунтується на двох ключових показниках: чистій приведеній вартості (NPV) та внутрішній нормі

рентабельності (IRR), які є стандартом інвестиційного аналізу та рекомендуються практиками PMI [19].

Фінансова модель враховує початкові інвестиції у розмірі 2,5 млн грн (бюджет розробки) та трирічний горизонт прогнозування (2027–2029 роки). Ставку дисконтування встановлено на консервативному рівні 20%, що компенсує інвестору високі ризики волатильності криптовалютного ринку та враховує поточну вартість капіталу в Україні. Дохідна частина формується з торгових комісій (0,15% від обсягу угод) та продажів преміум-підписок, тоді як витратна охоплює маркетинг, підтримку інфраструктури та операційну діяльність. Прогноз грошових потоків наведено у таблиці 2.14.

Таблиця 2.14.

Прогноз грошових потоків проєкту Trade&Earn

Показник	Рік 0 (розробка)	Рік 1 (2027)	Рік 2 (2028)	Рік 3 (2029)
1	2	3	4	5
Кількість активних користувачів	-	5 000	15 000	35 000
Середній місячний обсяг торгів на користувача (грн)	-	50 000	60 000	70 000
Дохід від комісій (грн)	-	4 500 000	16 200 000	44 100 000
Дохід від преміум-підписки (грн)	-	300 000	1 200 000	2 800 000
Сумарний дохід	-	4 800 000	17 400 000	46 900 000
Операційні витрати (грн)	-	3 600 000	8 400 000	18 000 000
Чистий грошовий потік (CF)	-2 500 000	1 200 000	9 000 000	28 900 000

Розрахунок NPV здійснюється за формулою:

$$NPV = -I_0 + \sum (CF_t / (1 + r)^t), \text{ де } t = 1..3.$$

Підставивши відповідні значення, отримуємо:

$$NPV = -2\,500\,000 + 1\,200\,000 / (1,20)^1 + 9\,000\,000 / (1,20)^2 + 28\,900\,000 / (1,20)^3 = -2\,500\,000 + 1\,000\,000 + 6\,250\,000 + 16\,724\,537 = 21\,474\,537 \text{ грн.}$$

Позитивне значення NPV понад 21 млн грн підтверджує абсолютну фінансову ефективність проекту та гарантує приріст капіталу інвестора.

Для визначення IRR розв'язується рівняння $NPV = 0$ відносно ставки дисконтування r . Розрахунок методом послідовних наближень дає результат $IRR \approx 178\%$.

Надзвичайно високе значення IRR (178%), яке суттєво перевищує ставку дисконтування, потребує управлінського пояснення. Для класичного реального сектора економіки такий показник виглядав би нереалістично. Проте для фінтех-стартапів і SaaS-платформ, до яких належить Trade&Earn, це є закономірним наслідком бізнес-моделі з нульовими або мінімальними маржинальними витратами. Після створення базової інфраструктури залучення кожного нового трейдера майже не збільшує операційні витрати компанії, тоді як обсяг генерованих комісій зростає експоненційно. Аналогічні показники IRR фіксуються на ранніх стадіях розвитку більшості успішних криптовалютних бірж та необанків.

Простий термін окупності платформи становить 14 місяців після запуску, дисконтований - 18 місяців. Ці показники свідчать про швидке повернення інвестицій. Результати аналізу зведено у таблиці 2.15.

Таблиця 2.15.

Зведені показники економічної ефективності проекту Trade&Earn

Показник	Значення	Критерій ефективності	Висновок
1	2	3	4
NPV (при $r = 20\%$)	21 474 537 грн	$NPV > 0$	Проект ефективний
IRR	178%	$IRR > r (20\%)$	Проект інвестиційно привабливий
Простий термін окупності	14 місяців	< 3 роки	Прийнятний
Дисконтований термін окупності	18 місяців	< 3 роки	Прийнятний

Інвестиційний аналіз доводить високу комерційну привабливість розробки portalу. Водночас модель залишається чутливою до зовнішніх загроз, зокрема до темпів приросту користувацької бази та регуляторних обмежень. Ці фактори вимагають створення надійного плану управління ризиками, що розглядається у Розділі 4.

РОЗДІЛ 3. ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ ПЛАНУВАННЯ ТА УПРАВЛІННЯ ПРОЄКТОМ

3.1. Календарно-сітьове планування у MS Project: діаграма Ганта та критичний шлях

Календарно-сітьову модель проєкту Trade&Earn побудовано у MS Project на основі WBS із Розділу 2. Модель містить 60 рядків, з яких 49 є робочими завданнями, 6 сумарними фазами та 5 контрольними точками (milestones). Дата початку 1 січня 2026 року, розрахована дата завершення 22 вересня 2026 року, що складає 38 робочих тижнів. Побудова моделі у спеціалізованому програмному забезпеченні дозволила розрахувати критичний шлях, виявити ресурсні конфлікти та отримати обґрунтовану дату завершення, яка недосяжна при ручному плануванні 60 взаємопов'язаних завдань.

Високорівнева WBS із Розділу 2 містила 19 пакетів робіт третього рівня, що виявилось недостатнім для точного моделювання. Пакет на кшталт «Тестування» неможливо призначити конкретному спеціалісту без подальшої декомпозиції. Тому структуру деталізовано до 49 окремих завдань. Найбільшого розширення зазнали фаза проєктування, доповнена проєктуванням REST API, системи логування та резервного копіювання, і фаза тестування, до якої додано кросбраузерне тестування та тестування доступності. Деталізація дозволила призначити кожне завдання конкретному спеціалісту та виявити потенційні ресурсні конфлікти ще на етапі планування.

Між завданнями каскадних фаз встановлено залежності типу Finish-to-Start (FS), оскільки проєктування архітектури не може розпочатись до фіксації вимог, а тестування не може стартувати до завершення розробки. При затримці будь-якого попереднього завдання MS Project автоматично перераховує дати всіх наступних, що дає менеджеру миттєвий сигнал про відхилення від плану. Залежності між спринтами також встановлено як послідовні, оскільки кожен спринт розпочинається лише після Demo і ретроспективи попереднього. Сукупність залежностей утворює єдиний сітьовий граф проєкту.

Частина робіт виконується паралельно, що скорочує загальну тривалість проєкту. На фазі проєктування архітектурне та UI/UX-проєктування відбуваються одночасно, оскільки залучають різні групи спеціалістів. Проєктування серверної та клієнтської архітектури також запаралелено після вибору технологічного стеку, адже ці потоки робіт не мають спільних ресурсів. На фазі тестування після функціонального тестування паралельно виконуються тестування безпеки, навантажувальне, кросбраузерне та тестування доступності, кожне з яких веде окремий спеціаліст. Без такого паралелізму фаза тестування тривала б 29 днів замість 22, а весь проєкт завершився б на тиждень пізніше.

У моделі зафіксовано п'ять контрольних точок з нульовою тривалістю, кожна з яких позначає момент прийняття управлінського рішення. КТ-1 (20 січня) фіксує затвердження статуту проєкту, без якого команда не може офіційно розпочати планування. КТ-2 (13 березня) позначає перехід до ітеративної розробки після затвердження архітектурних рішень, і з цього моменту будь-які зміни в архітектурі стають суттєво дорогими. КТ-3 (22 травня) є найважливішою з управлінської точки зору, оскільки замовник вперше бачить працюючий продукт і приймає рішення щодо пріоритетів беклогу на залишені спринти. КТ-4 (1 вересня) підтверджує готовність продукту до впровадження після успішного UAT. КТ-5 (22 вересня) фіксує комерційний запуск порталу.

Критичний шлях являє собою найдовшу послідовність залежних завдань, затримка будь-якого з яких зсуває дату завершення всього проєкту. Для Trade&Earn він проходить через усі шість фаз і складає 38 тижнів, а резерв часу (float) для кожного завдання на критичному шляху дорівнює нулю. Навіть одноденна затримка у проєктуванні серверної архітектури зсує початок усіх спринтів і дату комерційного запуску. Найбільший ризик зосереджений у фазі розробки, яка займає 100 робочих днів і складає понад 52% критичного шляху. Некритичні завдання мають невеликий резерв часу, наприклад, UI/UX-проєктування завершується раніше за архітектурне, що дозволяє залучити

дизайнера до підготовки мокапів для першого спринту без впливу на загальний графік.

Порівняння з попередньою оцінкою підтвердило точність детального планування. У Розділі 2 тривалість проєкту оцінювалась у 39 тижнів із завершенням 30 вересня за методом top-down. Після деталізації до 49 завдань, призначення ресурсів та врахування паралельного виконання робіт розрахована тривалість склала 38 тижнів із датою завершення 22 вересня за методом bottom-up. Різниця у тиждень пояснюється тим, що top-down оцінка не враховувала паралельну роботу різних груп спеціалістів, тоді як детальна модель у MS Project відобразила реальний розподіл навантаження між командою. Діаграму Ганта проєкту Trade&Earn наведено на рисунках 3.1-3.3.



Рис. 3.1. Укрупнена шкала часу (Timeline) проєкту Trade&Earn з фазами та контрольними точками

На рисунках 3.1 – 3.5 показано загальний вигляд проєкту у форматі Timeline, призначений для звітності перед стейкхолдерами. Шкала охоплює період з січня по вересень 2026 року, відображає шість послідовних фаз та контрольні точки КТ-1 - КТ-5. Фаза розробки з десяти спринтів візуально домінує, що наочно підтверджує її критичну вагу у загальному графіку.

Режим задачі	Task Name	Тривалість	Початок	Завершення	Попередники	Тип обмеження	Імена ресурсів
✓	КТ-1: Затвердження статуту	0 днів	Вт 20.01.26	Вт 20.01.26	6	Якомога раніше	
✓	Фаза 2. Планування	21 днів	Ср 21.01.26	Ср 18.02.26		Якомога раніше	
✓	2.1 Збір та аналіз вимог	9 днів	Ср 21.01.26	Пн 02.02.26		Якомога раніше	
✓	2.1.1 Функціональні вимоги	5 днів	Ср 21.01.26	Вт 27.01.26	7	Якомога раніше	BA (Бізнес-аналітик);Backend-1;Backend-2;PM (Менеджер проекту)
✓	2.1.2 Нефункціональні вимоги	4 днів	Ср 28.01.26	Пн 02.02.26	10	Якомога раніше	Backend-1;QA-1
✓	2.1.3 Регуляторні вимоги (KYC/AML)	4 днів	Ср 28.01.26	Пн 02.02.26	10	Якомога раніше	BA (Бізнес-аналітик);PM (Менеджер проекту)
✓	2.2 Побудова календарного плану	5 днів	Вт 03.02.26	Пн 09.02.26	11;12	Якомога раніше	BA (Бізнес-аналітик);PM (Менеджер проекту)
✓	2.3 Оцінка бюджету та ресурсів	4 днів	Вт 10.02.26	Пт 13.02.26	11;12	Якомога раніше	BA (Бізнес-аналітик)
✓	2.4 Планування ризиків	2 днів	Вт 10.02.26	Ср 11.02.26	11;12	Якомога раніше	PM (Менеджер проекту)
✓	2.5 Планування якості та критеріїв приймання	3 днів	Вт 03.02.26	Чт 05.02.26	11;12	Якомога раніше	QA-1
✓	2.6 Планування ризиків	3 днів	Пн 16.02.26	Ср 18.02.26	13;14	Якомога раніше	BA (Бізнес-аналітик);PM (Менеджер проекту);QA-1
✓	Фаза 3. Проектування	17 днів	Чт 19.02.26	Пт 13.03.26		Якомога раніше	
✓	3.1 Проектування архітектури порталу	12 днів	Чт 19.02.26	Пт 06.03.26		Якомога раніше	
✓	3.1.1 Вибір технологічного стеку	3 днів	Чт 19.02.26	Пн 23.02.26	17	Якомога раніше	Backend-1;Backend-2;DevOps;PM (Менеджер проекту)
✓	3.1.2 Проектування серверної архітектури	5 днів	Вт 24.02.26	Пн 02.03.26	20	Якомога раніше	Backend-1;Backend-2;Backend-3;Backend-4
✓	3.1.3 Проектування клієнтської архітектури	4 днів	Вт 24.02.26	Пт 27.02.26	20	Якомога раніше	Frontend-1;Frontend-2
✓	3.1.4 Проектування REST API	4 днів	Вт 03.03.26	Пт 06.03.26	21	Якомога раніше	Backend-1
✓	3.1.5 Проектування системи логування та моніторингу	3 днів	Вт 03.03.26	Чт 05.03.26	21	Якомога раніше	DevOps
✓	3.2 Проектування бази даних	7 днів	Вт 03.03.26	Ср 11.03.26	21	Якомога раніше	Backend-2
✓	3.3 Проектування системи резервного копіювання	2 днів	Чт 12.03.26	Пт 13.03.26	25	Якомога раніше	Backend-3;DevOps
✓	3.4 UI/UX-проектування	12 днів	Чт 19.02.26	Пт 06.03.26		Якомога раніше	
✓	3.4.1 Створення wireframes	4 днів	Чт 19.02.26	Вт 24.02.26	17	Якомога раніше	UI/UX Designer
✓	3.4.2 Розробка макетів інтерфейсу	5 днів	Ср 25.02.26	Вт 03.03.26	28	Якомога раніше	UI/UX Designer
✓	3.4.3 Прототипування та тестування UX	3 днів	Ср 04.03.26	Вт 06.03.26	29	Якомога раніше	Frontend-2;QA-1;UI/UX Designer
✓	3.5 Проектування модуля безпеки	5 днів	Вт 03.03.26	Пн 09.03.26	21	Якомога раніше	Backend-3;Backend-4
✓	КТ-2: Затвердження архітектури	0 днів	Пт 13.03.26	Пт 13.03.26	31;23;24;26;30	Якомога раніше	

Рис 3.2. Діаграма Ганта проекту Trade&Earn (Фази 1-3)

Режим задачі	Task Name	Тривалість	Початок	Завершення	Попередники	Тип обмеження	Імена ресурсів	одати новіт
✓	Фаза 4. Розробка (10 спринтів)	10 днів	Пн 16.03.26	Пт 31.07.26		Якомога раніше		
✓	Спринт 1: Авторизація, головна сторінка	10 днів	Пн 16.03.26	Пт 27.03.26	32	Якомога раніше	BA (Бізнес-аналітик);Backend-1;Backend-2;DevOps;Frontend-1;PM (Менеджер проекту);QA-1;UI/UX Designer	
✓	Спринт 2: Особистий кабінет	10 днів	Пн 30.03.26	Пт 10.04.26	34	Якомога раніше	BA (Бізнес-аналітик);Backend-1;Backend-2;DevOps;Frontend-1;Frontend-2;PM (Менеджер проекту);QA-1;UI/UX Designer	
✓	Спринт 3: Торговий модуль (початок), API котирувань	10 днів	Пн 13.04.26	Пт 24.04.26	35	Якомога раніше	BA (Бізнес-аналітик);Backend-1;Backend-2;Backend-3;DevOps;Frontend-1;Frontend-2;PM (Менеджер проекту);QA-1;UI/UX Designer	
	Спринт 4: Торговий модуль (продовження), торгова панель	10 днів	Пн 27.04.26	Пт 08.05.26	36	Якомога раніше	BA (Бізнес-аналітик);Backend-1;Backend-2;Backend-3;DevOps;Frontend-1;Frontend-2;PM (Менеджер проекту);QA-1;UI/UX Designer	
	Спринт 5: Торговий модуль (завершення)	10 днів	Пн 11.05.26	Пт 22.05.26	37	Якомога раніше	BA (Бізнес-аналітик);Backend-1;Backend-2;Backend-3;Backend-4;DevOps;Frontend-1;Frontend-2;PM (Менеджер проекту);QA-1;UI/UX Designer	
	КТ-3: Працюючий прототип	0 днів	Пт 22.05.26	Пт 22.05.26	38	Якомога раніше		
	Спринт 6: Модуль безпеки (2FA), KYC/AML (початок)	10 днів	Пт 25.05.26	Пт 05.06.26	39	Якомога раніше	BA (Бізнес-аналітик);Backend-1;Backend-2;Backend-3;Backend-4;DevOps;Frontend-1;PM (Менеджер проекту);QA-1;UI/UX Designer	
	Спринт 7: KYC/AML (завершення), зовнішні API	10 днів	Пн 08.06.26	Пт 19.06.26	40	Якомога раніше	BA (Бізнес-аналітик);Backend-1;Backend-2;Backend-3;Backend-4;DevOps;Frontend-1;PM (Менеджер проекту);QA-1;UI/UX Designer	
	Спринт 8: Освітній модуль, удосконалення UI	10 днів	Пн 22.06.26	Пт 03.07.26	41	Якомога раніше	BA (Бізнес-аналітик);DevOps;Frontend-1;Frontend-2;PM (Менеджер проекту);QA-1;UI/UX Designer	
	Спринт 9: Інтеграційне тестування у спринті	10 днів	Пн 06.07.26	Пт 17.07.26	42	Якомога раніше	BA (Бізнес-аналітик);Backend-1;Backend-2;DevOps;Frontend-1;Frontend-2;PM (Менеджер проекту);QA-1;UI/UX Designer	
	Спринт 10: Стабілізація, Code Freeze	10 днів	Пн 20.07.26	Пт 31.07.26	43	Якомога раніше	BA (Бізнес-аналітик);Backend-1;Backend-2;DevOps;Frontend-1;PM (Менеджер проекту);QA-1;QA-2	

Рис 3.3. Діаграма Ганта проекту Trade&Earn (Фаза 4)

	Фаза 5. Тестування	22 днів	Пн 03.08.26	Вт 01.09.26		Якомога раніше	
	5.1 Функціональне тестування	10 днів	Пн 03.08.26	Пт 14.08.26	44	Якомога раніше	Backend-1;Frontend-1;PM (Менеджер проекту);QA-1;QA-2
	5.2 Тестування безпеки (penetration testing)	7 днів	Пн 17.08.26	Вт 25.08.26	46	Якомога раніше	Backend-3;QA-1
	5.3 Навантажувальне тестування	5 днів	Пн 17.08.26	Пт 21.08.26	46	Якомога раніше	Backend-1;DevOps
	5.4 Кросбраузерне тестування	4 днів	Пн 17.08.26	Чт 20.08.26	46	Якомога раніше	Frontend-2
	5.5 Тестування доступності (accessibility)	3 днів	Пн 17.08.26	Ср 19.08.26	46	Якомога раніше	Frontend-1;UI/UX Designer
	5.6 Приймальне тестування (UAT)	5 днів	Ср 26.08.26	Вт 01.09.26	47;48;49;50	Якомога раніше	BA (Бізнес-аналітик);PM (Менеджер проекту);QA-1;QA-2
	КТ-4: Готовність до впровадження	0 днів	Вт 01.09.26	Вт 01.09.26	51	Якомога раніше	
	Фаза 6. Впровадження	15 днів	Ср 02.09.26	Вт 22.09.26		Якомога раніше	
	6.1 Підготовка експлуатаційної документації	4 днів	Ср 02.09.26	Пн 07.09.26	52	Якомога раніше	BA (Бізнес-аналітик)
	6.2 Розгортання продакшн-середовища	4 днів	Ср 02.09.26	Пн 07.09.26	52	Якомога раніше	Backend-1;DevOps;PM (Менеджер проекту)
	6.3 Налаштування системи моніторингу та алертів	3 днів	Вт 08.09.26	Чт 10.09.26	55	Якомога раніше	Backend-3;DevOps
	6.4 Міграція даних	6 днів	Вт 08.09.26	Вт 15.09.26	55	Якомога раніше	Backend-2;DevOps
	6.5 Навчання користувачів	4 днів	Вт 08.09.26	Пт 11.09.26	55	Якомога раніше	BA (Бізнес-аналітик);PM (Менеджер проекту)
	6.6 Запуск та моніторинг	5 днів	Ср 16.09.26	Вт 22.09.26	57;58;56	Якомога раніше	Backend-1;DevOps;PM (Менеджер проекту);QA-1
	КТ-5: Комерційний запуск	0 днів	Вт 22.09.26	Вт 22.09.26	59	Якомога раніше	

Рис. 3.4. Діаграма Ганта проекту Trade&Earn (Фази 5-6)

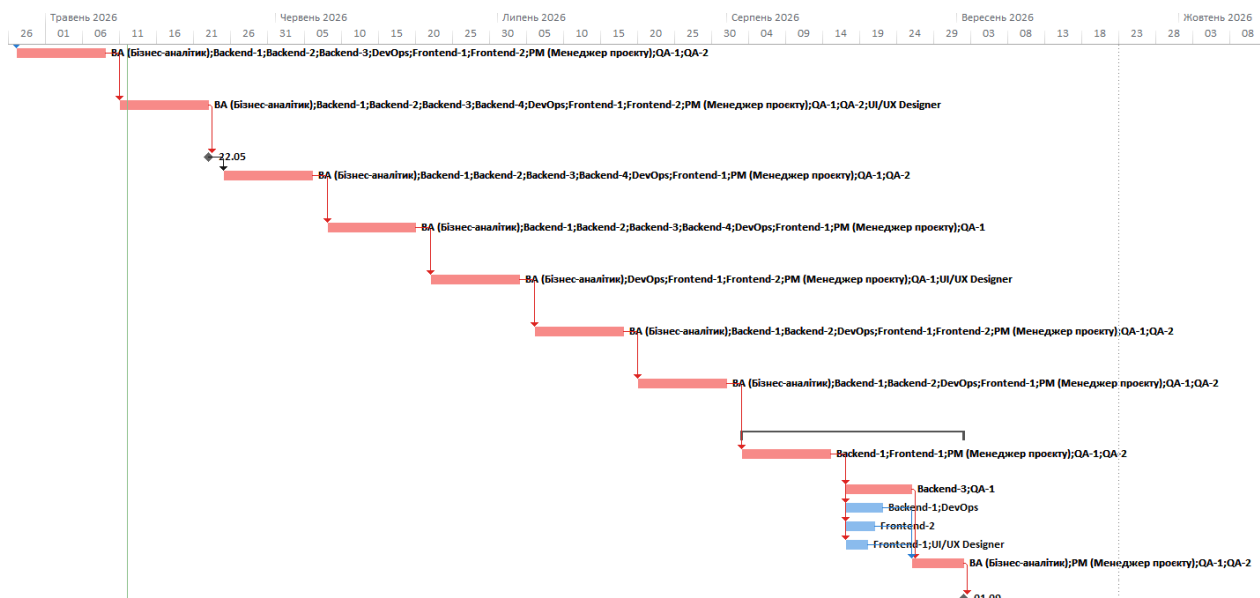


Рис. 3.5. Діаграма Ганта проєкту Trade&Earn (Фази 4-6)

На рисунках 3.2 та 3.3 показано детальну діаграму Ганта з усіма 60 рядками проєкту. Ліва частина містить таблицю завдань з тривалістю, датами та призначеними ресурсами, права відображає горизонтальні смуги завдань у часі. Критичні завдання виділено червоним кольором, некритичні позначено синім, контрольні точки відображаються як ромби. Імена ресурсів праворуч від кожної смуги дозволяють одразу бачити розподіл відповідальності та виявляти випадки одночасного призначення одного спеціаліста на кілька завдань.

Побудована модель підтверджує реалістичність плану, оскільки тривалість 38 тижнів вписується в обмеження 9 місяців, встановлене у Розділі 2. Нульовий резерв на критичному шляху та концентрація понад 52% тривалості у фазі розробки вказують на необхідність активного контролю ресурсів і ризиків, що розглядається у підрозділах 3.2 та 3.3.

3.2. Ресурсне планування та управління завантаженням команди

Для реалізації проєкту Trade&Earn сформовано команду з 12 спеціалістів, кожного з яких внесено до MS Project як окремий ресурс типу Work. Ресурси розподілено на п'ять функціональних груп: управління (PM, BA), бекенд-розробка (Backend-1 - Backend-4), фронтенд-розробка (Frontend-1, Frontend-2), дизайн (UI/UX Designer), тестування (QA-1, QA-2) та

інфраструктура (DevOps). Кожному ресурсу присвоєно тижневу ставку оплати, що відповідає ринковим даним для спеціалістів відповідного рівня. Ставки варіюються від 10 000 ₪/тиждень для QA-інженерів до 16 250 ₪/тиждень для менеджера проєкту. Таблицю ресурсів проєкту з їхніми ставками наведено у таблиці 3.1.

Таблиця 3.1.

Ресурси проєкту Trade&Earn у MS Project

№	Ресурс	Роль	Ставка (₪/тиждень)
1	2	3	4
1	PM (Менеджер проєкту)	Управління	16 250
2	BA (Бізнес-аналітик)	Аналітика	12 500
3	Backend-1	Бекенд-розробка	13 750
4	Backend-2	Бекенд-розробка	13 750
5	Backend-3	Бекенд-розробка	13 750
6	Backend-4	Бекенд-розробка	13 750
7	Frontend-1	Фронтенд-розробка	12 500
8	Frontend-2	Фронтенд-розробка	12 500
9	UI/UX Designer	Дизайн	11 250
10	QA-1	Тестування	10 000
11	QA-2	Тестування	10 000
12	DevOps	Інфраструктура	15 000
Разом			155 000

Чотири бекенд-розробники складають найбільшу групу в команді, що обумовлено складністю серверної частини криптовалютного порталу. Торговий модуль, модуль безпеки (2FA, KYC/AML), інтеграція з біржовими API та проєктування бази даних вимагають одночасної роботи кількох спеціалістів. За розрахунками у MS Project, скорочення команди бекенду до двох осіб подовжило б фазу розробки щонайменше на 4-5 тижнів через послідовне виконання завдань, які зараз виконуються паралельно.

Frontend-1 зосереджений на торговій панелі та візуалізації даних (графіки, Order Book), Frontend-2 відповідає за особистий кабінет та освітній модуль. На фазах проєктування та тестування ролі розділяються, Frontend-2 бере участь у прототипуванні UX спільно з дизайнером, а Frontend-1 виконує тестування доступності. Такий розподіл виключає ситуацію, коли обидва фронтед-розробники залежать від одного API-ендпоінту і блокують один одного.

QA-2 підключається до проєкту з четвертого спринту, коли обсяг функціоналу перевищує можливості одного тестувальника. У перших трьох спринтах QA-1 справляється самостійно, оскільки кількість завершених User Stories не перевищує 8-10 за спринт. З четвертого спринту з'являється торгова панель з численними сценаріями тестування, тому QA-2 залучається для паралельного виконання функціонального та регресійного тестування, що дозволяє не збільшувати тривалість спринтів при зростанні обсягу роботи.

UI/UX Designer активно працює на фазі проєктування (wireframes, мокапи, прототипування) та на спринтах 1 і 5, де створюються нові інтерфейсні елементи. У проміжних спринтах резерв завантаження дизайнера використовується для підготовки презентаційних матеріалів до демо замовнику або доопрацювання UI на основі зворотного зв'язку команди. DevOps-інженер задіяний протягом усього проєкту з різним рівнем завантаження. На фазі проєктування він формує технологічний стек, під час розробки підтримує CI/CD-пайплайн, а на фазі впровадження його навантаження зростає до максимуму одночасно з розгортанням продакшн-середовища та налаштуванням моніторингу.

Після призначення ресурсів на всі завдання у поданні Resource Usage виявлено кілька конфліктів перевантаження (overallocation). Конфлікти виникали, коли один спеціаліст призначався на два завдання, що виконуються паралельно. Зокрема, Backend-1 одночасно фігурував у проєктуванні серверної архітектури та проєктуванні REST API, графіки яких частково збігалися.

Аналогічна ситуація виникла з ВА, якому одночасно призначено побудову календарного плану та оцінку бюджету.

Для кожного конфлікту визначено пріоритетність завдань і прийнято рішення про зсув менш критичних робіт. Функцію Resource Leveling у MS Project застосовано для перерахунку дат з урахуванням цих рішень, щоб кожен спеціаліст виконував лише одне завдання у будь-який момент часу. Результатом вирівнювання стало подовження проєкту на 8 робочих днів, дата завершення змістилася з 14 на 22 вересня 2026 року. Це прийнятний компроміс, оскільки усунення перевантажень знижує ризик помилок через одночасне виконання кількох завдань, а загальна тривалість проєкту залишається в межах 9-місячного обмеження.

Після вирівнювання проведено перевірку подання Resource Usage, яке відображає завантаження кожного ресурсу по тижнях у годинах. Перевірка підтвердила, що жоден спеціаліст не перевищує 40 годин на тиждень. Рівень завантаження варіюється від 60% до 100% залежно від фази проєкту, а максимальне завантаження всієї команди припадає на фазу розробки (Спринти 3-7), коли одночасно працюють усі 12 спеціалістів.

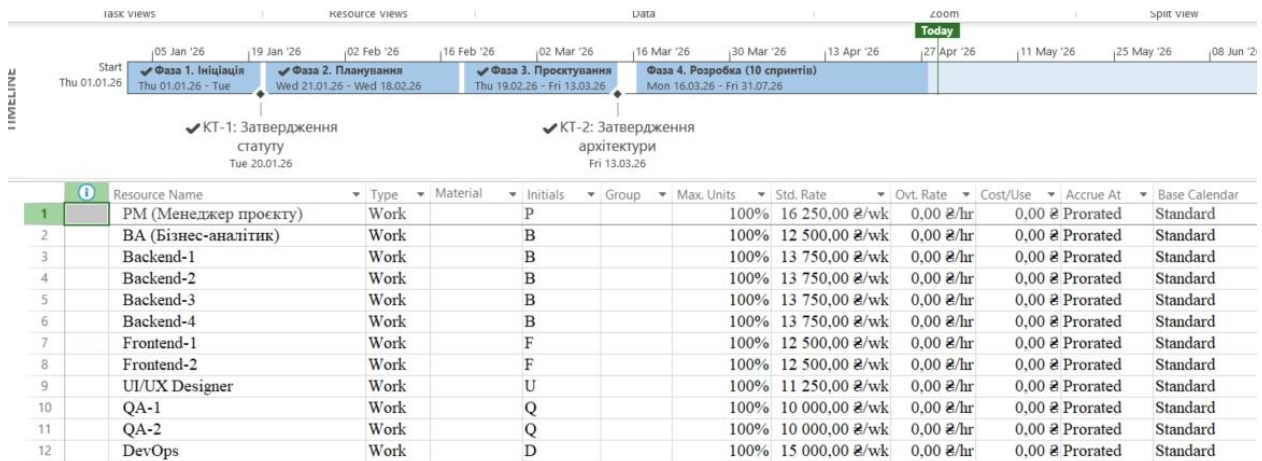


Рис. 3.6. Resource Sheet проєкту Trade&Earn у MS Project (12 ресурсів зі ставками)

На рисунку 3.6 відображено таблицю ресурсів проєкту. Для кожного ресурсу вказано тип (Work), максимальне завантаження (100%), тижневу ставку у гривнях та метод нарахування витрат (Prorated). Ставки визначено на

основі медіанних зарплат ІТ-спеціалістів в Україні, перерахованих із місячних у тижневі. На підставі цих ставок MS Project розраховує вартість кожного завдання, фази та проєкту загалом.

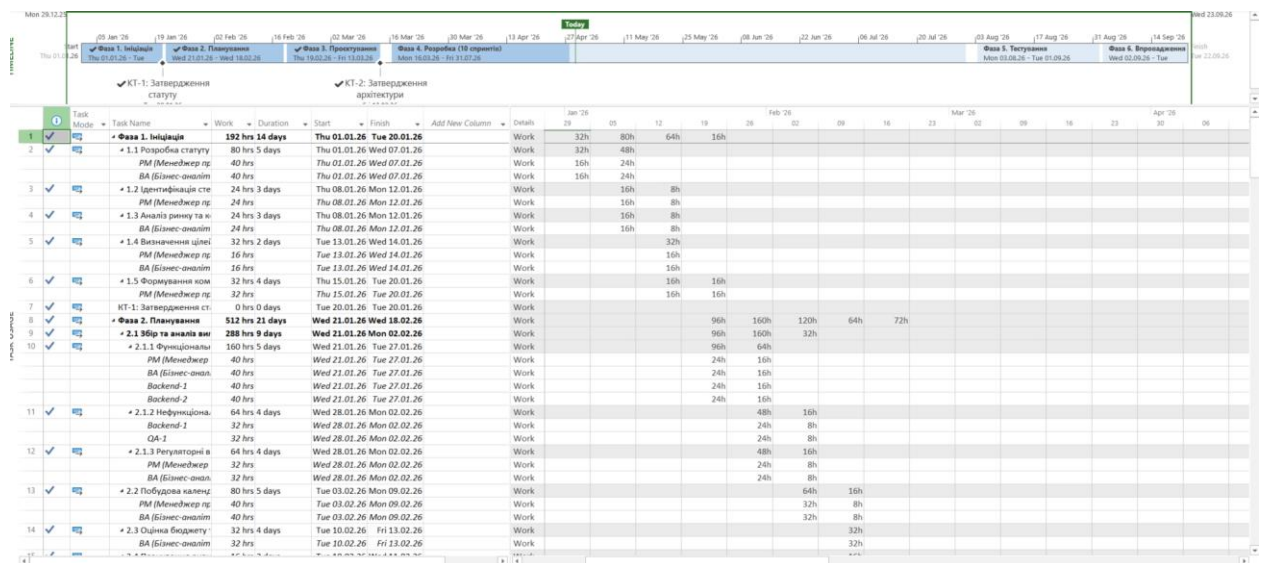


Рис. 3.7. Resource Usage проєкту Trade&Earn

На рисунку 3.7 відображено подання Resource Usage з деталізацією завантаження кожного ресурсу по тижнях. Ліва частина містить ієрархію ресурсів із переліком призначених завдань та загальною кількістю годин, права відображає розподіл годин у вигляді числової матриці. Це подання дозволяє одразу виявити тижні з пониженим або підвищеним завантаженням окремих спеціалістів.

На основі внесених ставок розраховано загальну вартість проєкту: сума склала 3 248 250 ₴, що перевищує попередню оцінку 2,5 млн грн із Розділу 2. Різниця пояснюється тим, що детальна bottom-up оцінка врахувала реальну кількість годин кожного спеціаліста на кожному завданні, тоді як top-down оцінка базувалась на усереднених показниках [3]. Виявлене перевищення бюджету на 30% є важливим результатом моделювання і підтверджує необхідність ресурсного планування із застосуванням спеціалізованого програмного забезпечення.

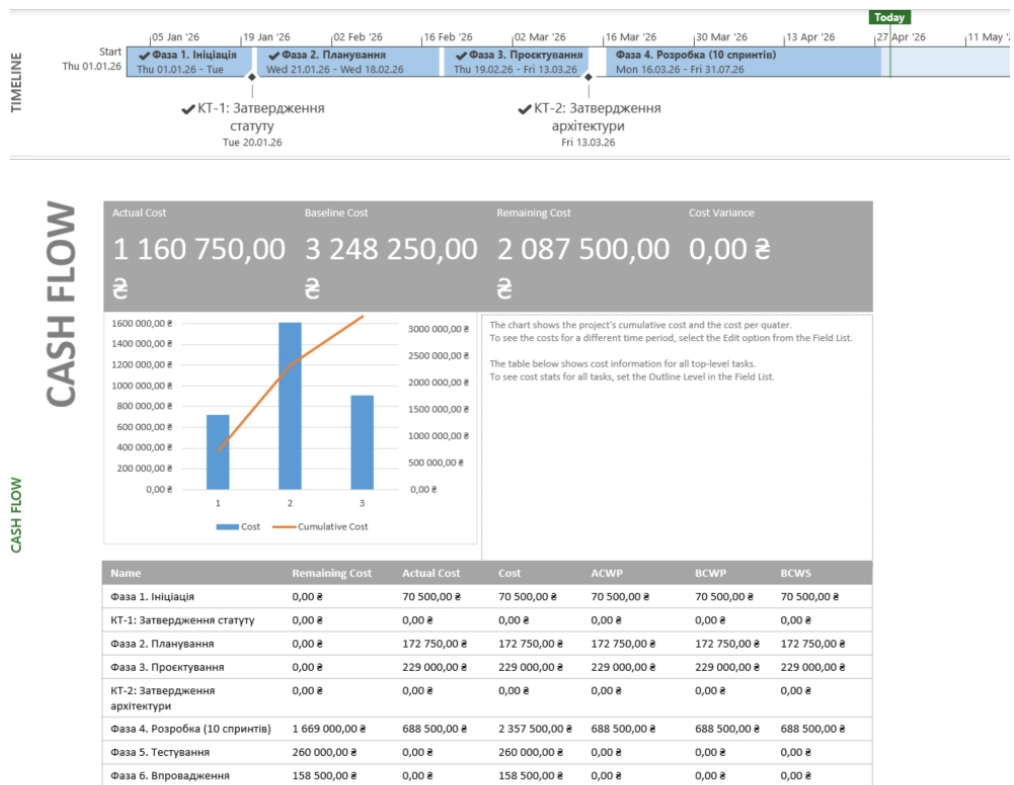


Рис. 3.8. Загальна вартість проєкту Trade & Earn

Отримане розрахункове значення вартості підтверджує, що детальне ресурсне планування у MS Project є не формальним, а необхідним управлінським інструментом: саме завдяки декомпозиції до рівня конкретних завдань і ресурсів вдалося виявити розрив між первинною оцінкою та реальною трудомісткістю ще на етапі планування, а не в процесі виконання. Це дозволяє менеджеру проєкту своєчасно скоригувати бюджетні очікування замовника та опрацювати варіанти оптимізації завантаження перед стартом фази розробки.

3.3. Управління ризиками проєкту

Управління ризиками є обов'язковою складовою планування будь-якого IT-проєкту. Для Trade&Earn ризики ідентифіковано на етапі планування методом мозкового штурму за участю менеджера проєкту, бізнес-аналітика та технічного ліда (Backend-1). Кожен ризик оцінено за двома параметрами, ймовірністю настання та потенційним впливом на проєкт за п'ятибальною шкалою[20]. Добуток цих двох оцінок формує інтегральний показник ризику

(Risk Score), який визначає пріоритетність реагування. Шкали оцінювання наведено у таблицях 3.2 та 3.3.

Таблиця 3.2.

Шкала оцінювання ймовірності ризику

Бал	Рівень	Інтерпретація
1	2	3
1	Дуже низька	Менше 10%
2	Низька	10-30%
3	Середня	30-50%
4	Висока	50-70%
5	Дуже висока	Понад 70%

Таблиця 3.3.

Шкала оцінювання впливу ризику

Бал	Рівень	Затримка	Перевитрата бюджету
1	2	3	4
1	Незначний	До 3 днів	До 2%
2	Помірний	3-7 днів	2-5%
3	Значний	1-2 тижні	5-10%
4	Серйозний	2-4 тижні	10-20%
5	Критичний	Понад місяць	Понад 20%

За результатами ідентифікації сформовано реєстр із 10 ризиків, наведений у таблиці 3.4.

Таблиця 3.4.

Реєстр ризиків проєкту Trade&Earn

ID	Ризик	Категорія	Ймовірність	Вплив	Risk Score	Пріоритет
1	2	3	4	5	6	7
R1	Зміна регуляторних вимог до КУС/AML під час розробки	Зовнішній	3	5	15	Критичний
R2	Звільнення ключового бекенд-розробника (Backend-1)	Ресурсний	3	4	12	Високий

R3	Збій або зміна API зовнішніх бірж (CoinGecko, Binance)	Технічний	4	3	12	Високий
R4	Недостатня продуктивність торгового модуля під навантаженням	Технічний	3	4	12	Високий
R5	Розширення обсягу робіт за ініціативою замовника (scope creep)	Управлінський	4	3	12	Високий
R6	Вразливість безпеки, виявлена під час penetration testing	Технічний	3	3	9	Середній
R7	Затримка інтеграції з KYC-провайдером через технічні обмеження	Технічний	3	3	9	Середній
R8	Перевищення бюджету понад 3,25 млн грн	Фінансовий	2	4	8	Середній
R9	Низька якість UX, виявлена під час UAT	Якість	2	3	6	Низький
R10	Недоступність хмарної інфраструктури під час розгортання	Інфраструктурний	1	4	4	Низький

Ризики з Risk Score 12 і вище віднесено до категорії критичних та високих і потребують детальних планів реагування. Ризики з оцінкою 6-9 підлягають моніторингу без окремих превентивних заходів. Ризики нижче 6

прийнято (acceptance), оскільки витрати на їхнє запобігання перевищують потенційні збитки.

Для візуалізації розподілу ризиків побудовано матрицю ймовірність × вплив (таблиця 3.5), яка наочно показує, які ризики потрапляють у червону зону (негайне реагування), жовту (моніторинг) та зелену (прийнятні).

Таблиця 3.5.

Матриця ймовірність × вплив проєкту Trade&Earn

Ймовірність / Вплив	1 (Незначний)	2 (Помірний)	3 (Значний)	4 (Серйозний)	5 (Критичний)
1	2	3	4	5	6
5 (Дуже висока)					
4 (Висока)			R3, R5		
3 (Середня)			R6, R7	R2, R4	R1
2 (Низька)			R9	R8	
1 (Дуже низька)				R10	

З матриці видно, що найнебезпечнішим є ризик R1, який поєднує середню ймовірність із критичним впливом. Чотири ризики (R2, R3, R4, R5) мають однаковий Risk Score 12, проте потребують різних стратегій реагування. Зелена зона (R9, R10) не містить ризиків, що безпосередньо загрожують термінам або бюджету проєкту.

Нижче наведено плани реагування на п'ять найкритичніших ризиків.

R1. Зміна регуляторних вимог до KYC/AML (Risk Score: 15)

Криптовалютна галузь перебуває у стадії активного регулювання. Регламент MiCA (Markets in Crypto-Assets) набув чинності у 2024 році, а Україна продовжує формувати власне законодавство щодо віртуальних активів[21]. Існує ймовірність, що протягом 9 місяців розробки з'являться нові вимоги до верифікації користувачів або звітності. Вплив оцінено як критичний, оскільки зміна вимог може зачіпати одночасно бекенд, фронтенд і базу даних.

Стратегія реагування - зниження (mitigation). Модуль KYC/AML спроектовано як окремий мікросервіс із взаємодією через REST API, що дозволяє змінити KYC-провайдера або додати нові перевірки без переписування торгового модуля. Бізнес-аналітик щомісячно відстежує законодавчі ініціативи НКЦПФР та Мінцифри. У разі появи нових вимог після КТ-3 (22 травня) ініціюється Change Request із перерахунком бюджету та термінів.

R2. Звільнення ключового бекенд-розробника (Risk Score: 12)

Backend-1 є технічним лідом і єдиним спеціалістом, задіяним на всіх шести фазах проєкту. Його звільнення під час розробки спричинить затримку мінімум на 2-3 тижні через пошук та адаптацію заміни, а також ризик втрати знань про архітектурні рішення.

Стратегія реагування - зниження (mitigation). Для критичних компонентів (торговий модуль, модуль безпеки) застосовується практика парного програмування, щоб знання розподілялись між Backend-1 та Backend-2. Архітектурні рішення фіксуються у Confluence відразу після затвердження, а код-рев'ю є обов'язковим для кожного pull request. Як резервний план, підтримується контакт із двома зовнішніми кандидатами рівня Middle+, залучення яких можливе протягом двох тижнів.

R3. Збій або зміна API зовнішніх бірж (Risk Score: 12)

Торговий модуль залежить від зовнішніх API для отримання котирувань у реальному часі. CoinGecko, Binance та інші провайдери періодично змінюють формати відповідей або тимчасово припиняють роботу сервісів, а за статистикою CoinGecko API оновлюється 2-3 рази на рік.

Стратегія реагування - зниження та планування обхідних рішень (mitigation + workaround). Архітектура торгового модуля передбачає шар абстракції (Adapter Pattern), тому при зміні API достатньо оновити один адаптер. Додатково підключено двох незалежних провайдерів котирувань із автоматичним перемиканням при збої, а кешування забезпечує роботу торгової панелі протягом 5 хвилин навіть при повній недоступності зовнішніх API.

R4. Недостатня продуктивність торгового модуля під навантаженням (Risk Score: 12)

Торговий модуль повинен обробляти одночасні запити від кількох сотень користувачів. Виявлення проблем з продуктивністю лише на фазі тестування (Спринт 9) зробило б переробку коду дорогою та ризикованою і могло затримати запуск на 2-4 тижні.

Стратегія реагування - запобігання (avoidance). Навантажувальне тестування окремих компонентів розпочинається зі Спринту 5, а не відкладається до фази тестування, що дозволяє виявляти вузькі місця, коли вартість виправлення мінімальна. DevOps налаштовує стрес-тести у CI/CD-пайплайні для автоматичної перевірки кожної збірки на регресію продуктивності. Для Order Matching Engine передбачено горизонтальне масштабування.

R5. Розширення обсягу робіт (scope creep) (Risk Score: 12)

Розширення обсягу робіт є одним із найпоширеніших ризиків у IT-проєктах[33]. Після КТ-3, коли замовник вперше побачить працюючий прототип, існує висока ймовірність запиту нових функцій (мобільний додаток, підтримка DeFi-протоколів), особливо з огляду на швидкі зміни криптовалютного ринку.

Стратегія реагування - зниження та передача (mitigation + transfer). Гібридна методологія проєкту передбачає оформлення будь-якого нового запиту як Change Request із обов'язковою оцінкою впливу на бюджет і терміни, а рішення про включення приймається на Sprint Planning за участю Product Owner. Контракт із замовником фіксує обсяг Must Have вимог, а всі доповнення оплачуються окремо.

Моніторинг ризиків здійснюється на щотижневих статус-мітингах. Для кожного ризику категорій «Критичний» та «Високий» визначено конкретну подію, що сигналізує про наближення ризику. Тригером для R1 є публікація проєкту нормативного акту на сайті НКЦПФР, для R4 це перевищення часу відгуку API у 200 мс під час тестування. Реєстр ризиків переглядається після

кожної контрольної точки з переоцінкою ймовірностей на основі актуальної інформації.

3.4. Технічна архітектура порталу Trade&Earn як артефакт проєктного планування

На фазі проєктування розроблено архітектурну схему порталу Trade&Earn, яка є ключовим артефактом, що фіксує технічні рішення перед початком розробки та слугує основою для декомпозиції завдань у спринтах. Без затвердженої архітектури команда не може обґрунтовано оцінювати трудомісткість завдань і розподіляти відповідальність між спеціалістами, тому розробка схеми виконується на підготовчому етапі і фіксується на КТ-2 (13 березня). Архітектура побудована за принципом розділення відповідальності (Separation of Concerns) і складається з п'яти логічних шарів. Клієнтський шар, API Gateway, шар бекенд-сервісів, шар даних та зовнішні інтеграції. Таке структурування дозволило чітко розмежувати зони відповідальності між фронтенд-розробниками, бекенд-командою та DevOps-інженером і вести паралельну розробку без взаємних блокувань. Архітектурну схему порталу Trade&Earn наведено на рисунку 3.9.

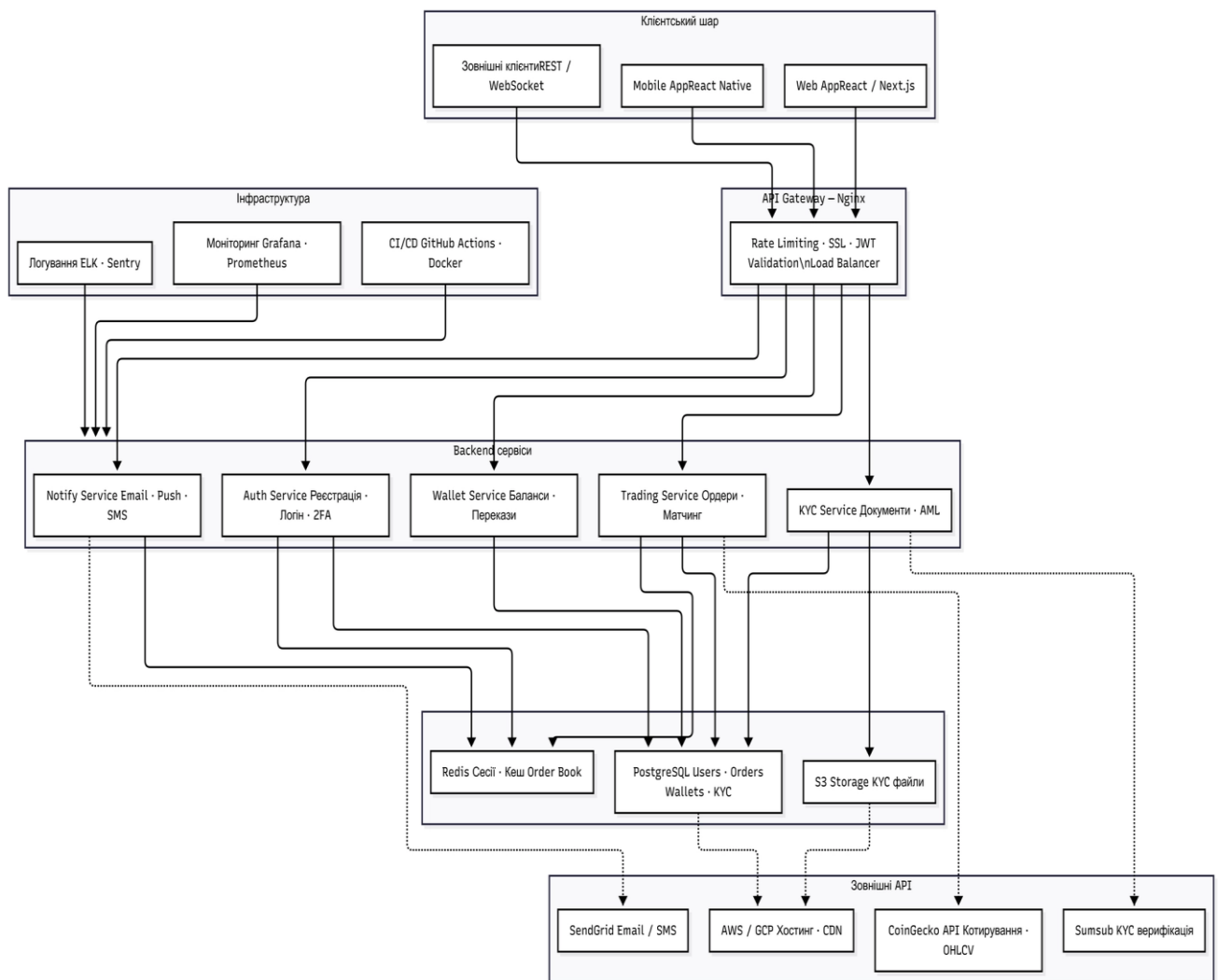


Рис. 3.9. Архітектура порталу Trade&Earn

Найважливішим рішенням з точки зору управління проектом стала декомпозиція бекенду на п'ять незалежних сервісів: Auth Service, Trading Service, Wallet Service, KYC Service та Notify Service. Кожен сервіс може розроблятися, тестуватися та змінюватися незалежно від інших, що усуває ситуацію, коли блокування одного завдання зупиняє роботу всієї команди[22]. Це рішення безпосередньо визначило організацію спринтів у MS Project. Спринт 1 зосереджений виключно на Auth Service, Спринти 3-5 на Trading Service, Спринти 6-7 на KYC Service. Якщо один сервіс потребує додаткового часу, решта команди продовжує роботу над іншими компонентами, що знижує ризик каскадних затримок на критичному шляху.

Для клієнтського шару обрано єдиний технологічний стек на основі React-екосистеми для веб та мобільного додатка. З управлінської точки зору це

рішення скорочує потребу в спеціалізованих ресурсах, обидва фронтенд-розробники можуть працювати над будь-яким клієнтом без додаткового навчання, що підвищує взаємозамінність у команді та знижує ресурсний ризик R2. Усі запити від клієнтів проходять через єдину точку входу - API Gateway, що дозволяє DevOps-інженеру налаштовувати безпеку та масштабування централізовано без залучення бекенд-розробників і скорочує координаційні витрати[24].

Шар даних складається з трьох компонентів, кожен з яких обрано для конкретного типу навантаження. PostgreSQL є основною реляційною СУБД і зберігає дані користувачів, ордерів та гаманців. Redis використовується як кеш котирувань та буфер для Order Book завдяки мінімальній затримці. S3 Storage зберігає файли KYC-документів, які через великий обсяг не підходять для реляційної бази. Поділ на три спеціалізованих сховища збільшив обсяг роботи на Фазі 3. Backend-2 отримав окремі завдання на проектування кожного компонента, але забезпечив горизонтальне масштабування кожного шару незалежно від інших.

Модульна архітектура також безпосередньо знижує ризики, зафіксовані у реєстрі підрозділу 3.3. KYC Service виокремлено в окремий мікросервіс, тому зміна регуляторних вимог (ризик R1) не потребує переписування торгового модуля. Для захисту від збоїв зовнішніх API (ризик R3) передбачено шар абстракції (Adapter Pattern) та резервні провайдери котирувань із автоматичним перемиканням[23]. Таким чином, архітектурні рішення та реєстр ризиків є взаємопов'язаними документами, що формуються паралельно на одній фазі проєкту.

Обрана архітектура визначила склад команди та розподіл завдань у MS Project. Backend-1 відповідає за Trading Service та REST API, Backend-2 за базу даних та Wallet Service, Backend-3 за Auth Service та модулі безпеки, Backend-4 за зовнішні інтеграції та KYC Service. Frontend-1 реалізує торгову панель, Frontend-2 особистий кабінет та освітній модуль, DevOps-інженер забезпечує інфраструктурний шар та CI/CD-пайплайн. Таке відображення архітектурних

компонентів на ролі команди гарантує, що кожен модуль має відповідального виконавця і жоден компонент не залишається без власника.

3.5. Проєктування структури бази даних порталу

Проєктування структури бази даних є одним із тих завдань, де помилки на ранньому етапі обходяться найдорожче: зміна схеми після запуску розробки тягне за собою переписування міграцій, правки у сервісах і повторне тестування. Саме тому ER-діаграма розроблена на Фазі 3 і є обов'язковим артефактом, без якого неможливо коректно оцінити трудомісткість бекенд-завдань[25]. Роботу виконав Backend-2 відповідно до розподілу ролей, описаного у підрозділі 3.2. Модель містить 10 таблиць, згрупованих у чотири блоки відповідно до бекенд-сервісів архітектури з підрозділу 3.4. ER-діаграму бази даних порталу Trade&Earn наведено на рисунку 3.10.

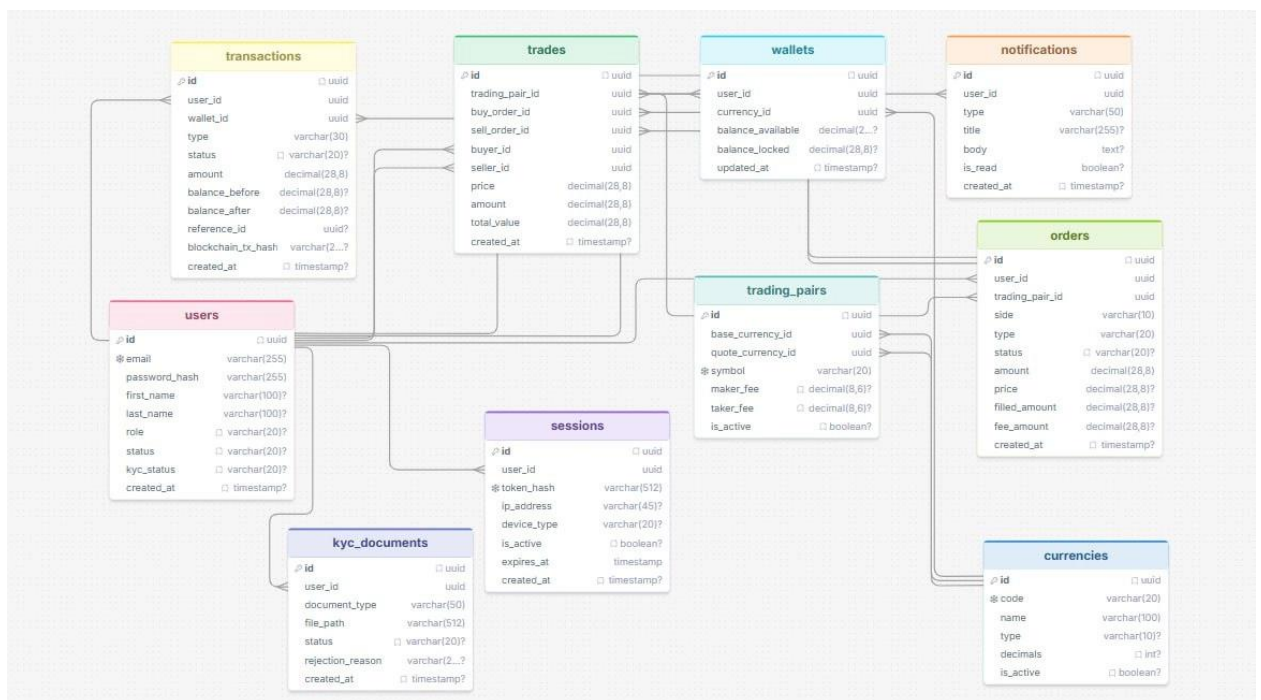


Рис. 3.10. ER-діаграма бази даних порталу Trade&Earn (10 таблиць)

Таблиці згруповано за сервісами, а не за технічними ознаками, це рішення, яке безпосередньо вплинуло на організацію роботи команди. Backend-2 працює виключно з таблицями Wallet Service (wallets, transactions), Backend-3 з Auth Service (users, sessions, kyc_documents), Backend-1 з Trading

Service (orders, trades, trading_pairs, currencies). Кожен розробник знає межі своєї зони і не заходить у чужу без узгодження. На практиці це скоротило кількість конфліктів злиття у Git і спростило код-рев'ю. Рецензент перевіряє зміни в межах одного функціонального блоку, а не намагається утримати в голові всю схему цілком.

Побудова бази даних розбита на п'ять міграційних хвиль, кожна з яких прив'язана до конкретного спринту. У Спринті 1 створюються таблиці users, sessions та wallets, це мінімальний набір для роботи Auth Service. У Спринті 3 до них додаються currencies, trading_pairs та orders. У Спринті 5, коли торговий модуль досягає етапу матчінгу, з'являються trades та transactions. Спринти 6 та 8 завершують схему таблицями kus_documents та notifications відповідно. Такий підхід дає QA-інженерам можливість тестувати кожен блок одразу після спринту, а не чекати на повну готовність усіх 10 таблиць. Виявлені помилки локалізовані у свіжих змінах, а не розкидані по всій схемі.

Кілька конкретних рішень щодо структури таблиць прийнято під час KT-2 після уточнення вимог замовника. Баланс у таблиці wallets розділено на два поля доступний та заблокований в активних ордерах, бо без цього торговий модуль не може коректно перевіряти достатність коштів при виставленні ордера. Таблиця transactions фіксує стан балансу до і після кожної операції, ця вимога з'явилась після того, як замовник запросив можливість повного аудиту фінансових переміщень. Обидва уточнення збільшили оцінку трудомісткості Backend-2 на три дні, що було відображено у MS Project як коригування тривалості відповідних завдань без впливу на загальний графік.

Усі первинні ключі мають тип UUID, а між таблицями встановлено зовнішні ключі на рівні СУБД. UUID унеможливорює передбачення ідентифікаторів записів, це стандартна вимога безпеки для фінансових платформ. Зовнішні ключі гарантують цілісність даних, видалити користувача з активними ордерами або ненульовим балансом неможливо на рівні бази даних, а не лише на рівні коду. Це знижує ризик виникнення «зламаних»

записів при помилках у логіці застосунку і зменшує навантаження на QA, тому що частину перевірок виконує сама СУБД.

3.6. Мокапи інтерфейсу як результат етапу UI/UX-проектування

Затвердження мокапів до старту розробки є обов'язковою умовою гібридної методології. Без візуальних специфікацій кожен спринт витрачав би 1-2 дні двотижневого циклу на узгодження дизайну, що безпосередньо впливало б на виконання графіка у MS Project.

Прототипування виконано UI/UX Designer на Фазі 3 (завдання 3.3). Мокапи затверджено замовником на КТ-2 (13 березня), після чого фронтенд-розробники отримали можливість розпочати реалізацію з першого спринту. Дизайн-система побудована на темній кольоровій схемі з акцентним зеленим. Темний інтерфейс знижує навантаження на зір для трейдерів, які проводять за екраном 4-8 годин на добу, а зелений колір є стандартним позначенням прибутку на фінансових ринках. Єдиний стиль забезпечує консистентність між п'ятьма основними екранами порталу. Мокапи п'яти основних екранів наведено на рисунках 3.11-3.14.

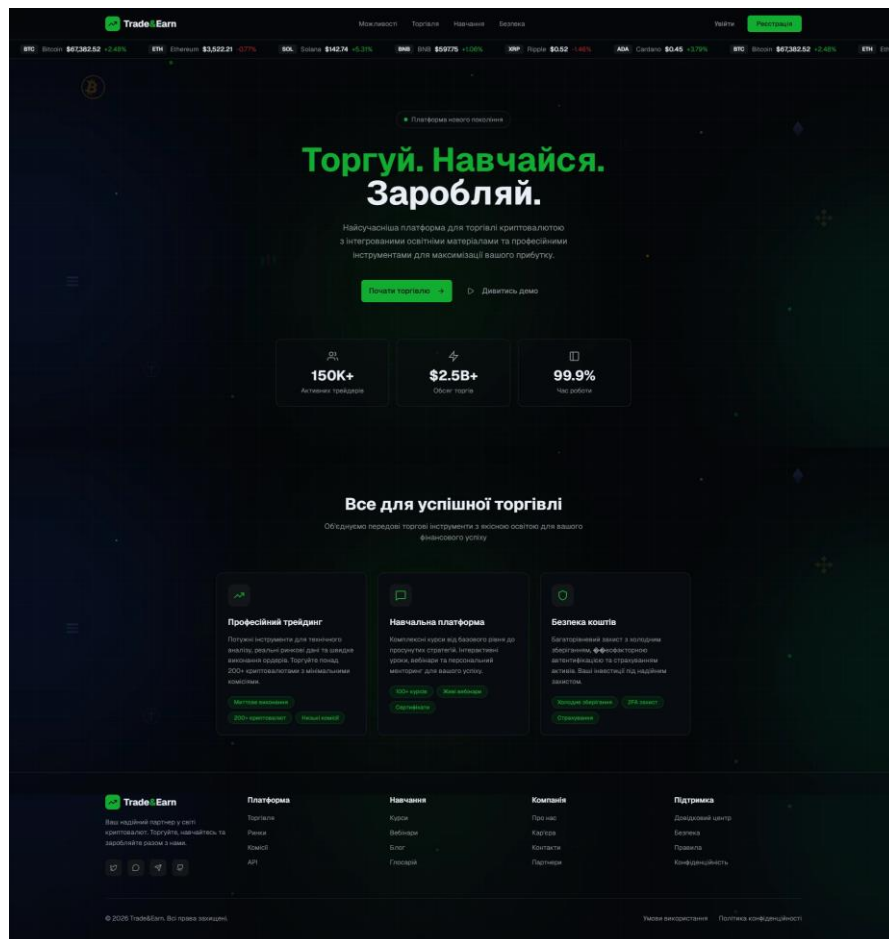


Рис. 3.11. Мокап головної сторінки порталу Trade&Earn

Головна сторінка (рис. 3.11) орієнтована на залучення нових користувачів. У верхній частині розміщено тікер із котируваннями у реальному часі. Центральний блок містить слоган «Торгуй. Навчайся. Заробляй.», СТА-кнопки «Почати торгівлю» та «Дивитись демо», а також три метрики довіри, 150K+ активних трейдерів, \$2.5B+ обсяг торгів, 99.9% часу роботи. Нижче представлено три ціннісні пропозиції порталу, професійний трейдинг, навчальна платформа та безпека коштів. Структура відповідає патерну F-подібного сканування, що відповідає природному руху очей при читанні веб-контенту[26].

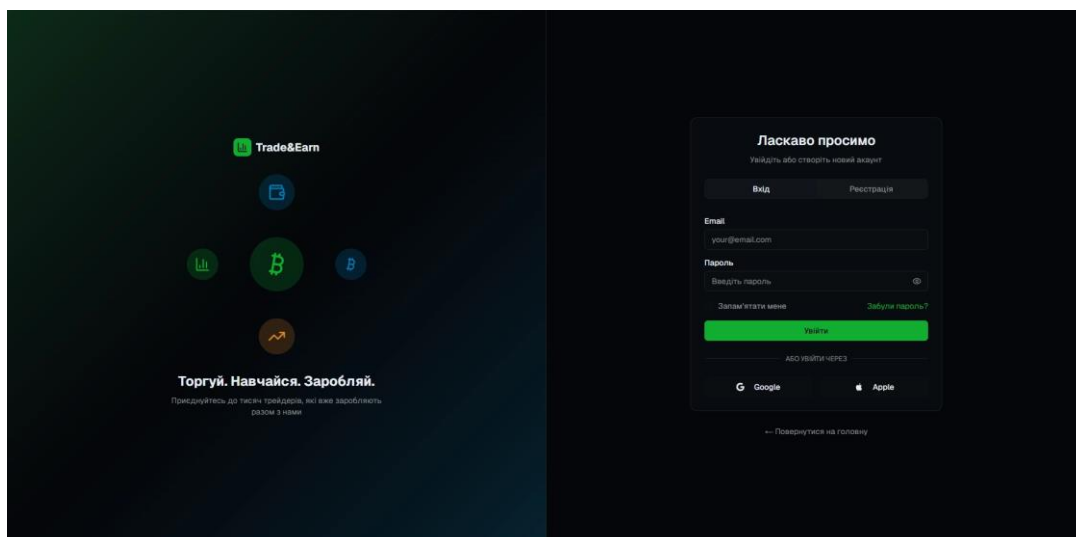


Рис. 3.12. Мокап сторінки реєстрації та входу

Сторінка автентифікації (рис. 3.12) використовує двопанельний макет, де ліва частина, це брендинг порталу, а права, це форма входу з перемикачем «Вхід / Реєстрація». Форма містить поля email та пароля, кнопку «Увійти», посилання «Забули пароль?» та OAuth-входи через Google і Apple. Підтримка OAuth знижує бар'єр реєстрації, за даними UX-досліджень, це підвищує конверсію на 20-30%.

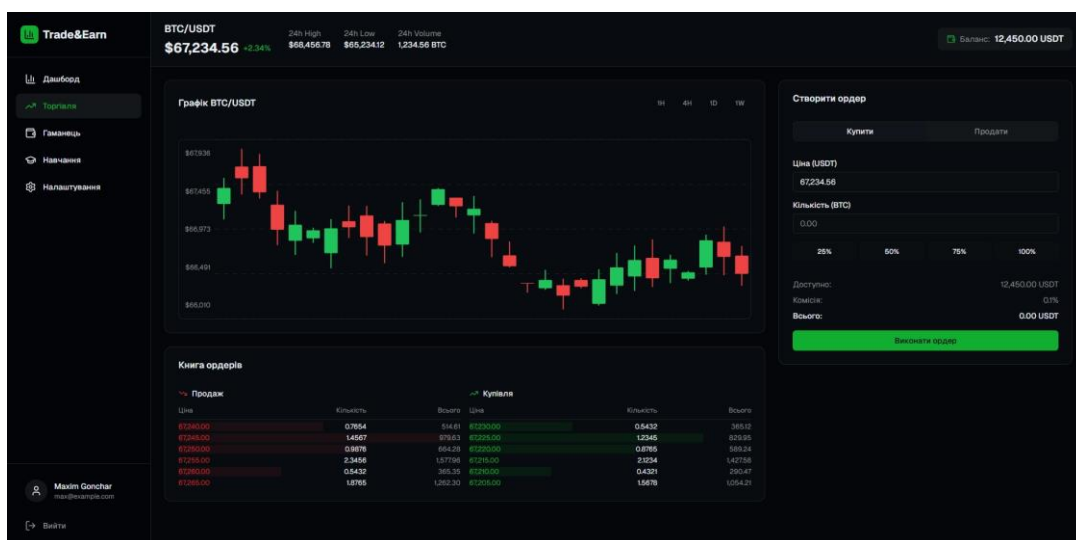


Рис. 3.13. Мокап торгової панелі (Trading Dashboard)

Торгова панель (рис. 3.13) є основним робочим екраном і складається з чотирьох зон. Ліва панель, це навігація між розділами (Дашборд, Торгівля, Гаманець, Навчання, Налаштування). Центральна зона, це графік BTC/USDT (candlestick) з перемикачем таймфреймів (1H, 4H, 1D, 1W) та поточними

показниками, ціна, зміна за 24 год, максимум, мінімум, обсяг. Права панель, це форма ордеру (Купити / Продати) з полями ціни, кількості та кнопками швидкого вибору обсягу (25/50/75/100%). Нижня зона, це Order Book з ордерами на продаж (червоні) та купівлю (зелені), що відображає глибину ринку.

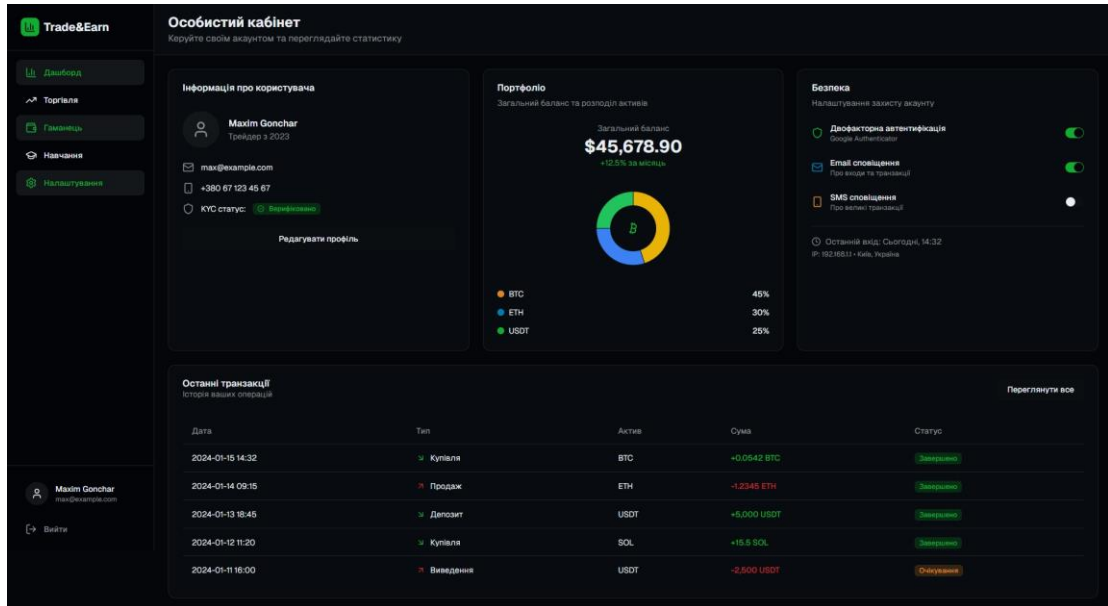


Рис. 3.14. Мокап особистого кабінету

Особистий кабінет (рис. 3.14) містить три блоки. «Інформація про користувача», ім'я, email, телефон, KYC-статус. «Портфоліо», загальний баланс (\$45,678.90), динаміка за місяць (+12.5%) та кільцева діаграма розподілу активів (BTC 45%, ETH 30%, USDT 25%). «Безпека», перемикачі 2FA (Google Authenticator), email та SMS-сповіщень, інформація про останній вхід. У нижній частині є таблиця транзакцій (дата, тип, актив, сума, статус). Об'єднання KYC-статусу і налаштувань безпеки на одному екрані мінімізує переходи між сторінками.

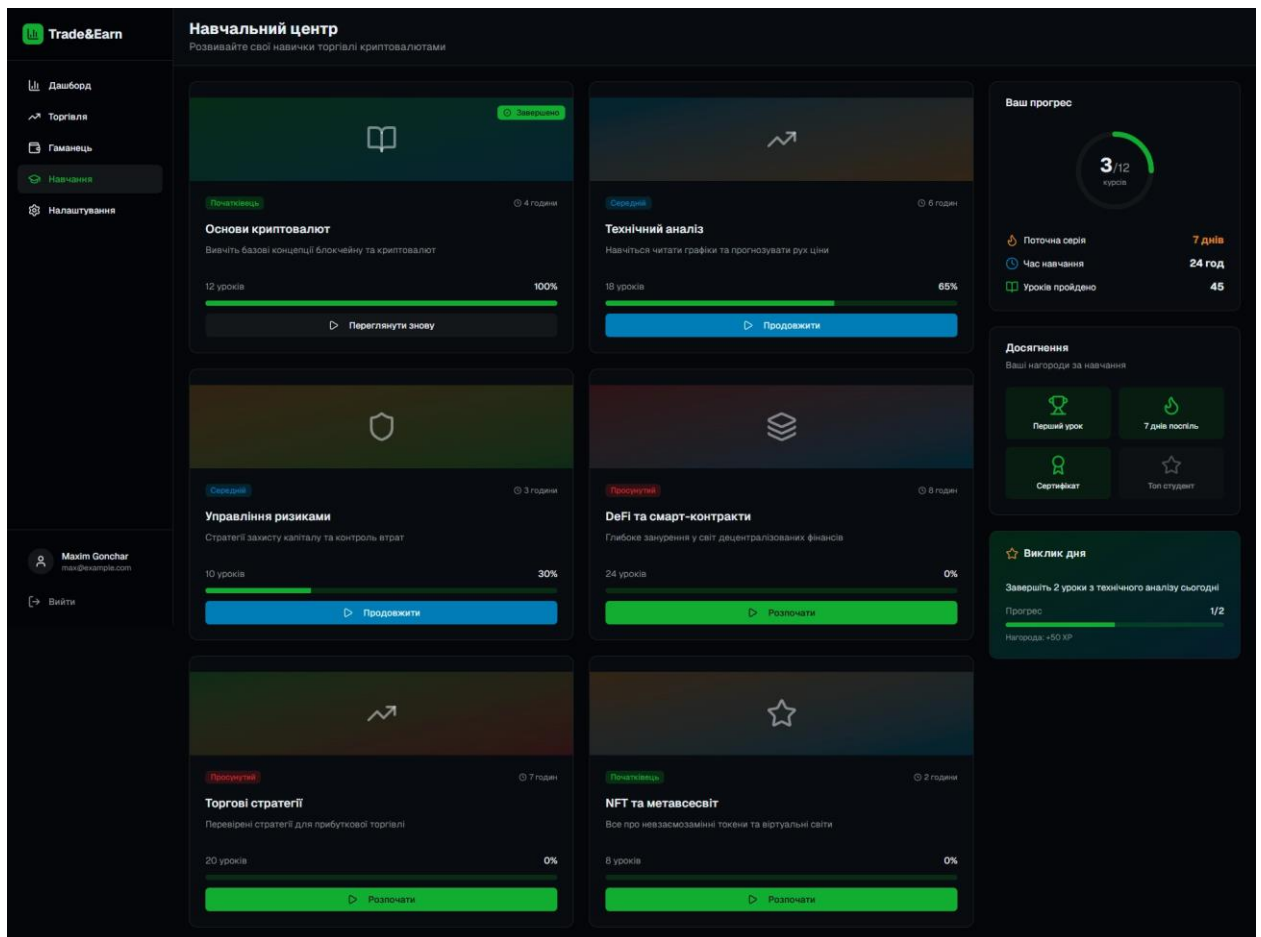


Рис. 3.15. Мокап освітнього модуля (Навчальний центр)

Освітній модуль (рис. 3.15) відрізняє Trade&Earn від типових криптовалютних бірж. Шість курсів подано у вигляді карток із мітками рівня складності, тривалістю, прогрес-баром та кнопкою дії. Права панель відображає персональну статистику. Прогрес (3 з 12 курсів), поточну серію (7 днів), загальний час (24 год) та кількість пройдених уроків (45). Елементи гейміфікації бейджі та «Виклик дня» є стандартною практикою для підвищення утримання користувачів на освітніх платформах.

Затверджені мокапи визначили розподіл роботи між фронтенд-розробниками відповідно до принципу, описаного у підрозділі 3.2. Frontend-1 реалізує торгову панель (рис. 3.10), зокрема графік свічок, форму ордера та Order Book, найскладніший екран через інтеграцію з WebSocket. Frontend-2 відповідає за особистий кабінет (рис. 3.11), освітній модуль (рис. 3.12) та сторінки автентифікації (рис. 3.9). Головна сторінка розподілена між обома, Frontend-1 реалізує тікер котирувань, Frontend-2 решту лендінгу.

3.7. Блок-схеми ключових алгоритмів обробки даних

Формалізація бізнес-процесів є управлінським інструментом фази проєктування, затверджені блок-схеми фіксують score кожного модуля, усувають неоднозначність технічного завдання та слугують підставою для формування тест-кейсів QA-інженерами.

На Фазі 3 (завдання 3.1) бізнес-аналітик формалізував три ключові процеси порталу Trade&Earn у вигляді блок-схем (flowcharts). Діаграми описують послідовність дій користувача і системи, точки прийняття рішень та альтернативні гілки виконання. Без формалізованих процесів кожен розробник інтерпретував би бізнес-логіку самостійно, що призвело б до неузгодженостей між модулями при інтеграційному тестуванні. Процес реєстрації та KYC-верифікації, блок-схему процесу наведено на рисунку 3.16.

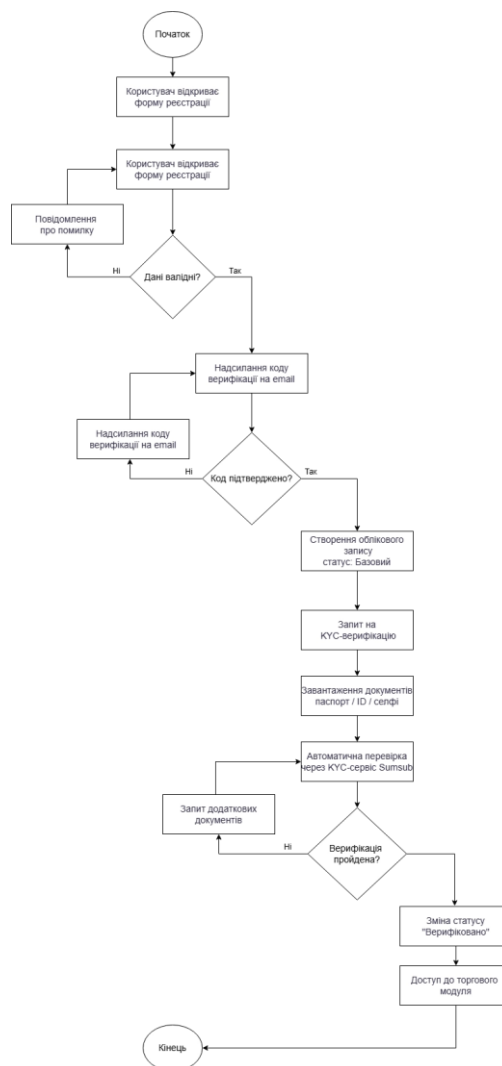


Рис. 3.16. Блок-схема процесу реєстрації та KYC-верифікації

Процес складається з двох послідовних етапів. Створення облікового запису (валідація даних, підтвердження email) та KYC-верифікації через зовнішній сервіс Sumsb. Після успішного проходження статус облікового запису змінюється з «Базовий» на «Верифіковано», що відкриває доступ до торгового модуля. Процес безпосередньо пов'язаний із таблицями users (поле kyc_status) та kyc_documents (поля status, rejection_reason) з підрозділу 3.5.

З точки зору планування, процес розподілено між двома спринтами. У Спринті 1 Backend-3 реалізує реєстрацію та підтвердження email, у Спринті 6 Backend-4 додає інтеграцію з Sumsb. Такий поділ дозволяє тестувати Auth Service незалежно від готовності KYC-модуля. Процес створення та виконання торгового ордеру блок-схему процесу наведено на рисунку 3.17.

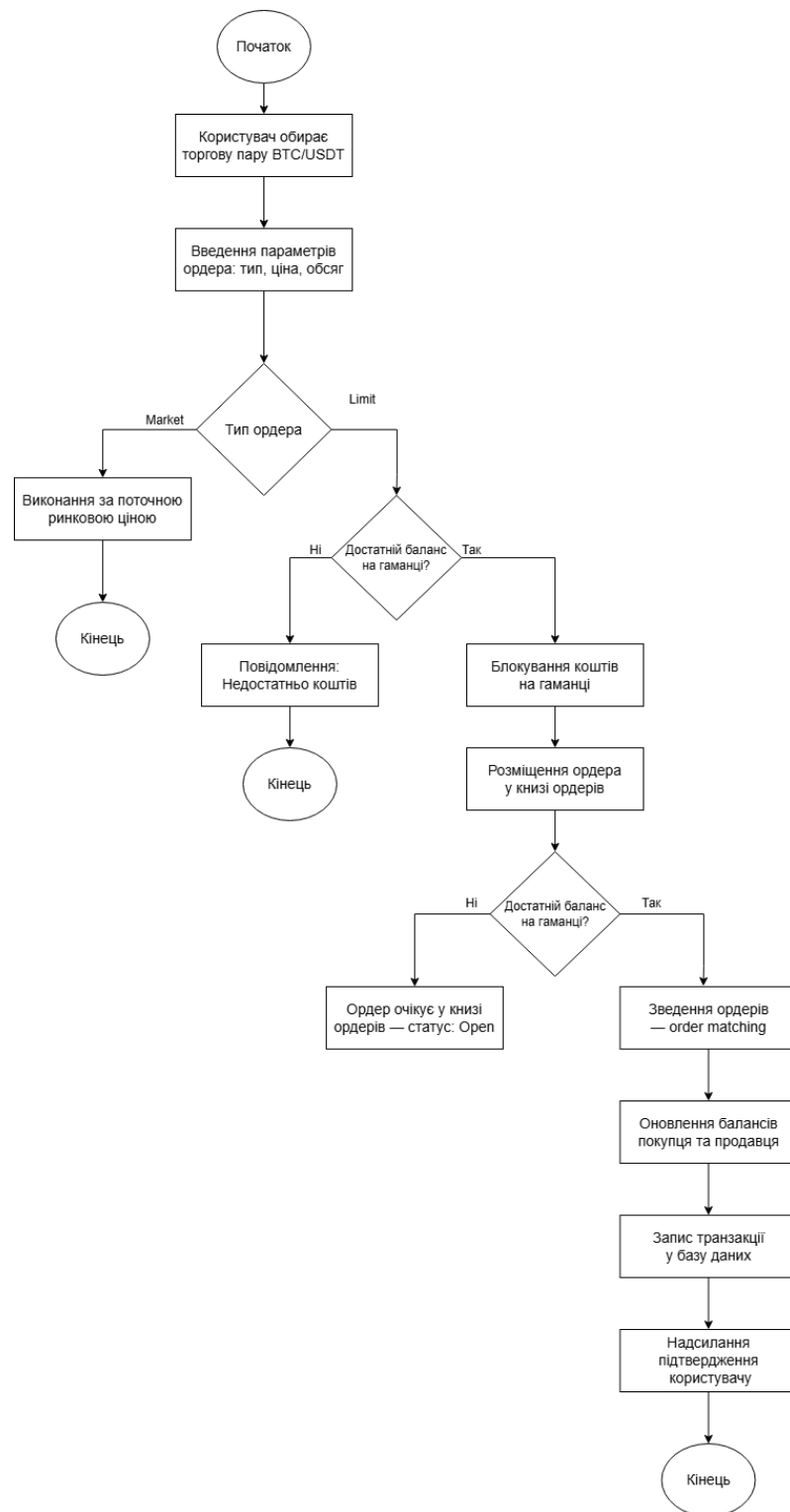


Рис. 3.17. Блок-схема процесу створення та виконання торгового ордеру

Процес розгалужується залежно від типу ордеру. Market-ордер виконується негайно за ринковою ціною. Limit-ордер передбачає перевірку балансу, блокування відповідної суми (переміщення з `balance_available` до `balance_locked`, як описано у підрозділі 3.5) та розміщення в Order Book до появи зустрічного ордеру. Після зведення (`order matching`) система оновлює

баланси, фіксує записи у таблицях `trades` і `transactions` та надсилає підтвердження через `Notify Service`.

Цей процес є найскладнішим технічно і реалізується у Спринтах 3-5 силами `Backend-1`. Продуктивність алгоритму `matching` є предметом ризику R4, тому навантажувальне тестування торгового модуля передбачено вже з 5-го спринту до завершення розробки. Процес двофакторної автентифікації (2FA), блок-схему процесу наведено на рисунку 3.18.

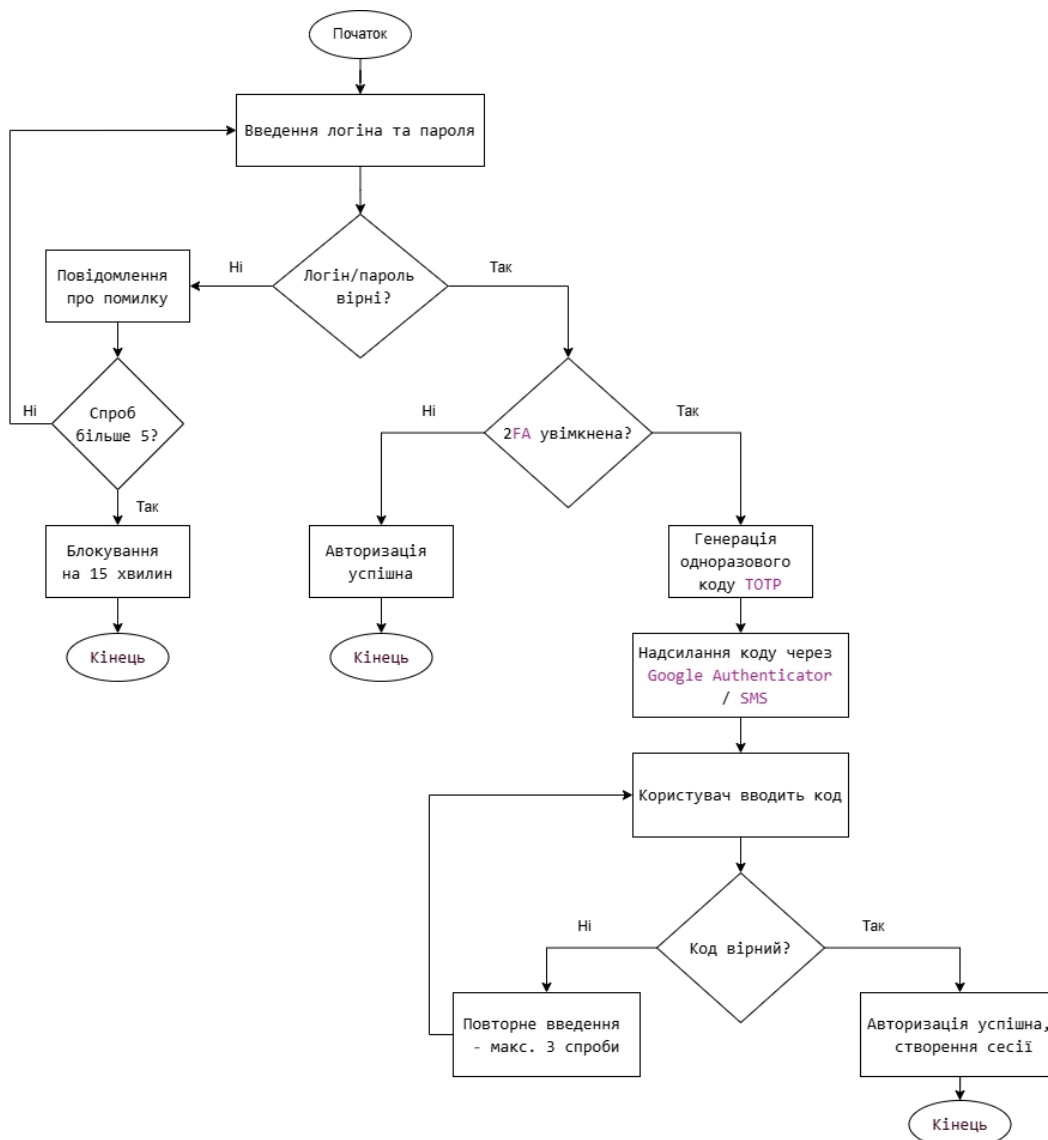


Рис. 3.18. Блок-схема процесу двофакторної автентифікації (2FA)

Після успішної перевірки логіна і пароля система визначає, чи активована 2FA. За її наявності генерується TOTP-код (Google Authenticator або SMS). При п'яти невдалих спробах входу акаунт блокується на 15 хвилин, це захист від brute-force атак, реалізований на рівні таблиці `sessions` та `Auth`

Service. Процес реалізується у Спринті 2 (Backend-3) і є обов'язковою вимогою для подальшої інтеграції з КУС-провайдером.

Із трьох блок-схем QA-інженери сформували 28 тест-кейсів для функціонального тестування, де кожна гілка діаграми (успішний шлях, помилка валідації, недостатній баланс, блокування акаунту) відповідає окремому тестовому сценарію. Це скоротило час підготовки тестової документації та забезпечило повне покриття бізнес-логіки трьох модулів.

3.8. Управління проєктом на стадії виконання: процес спринтів, standup, ретроспективи

Візуалізація ходу спринтів у Trello є оперативним інструментом контролю відхилень. Стан дошки на будь-який момент показує, які завдання виконуються, де виникають затримки та чи відповідає темп роботи плановій продуктивності команди.

Для оперативного управління спринтами використовується Trello як Scrum-дошка[27]. Кожен спринт має окрему дошку з п'ятьма стовпцями: Беклог, До виконання, У виконанні, Перевірка коду та Завершено. Картки позначено кольоровими мітками за ролями (бекенд, фронтенд, тестування, інфраструктура, дизайн, документація) та унікальним ідентифікатором TE-XXX, де перша цифра відповідає номеру спринту. Така система дозволяє одразу бачити, які ролі задіяні на кожному етапі та де виникають черги.

Для демонстрації Scrum-процесу розглянуто два спринти. Спринт 3 (початок торгового модуля) та Спринт 5 (його завершення). Обрано саме ці спринти, оскільки вони охоплюють найскладнішу частину проєкту, що лежить на критичному шляху і є предметом ризику R4 (недостатня продуктивність торгового модуля). Стан дошок обох спринтів наведено на рисунках 3.19 та 3.20.

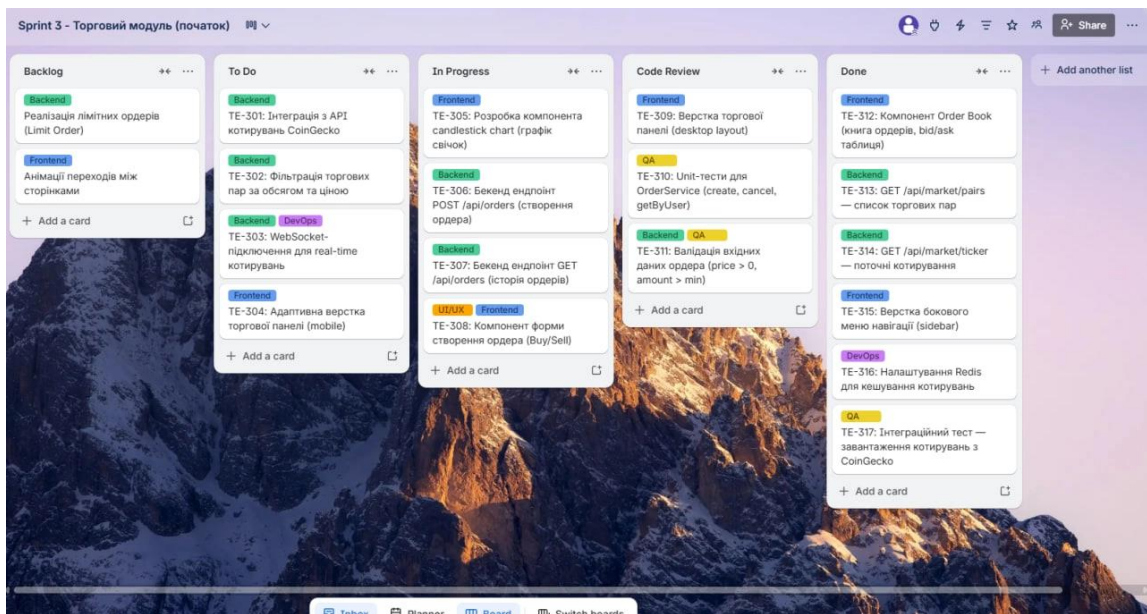


Рис. 3.19. Дошка Спринту 3 у Trello (Торговий модуль: початок)

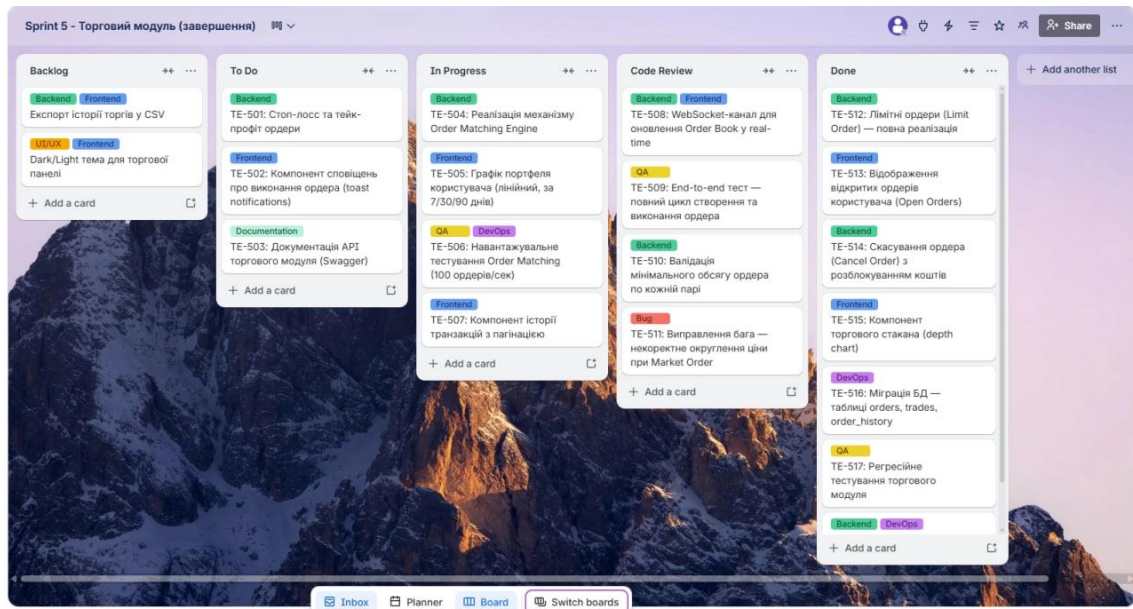


Рис. 3.20. Дошка Спринту 5 у Trello (Торговий модуль: завершення)

Спринт 3: Торговий модуль (початок)

На планування спринту команда відібрала 17 завдань із беклогу продукту. Мета якої створити базову інфраструктуру торгового модуля, а саме програмний інтерфейс для ордерів, інтеграція з CoinGecko для котирувань та побудова торгової панелі на фронтенді.

На момент фіксації стану дошки (рис. 3.16) шість завдань завершено. Два програмних інтерфейси для ринкових даних, компонент книги ордерів, бічна панель навігації, налаштування кешування котирувань та інтеграційний

тест для зовнішнього API. Чотири завдання перебувають у виконанні: графік свічок, інтерфейс форми ордера, створення та перегляд ордерів. На перевірці коду є три завдання: верстка торгової панелі, модульні тести та валідація вхідних параметрів ордера. У беклозі залишилися два завдання, які не вмістились у спринт за оцінкою складності.

На ретроспективі зафіксовано, що інтеграція з CoinGecko пройшла швидше за оцінку завдяки якісній документації постачальника, верстка торгової панелі затрималась через відсутність мокапів мобільної версії. Прийняте рішення, мокапи мобільної версії готуються за тиждень до початку спринту, де вони потрібні.

Спринт 5: Торговий модуль (завершення)

Мета Спринту 5, це реалізація механізму зведення ордерів, навантажувальне тестування та підключення оновлень у реальному часі через протокол WebSocket. Цей спринт є критичним, тому що, після нього замовник вперше бачить повністю працюючий торговий модуль на контрольній точці КТ-3 (22 травня).

На момент фіксації стану дошки (рис. 3.17) шість завдань завершено: лімітні ордери, скасування ордера з розблокуванням коштів, відображення відкритих ордерів, графік глибини ринку, міграція бази даних та регресійне тестування. У виконанні є чотири завдання, механізм зведення ордерів, навантажувальне тестування (цільовий показник 100 ордерів на секунду), графік портфеля та таблиця транзакцій з посторінковою навігацією. На перевірці коду виявлено помилку TE-511 некоректне округлення ціни при ринковому ордері через використання числа з рухомою комою замість типу з фіксованою точністю. Виявлення помилки на етапі перевірки коду, а не у продакшні, підтверджує ефективність обов'язкової взаємної перевірки.

На ретроспективі зафіксовано позитивний результат, механізм зведення ордерів витримав навантажувальне тестування з показником 120 ордерів на секунду, що перевищує цільові 100 і знижує оцінку ймовірності ризику R4 з рівня 3 до 2. За результатами помилки TE-511 до настанов з оформлення коду

додано правило, використання числа з рухомою комою у фінансових обчисленнях заборонено.

Порівняння двох спринтів відображає закономірну еволюцію. Спринт 3 зосереджений на побудові базової інфраструктури, Спринт 5 на складних алгоритмічних компонентах. Кількість завершених завдань залишається стабільною (по шість у кожному спринті), що свідчить про коректну оцінку продуктивності команди під час планування. Таким чином, Trello виконує не лише функцію візуалізації, а й є джерелом даних для аналізу відхилень і коригування ризиків у реєстрі.

РОЗДІЛ 4. ПРАКТИКА ТА ОЦІНКА РЕЗУЛЬТАТІВ

4.1. Спринт 1: планування, реалізація, проблеми та результати ревію

Перший спринт став початковим етапом практичного застосування гібридної моделі управління проєктом Trade&Earn. На цьому етапі відбувся перехід від каскадного планування, реалізованого у попередніх розділах, до ітеративної розробки в межах Scrum. Беклог спринту був сформований на основі WBS та результатів архітектурного проєктування, що забезпечило узгодженість задач із загальною структурою проєкту.

Основною метою Спринту 1 було створення базової технічної основи системи. У межах ітерації виконано налаштування середовища розробки, організацію репозиторію, розгортання CI/CD-пайплайну, а також створення каркасу бекенд- і фронтенд-застосунків. Окрім цього, реалізовано первинну інтеграцію із зовнішніми API для отримання ринкових даних, що є критично важливим для функціонування торгового модуля.

За результатами спринту команда отримала перший працюючий інкремент продукту у вигляді базового прототипу. Це дозволило перевірити обрані технічні рішення, а також забезпечило можливість ранньої демонстрації результатів замовнику. Наявність інкременту вже після першої ітерації підтверджує ефективність використання Scrum-компоненти в межах гібридної моделі[28].

Спринт 1 мав адаптаційний характер і був спрямований не лише на реалізацію функціоналу, але й на налаштування процесу взаємодії в команді. У ході виконання відбувалося уточнення оцінок задач і координація між різними ролями, що є типовим для початкових ітерацій. Виявлені відхилення у виконанні окремих задач не розглядаються як недоліки управління, а як закономірний результат переходу до практичної реалізації[30].

Під час Sprint Review замовнику було продемонстровано функціональний прототип системи. Отриманий зворотний зв'язок стосувався переважно зручності інтерфейсу та необхідності його подальшого

вдосконалення. Це дозволило скоригувати пріоритети беклогу та підготувати основу для наступного спринту.

Таким чином, перший спринт виконав свою ключову функцію як адаптаційний етап впровадження ітеративного підходу та забезпечив основу для подальшої стабілізації процесу розробки.

4.2. Спринт 2: розвиток функціоналу та уточнення процесу

Другий спринт було організовано з урахуванням результатів попередньої ітерації, що дозволило перейти від етапу адаптації процесу до його стабілізації. Основною метою стало підвищення узгодженості роботи команди та точності планування без зміни загальної логіки управління.

Під час планування Спринту 2 було здійснено більш детальну декомпозицію задач, що забезпечило точнішу оцінку їх трудомісткості[29]. Особливу увагу приділено узгодженню залежностей між бекендом і фронтом, що дозволило зменшити ризик блокувань у процесі розробки. Крім того, тестувальники були залучені до виконання задач з початку спринту, що сприяло більш ранньому виявленню дефектів і підвищенню якості результату.

У межах другого спринту реалізовано функціонал особистого кабінету користувача, розширено інтеграцію із зовнішніми API та вдосконалено інтерфейс відповідно до зауважень замовника. Також було уточнено критерії готовності задач, що забезпечило єдине розуміння результату серед усіх учасників команди.

Отримані результати підтверджують, що команда успішно перейшла від етапу первинної адаптації до більш стабільного виконання задач. Співвідношення між запланованим і фактично виконаним обсягом робіт покращилося, а процес розробки став більш передбачуваним. Це свідчить про ефективність використання ітеративного підходу в межах гібридної моделі та здатність процесу до самовдосконалення.

4.3. Порівняння планових і фактичних показників проєкту

Оцінка результатів реалізації проєкту передбачає зіставлення планових показників, визначених на етапі календарно-сітьового та ресурсного планування, з фактичними результатами, отриманими під час виконання спринтів[33]. Аналіз проводиться за трьома ключовими параметрами: термінами, бюджетом і обсягом робіт.

Згідно з розрахунками у MS Project, планова тривалість проєкту становить 38 тижнів із датою завершення 22 вересня 2026 року . У межах перших двох спринтів істотних відхилень від загального календарного графіка не зафіксовано. Невеликі відхилення на рівні окремих задач мають локальний характер і пояснюються уточненням технічних рішень та координацією між компонентами системи.

Фінансовий аналіз показує, що після деталізації плану загальна вартість проєкту становить 3 248 250 грн, що перевищує початкову оцінку 2,5 млн грн приблизно на 30% . Це перевищення є результатом більш точної оцінки, отриманої під час планування, і не свідчить про неефективність виконання. Фактичні витрати на етапі реалізації відповідають скоригованому бюджету.

Щодо обсягу робіт, у першому спринті спостерігалися незначні відхилення між запланованими і фактично виконаними задачами, що пояснюється адаптаційним характером початкової ітерації. У другому спринті ці відхилення були мінімізовані за рахунок уточнення оцінок і покращення координації між учасниками проєкту. Така динаміка відповідає очікуваній логіці розвитку Scrum-процесу.

Загалом аналіз показує, що всі виявлені відхилення мають контрольований характер і не впливають на досягнення стратегічних параметрів проєкту. Це підтверджує узгодженість між каскадним плануванням і ітеративною реалізацією.

На етапі планування для проєкту Trade&Earn було сформовано реєстр із 10 ризиків із визначеними стратегіями реагування (підрозділ 3.3). Виконання двох перших спринтів дало змогу перевірити, наскільки фактичний перебіг

подій відповідає запланованим оцінкам. Результати переоцінки реєстру наведено у таблиці 4.1.

Таблиця 4.1.

Результати моніторингу реєстру ризиків за підсумками Спринтів 1–2

Ризик	Планова оцінка (Risk Score)	Фактичний статус	Чи спрацював тригер	Вжиті заходи / Результат
R1. Зміна регуляторних вимог KYC/AML	15	Активний, під моніторингом	Ні	Бізнес-аналітик здійснив моніторинг сайту НКЦПФР; нових нормативних актів не опубліковано.
R2. Звільнення ключового бекенд-розробника	12	Активний, під контролем	Ні	Практика парного програмування застосована для Trading Service та модуля безпеки; архітектурні рішення зафіксовано в Confluence. Кадрових змін у команді не відбулось
R3. Збій або зміна API зовнішніх бірж	12	Знижений	Частково	У Спринті 3 зафіксовано короточасну недоступність CoinGecko API (15 хв). Автоматичне перемикання на резервного провайдера спрацювало коректно; торгова панель продовжила роботу завдяки кешу котирувань. Адаптер оновлено
R4. Недостатня продуктивність торгового модуля	12	Закритий	Так	Навантажувальне тестування у Спринті 5 показало 120 ордерів/сек при цільових 100. Risk Score знижено з 12 до 8. Ризик переведено до категорії моніторингу
R5. Розширення обсягу робіт (scope creep)	12	Активний, контрольований	Частково	На Sprint Review Спринту 2 замовник ініціював запит на додавання мобільного додатку. Запит оформлено як Change Request; після оцінки впливу замовник відклав функцію до наступної версії. Механізм реагування підтвердив ефективність

Порівняння планових оцінок із фактичними результатами свідчить про ефективність розробленого реєстру ризиків як інструменту управління. Чотири з п'яти критичних ризиків залишаються під контролем без настання негативних наслідків, ризик R4 закрито достроково завдяки превентивній стратегії навантажувального тестування з Спринту 5. Єдиний випадок часткового спрацювання тригера (R3) підтвердив коректність закладених архітектурних рішень. Резервний механізм перемикання провайдерів котирувань спрацював автоматично і не призвів до простою торгового модуля. Таким чином, відхилення між плановими оцінками ризиків та фактичним перебігом подій мають контрольований характер, що підтверджує достатність розроблених стратегій реагування для даного проєкту.

4.4. Оцінка адекватності гібридної моделі управління

Практичне застосування гібридної моделі (Waterfall + Scrum) у проєкті Trade&Earn дозволяє оцінити її ефективність з точки зору досягнення поставлених цілей[31]. Як було визначено у попередніх розділах, ключовим завданням є поєднання передбачуваності для замовника з гнучкістю для команди розробки.

Каскадна складова моделі забезпечила чітку структуру проєкту, наявність календарного плану, контрольних точок і можливість довгострокового прогнозування. Це дозволило задовольнити вимоги замовника щодо прозорості процесу та контролю ключових етапів. Використання MS Project дало змогу визначити критичний шлях і врахувати ресурсні обмеження ще на етапі планування .

Scrum-компонента забезпечила гнучкість на етапі реалізації. Ітеративна розробка, регулярні демонстрації результатів і ретроспективи дозволили оперативно реагувати на зворотний зв'язок і уточнювати функціональні вимоги. Це особливо важливо для проєктів у динамічному середовищі, де вимоги можуть змінюватися протягом реалізації .

У процесі застосування моделі було виявлено певні обмеження, пов'язані з необхідністю узгодження детального довгострокового плану з

гнучкістю коротких ітерацій. Це вимагає від менеджера проєкту постійного балансування між дотриманням календарного графіка та адаптацією до змін. Водночас зазначені особливості не знижують ефективність моделі, а відображають її специфіку.

Отже, гібридна модель управління продемонструвала свою адекватність для проєкту Trade&Earn, забезпечивши баланс між контролем і гнучкістю та дозволивши ефективно координувати роботу команди в умовах змінного середовища[32].

4.5. Рекомендації щодо масштабування гібридної моделі

Результати дослідження дозволяють сформулювати рекомендації щодо застосування гібридної моделі управління в інших ІТ-проєктах. Такий підхід є доцільним у випадках, коли проєкт має жорсткі обмеження за термінами і бюджетом, але водночас характеризується змінністю функціональних вимог.

Гібридна модель особливо ефективна для проєктів із регуляторними вимогами, де частина робіт повинна виконуватися у чітко визначеній послідовності, а інша частина потребує гнучкості. До таких проєктів належать фінтех-рішення, медичні інформаційні системи та державні цифрові сервіси[35].

Важливою умовою успішного застосування є наявність чітко визначеної архітектури на ранніх етапах. Саме вона забезпечує стабільність системи та дозволяє безпечно впроваджувати зміни у межах спринтів. Крім того, необхідною є достатня зрілість команди, яка повинна володіти як інструментами каскадного планування, так і принципами Agile.

Практичні результати проєкту Trade&Earn підтверджують, що гібридна модель може бути ефективно застосована у складних ІТ-проєктах із поєднанням регуляторних вимог і змінного середовища. За умови дотримання визначених критеріїв вона забезпечує баланс між передбачуваністю та гнучкістю і може розглядатися як універсальний підхід для подібних задач[36].

ВИСНОВОК

У даній дипломній роботі було досліджено проблему управління проєктом розробки інтернет-порталу для торгівлі криптовалютою на прикладі проєкту Trade&Earn компанії ТОВ «ТрейдСофт». Актуальність обраної теми обумовлена стрімким розвитком криптовалютного ринку в Україні, появою відповідної законодавчої бази та зростаючою потребою у якісних інструментах управління подібними проєктами.

У ході дослідження було встановлено, що головною управлінською проблемою при реалізації таких проєктів є суперечність між очікуваннями замовника щодо передбачуваності та жорсткого плану, і потребою команди розробки у гнучкості та самоорганізації. Ця суперечність унеможливило застосування будь-якої однієї класичної методології у чистому вигляді, що і зумовило вибір гібридного підходу Waterfall + Scrum як основи моделі управління.

Для обґрунтування цього вибору було проведено порівняльний аналіз чотирьох методологій за допомогою методу зваженої оцінки. Гібридна модель отримала найвищий бал і перевершила найближчого конкурента більш ніж на 30%, що математично підтвердило правильність прийнятого рішення. При цьому до стандартного переліку критеріїв оцінювання було додано параметр врахування методологічного конфлікту між стейкхолдерами, що дозволило підвищити точність аналізу стосовно реальних умов проєкту.

На етапі планування було розроблено повний комплекс управлінських інструментів: ієрархічну структуру робіт, матрицю відповідальності RACI, беклог продукту та систему контрольних точок. Побудова детальної календарно-сітьової моделі у MS Project дозволила уточнити тривалість проєкту до 38 тижнів та виявити реальну вартість його реалізації - 3 248 250 грн, що на 30% перевищило первинну оцінку. Цей результат наочно продемонстрував, наскільки важливим є детальне ресурсне планування із застосуванням спеціалізованого програмного забезпечення, адже саме воно дозволяє уникнути бюджетних сюрпризів вже під час виконання робіт.

Паралельно з управлінськими інструментами було розроблено ключові технічні артефакти фази проєктування: архітектуру порталу на основі мікросервісного підходу, ER-діаграму бази даних, мокапи інтерфейсу та блок-схеми бізнес-процесів. Було доведено, що ці артефакти виконують не лише технічну, але й управлінську функцію - вони фіксують обсяг робіт до початку розробки, чітко розмежовують відповідальність між спеціалістами та слугують основою для декомпозиції завдань у спринтах. Крім того, встановлено прямий зв'язок між архітектурними рішеннями та управлінням ризиками: модульна мікросервісна архітектура безпосередньо знизилася ймовірність двох найбільш критичних технічних ризиків проєкту.

Практичне застосування гібридної моделі підтвердило її ефективність на стадії виконання. Аналіз двох спринтів показав стабільну продуктивність команди, успішне проходження навантажувального тестування торгового модуля з результатом 120 ордерів на секунду при цільових 100, а також своєчасне виявлення критичної помилки на етапі перевірки коду. Порівняння планових і фактичних показників свідчить про те, що всі відхилення носили локальний і контрольований характер та не вплинули на досягнення ключових параметрів проєкту.

Практичне застосування моделі в інших проєктах потребує врахування кількох умов. Команда має мати досвід роботи з обома підходами. У командах понад 15 осіб доцільно розглянути SAFe або LeSS замість класичного Scrum. У проєктах з іншими регуляторними вимогами склад каскадної фази змінюється, але механізм переходу до ітеративної розробки залишається незмінним.

Дослідження охоплювало команду з 12 осіб і два спринти, що залишає простір для подальшої роботи. Актуальними напрямками є перевірка моделі на проєктах з командою 30 і більше осіб, порівняльний аналіз із фреймворками SAFe та LeSS, а також дослідження застосовності моделі у медичних інформаційних системах і державних цифрових сервісах, де регуляторне середовище суттєво відрізняється від фінтех.

Таким чином, мети дослідження досягнуто, усі поставлені завдання вирішено. Розроблена гібридна модель управління забезпечує баланс між передбачуваністю для замовника та гнучкістю для команди розробки, і може бути рекомендована до застосування в інших ІТ-проектах з аналогічними характеристиками - зокрема у сферах фінтех, медичних інформаційних систем та державних цифрових сервісів, де регуляторні вимоги поєднуються зі змінністю функціонального середовища.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. CoinGecko. Top Crypto Exchanges Ranked by Trust Score. URL: <https://www.coingecko.com/en/exchanges> (дата звернення: 15.03.2026).
2. Statista. Cryptocurrency - Statistics & Facts. URL: <https://www.statista.com/topics/2514/blockchain/> (дата звернення: 15.03.2026).
3. Закон України «Про віртуальні активи» від 17.02.2022 № 2074-IX. URL: <https://zakon.rada.gov.ua/laws/show/2074-20> (дата звернення: 10.03.2026).
4. PMI. What is Project Management? Project Management Institute. URL: <https://www.pmi.org/about/what-is-project-management> (дата звернення: 12.03.2026).
5. Atlassian. Waterfall Methodology: A Complete Guide. URL: <https://www.atlassian.com/agile/project-management/waterfall-methodology> (дата звернення: 12.03.2026).
6. The Scrum Guide. The Definitive Guide to Scrum: The Rules of the Game. 2020. URL: <https://scrumguides.org/scrum-guide.html> (дата звернення: 12.03.2026).
7. Atlassian. What is Kanban? URL: <https://www.atlassian.com/agile/kanban> (дата звернення: 12.03.2026).
8. Atlassian. What is a Work Breakdown Structure (WBS)? URL: <https://www.atlassian.com/work-management/project-management/wbs> (дата звернення: 14.03.2026).
9. НКЦПФР. Реєстр постачальників послуг, пов'язаних з віртуальними активами. URL: <https://www.nssmc.gov.ua/> (дата звернення: 10.03.2026).
10. Asana. Hybrid Project Management: How to Get Started. URL: <https://asana.com/resources/hybrid-project-management> (дата звернення: 14.03.2026).
11. PMI. A Guide to the Project Management Body of Knowledge (PMBOK Guide). 7th Edition. 2021. URL: <https://www.pmi.org/pmbok-guide-standards> (дата звернення: 18.03.2026).
12. PMI. Pulse of the Profession 2024. URL: <https://www.pmi.org/learning/thought-leadership/pulse> (дата звернення: 18.03.2026).

13. Asana. Weighted Scoring Model: How to Prioritize Projects. URL: <https://asana.com/resources/weighted-scoring> (дата звернення: 18.03.2026).
14. Boehm B., Turner R. Balancing Agility and Discipline: A Guide for the Perplexed. Addison-Wesley, 2004. 266 p.
15. Wysocki R. Effective Project Management: Traditional, Agile, Extreme. 7th ed. Wiley, 2014. 704 p.
16. Методичні вказівки до написання кваліфікаційної роботи магістра за спеціальністю 122 «Комп'ютерні науки», освітньо-наукова програма «Управління проектами» (для денної форми навчання). Морозов В.В., Тімінський О.Г., Коломієць А.С. 2024.
17. Atlassian. How to Estimate Project Costs. URL: <https://www.atlassian.com/work-management/project-management/project-cost-estimation> (дата звернення: 20.03.2026).
18. DOU.ua. Зарплати українських ІТ-спеціалістів - січень 2025. URL: <https://dou.ua/lenta/articles/salary-report-january-2025/> (дата звернення: 20.03.2026).
19. Investopedia. Net Present Value (NPV). URL: <https://www.investopedia.com/terms/n/npv.asp> (дата звернення: 20.03.2026).
20. ISO 31000:2018. Risk management - Guidelines. International Organization for Standardization, 2018. URL: <https://www.iso.org/standard/65694.html> (дата звернення: 20.03.2026).
21. Regulation (EU) 2023/1114 of the European Parliament and of the Council of 31 May 2023 on markets in crypto-assets (MiCA). Official Journal of the European Union, L 150, 9.6.2023, p. 40–205. URL: <https://eur-lex.europa.eu/eli/reg/2023/1114/oj> (дата звернення: 20.03.2026).
22. Sigala N. S. "Microservices Architecture in Cloud Computing: A Software Engineering Perspective on Design, Deployment, and Management." International Journal of Research and Innovation in Social Science (IJRISS), Vol. 9, No. 2, 2025. URL: <https://rsisinternational.org/journals/ijriss/articles/microservices-architecture->

[in-cloud-computing-a-software-engineering-perspective-on-design-deployment-and-management/](#) (дата звернення: 20.03.2026).

23. Taibi D. et al. "Patterns in Design of Microservices Architecture: IT Practitioners' Perspectives." *WIPIEC Journal*, 2022. URL: <https://wipiec.digitalheritage.me/index.php/wipiecjournal/article/view/101> (дата звернення: 20.03.2026).

24. Cerny T. et al. "Modeling microservice architectures." *Journal of Systems and Software (JSS)*, Vol. 211, 2024. <https://www.sciencedirect.com/science/article/abs/pii/S0164121224000840> (дата звернення: 20.03.2026).

25. Nurfaizah, Astuti H. "Application of Database Normalization in Increasing Data Storage Efficiency." *ITALIC Journal*, Vol. 4, No. 2, 2025. URL: <https://journal.pandawan.id/italic/article/view/799> (дата звернення: 20.03.2026).

26. Nielsen Norman Group. F-Shaped Pattern of Reading on the Web: Misunderstood, But Still Relevant. 2017. URL: <https://www.nngroup.com/articles/f-shaped-pattern-reading-web-content/> (дата звернення: 20.03.2026).

27. Atlassian. Scrum Board: Getting Started with Agile. *Atlassian Agile Coach*, 2026. URL: <https://www.atlassian.com/agile/project-management/scrum-board> (дата звернення: 20.03.2026).

28. Ramos R. et al. "Exploring Retrospective Meeting Practices and the Use of Data in Agile Software Development." *arXiv*, 2025. URL: <https://arxiv.org/html/2502.03570v1> (дата звернення: 20.03.2026).

29. Lenarduzzi V. et al. "Effort Estimation in Agile Projects Using Adaptive AI." *Theses Journal*, 2024. URL: <https://thesesjournal.com/index.php/1/article/download/392/354> (дата звернення: 20.03.2026).

30. Ciric D. et al. "Goals and challenges in hybrid software development approaches." *Journal of Software: Evolution and Process*, Wiley, 2022.

- <https://onlinelibrary.wiley.com/doi/full/10.1002/smr.2382> (дата звернення: 20.03.2026).
31. Khoza L., Marnewick C. "Waterfall and Agile information system project success rates – A South African perspective." *South African Computer Journal (SACJ)*, Vol. 32, No. 1, 2020. URL: <https://sacj.cs.uct.ac.za/index.php/sacj/article/view/683> (дата звернення: 20.03.2026)
32. Adelakun O., Garcia R., Tabaka T. "Hybrid Project Management: Agile with Discipline." *Journal of Computer Information Systems*, 2021. Vol. 61, No. 4. <https://aisel.aisnet.org/confirm2017/14/> (дата звернення: 20.03.2026)
33. PMI. Earned Value Management. A Guide to the Project Management Body of Knowledge (PMBOK Guide). 6th ed. Project Management Institute, 2017. P. 261–270. <https://philarchive.org/archive/ADEMAB> (дата звернення: 20.03.2026)
34. Mohamad N. et al. "Factors Affecting Scope Creep in Project Management: Identify the Key Factors Contributing to Scope Creep." *International Journal of Engineering Technology and Research Management (IJETRM)*, 2022. URL: <https://www.academia.edu/128281492> (дата звернення: 20.03.2026)
35. VersionOne ([Digital.ai](https://digital.ai)). 17th State of Agile Report. 2021. URL: <https://digital.ai/resource-center/analyst-reports/state-of-agile-report> (дата звернення: 20.03.2026)
36. Reiff J., Schlegel D. "Hybrid Project Management: Scoping Review." *Heliyon* (Elsevier), 2025 <https://www.sciencedirect.com/science/article/pii/S2666721525000079> (дата звернення: 20.03.2026)
37. Публікації. Гончар М.Ю., Хлевний А.О. Development of an Internet Portal for Trading Education with Blockchain Technology Integration and Tokenized Reward System // Збірник тез доповідей XI Міжнародної конференції «Information Technology and Implementation (IT&I-2025)». – Київ: Видавничий дім «Каравела», 2025. – С. 232. – ISBN 978-960-801-928-7.