

Харківський національний університет міського господарства  
імені О.М. Бекетова  
ННІ енергетичної, інформаційної та транспортної інфраструктури  
Кафедра управління проєктами в міському господарстві і будівництві

## **Пояснювальна записка до кваліфікаційної роботи бакалавра**

на тему: «Управління проєктом розробки інформаційної системи підтримки  
B2B-комунікацій»

Виконав: студент 4 курсу, групи УП(кн)2022-1  
Спеціальності 122 – Комп'ютерні науки  
Освітня програма «Комп'ютерні науки.  
Управління проєктами»

Камаєв Станіслав Владиславович

(ПІБ повністю, підпис)

Керівник: Косенко Н.В.

(прізвище та ініціали, підпис)

Рецензент: Косенко В.В.

(прізвище та ініціали, підпис)

Харків – 2026 року

Харківський національний університет міського господарства  
імені О.М Бекетова

Навчально-науковий інститут енергетичної, інформаційної та  
транспортної інфраструктури

Кафедра управління проєктами в міському господарстві і будівництві

Освітньо-кваліфікаційний рівень бакалавр

Спеціальність 122 – Комп'ютерні науки

Освітня програма «Комп'ютерні науки. Управління проєктами»



ЗАТВЕРДЖУЮ  
Завідувач кафедри УПМГБ  
д.т.н., проф. Чумаченко І.В.  
“ 15 ” 06 202\_ року

### З А В Д А Н Н Я НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА студенту

Камаєву Станіславу Владиславовичу  
(прізвище, ім'я, по батькові)

1. Тема роботи «Управління проєктом розробки інформаційної системи підтримки B2B-комунікацій»

Керівник Косенко Наталія Вікторівна, к.т.н, доцент

( прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 440-03 від 22.05.2026р.

2. Строк подання студентом роботи 10.06.2026 р.

3. Вихідні дані до роботи: матеріали переддипломної практики, інформація періодичних видань, книг, монографій, матеріали з сіткового планування та управління, інтернет-ресурси.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Теоретичні основи управління проєктами розробки інформаційних систем підтримки b2b-комунікацій; 2. Постановка задачі та проєктування системи «Ksv Code Studio»; 3. Управління життєвим циклом розробки csm-екосистеми.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) Презентація в форматі MS Power Point. Обов'язкові елементи – діаграма Ганта, сітьовий графік.

6. Консультанти розділів проекту

| Розділ        | Прізвище, ініціали та посада консультанта         | Підпис, дата                                                                        |                                                                                     |
|---------------|---------------------------------------------------|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
|               |                                                   | завдання видав                                                                      | завдання прийняв                                                                    |
| Нормоконтроль | Косенко Н.В., доц.,<br>к.т.н., доц. каф.<br>УПМГБ |  |  |

7. Дата видачі завдання 12.05.2026 р.

### КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів роботи                          | Строк виконання етапів роботи | Примітка |
|-------|----------------------------------------------|-------------------------------|----------|
| 1     | Розробка 1-го розділу кваліфікаційної роботи | 25.05.2025                    | Виконано |
| 2     | Розробка 2-го розділу кваліфікаційної роботи | 01.06.2025                    | Виконано |
| 3     | Розробка 3-го розділу кваліфікаційної роботи | 05.06.2025                    | Виконано |
| 4     | Нормоконтроль                                | 10.06.2025                    | Виконано |
| 5     | Попередній захист кваліфікаційної роботи     | 12.06.2025                    | Виконано |
| 6     | Рецензування кваліфікаційної роботи          | 15.06-19.06.2025              | Виконано |
| 7     | Захист на ЕК                                 | 24.06.2025                    |          |

Студент

  
(підпис)

Камаєв С.В.

(прізвище та ініціали)

Керівник роботи

  
(підпис)

Косенко Н.В.

(прізвище та ініціали)

## РЕФЕРАТ

Тема кваліфікаційної роботи бакалавра: «Управління проєктом розробки інформаційної системи підтримки B2B-комунікацій». Робота містить вступ, 3 розділи, висновки, список з 16 використаних джерел інформації. Пояснювальна записка виконана на 93 сторінках, містить 6 рисунків, 34 таблиці і 4 додатки.

Метою кваліфікаційної роботи є розробка проєкту створення інформаційної системи підтримки B2B-комунікацій для фриланс-студії «KSV Code Studio».

Об'єкт дослідження: процеси управління IT-проєктами створення інформаційної системи підтримки B2B-комунікацій.

Предмет дослідження: методи, моделі, інструментальні засоби та практики управління проєктами на прикладі розробки інформаційної системи для «KSV Code Studio».

У першому розділі кваліфікаційної роботи були розглянуті теоретичні основи вибору гнучких методологій, проаналізовано стратегічну необхідність використання сучасних IT-рішень у галузі автоматизації B2B-комунікацій.

У другому розділі було визначено поточний операційний стан студії, спроектовано цільову архітектуру процесів, здійснено збір та пріоритизацію вимог.

В третьому розділі було розроблено проєкт створення інформаційної системи підтримки B2B-комунікацій для фриланс-студії «KSV Code Studio».

УПРАВЛІННЯ ПРОЄКТАМИ, AGILE, CRM-СИСТЕМА, B2B-КОМУНІКАЦІЇ, ЖИТТЄВИЙ ЦИКЛ РОЗРОБКИ, РЕІНЖИНІРИНГ ПРОЦЕСІВ, МАТРИЦЯ ТРАСУВАННЯ, УПРАВЛІННЯ РИЗИКАМИ

## ЗМІСТ

|                                                                                                                               |    |
|-------------------------------------------------------------------------------------------------------------------------------|----|
| ВСТУП .....                                                                                                                   | 8  |
| РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ УПРАВЛІННЯ ПРОЄКТАМИ РОЗРОБКИ ІНФОРМАЦІЙНИХ СИСТЕМ ПІДТРИМКИ В2В-КОМУНІКАЦІЙ .....                 | 11 |
| 1.1 Аналіз підходів та методів управління ІТ-проєктами .....                                                                  | 11 |
| 1.2 Аналіз використання інформаційних технологій у галузі автоматизації В2В.....                                              | 16 |
| 1.2.1 Специфіка та потреби В2В сектору в ІТ-рішеннях .....                                                                    | 16 |
| 1.2.2 Технології оптимізації клієнтського досвіду (Customer Experience)..                                                     | 19 |
| 1.2.3 Автоматизація внутрішніх бізнес-процесів (Business Process Automation).....                                             | 21 |
| Висновки до розділу 1 .....                                                                                                   | 24 |
| РОЗДІЛ 2 ПОСТАНОВКА ЗАДАЧІ ТА ПРОЄКТУВАННЯ СИСТЕМИ «KSV CODE STUDIO» .....                                                    | 26 |
| 2.1 Розгорнута постановка задачі: аналіз вихідної ситуації (модель AS-IS) та проєктування цільового стану (модель TO-BE)..... | 26 |
| 2.1.1 Ідентифікація та характеристика існуючих бізнес-процесів студії ....                                                    | 26 |
| 2.1.2 Моделювання вихідної ситуації (AS-IS) та SWOT-аналіз.....                                                               | 33 |
| 2.1.3 Проєктування цільової моделі (TO-BE) та дорожня карта реінжинірингу .....                                               | 38 |
| 2.2 Аналіз обмежень, збір вимог та їх пріоритизація.....                                                                      | 43 |
| 2.2.1 Процес збору та класифікації вимог .....                                                                                | 44 |
| 2.2.2 Методи пріоритизації беклогу: MoSCoW та Капо .....                                                                      | 46 |
| 2.3 Опис сценаріїв діалогу з системою та матриця трасування .....                                                             | 48 |
| 2.3.1 Моделювання взаємодії акторів із системою (Use Case Diagram).....                                                       | 49 |
| 2.3.2 Формалізація користувацьких сценаріїв (User Stories та BDD) .....                                                       | 50 |
| 2.3.3 Забезпечення якості змісту: Матриця трасування вимог (RTM) .....                                                        | 52 |

|                                                                                            |    |
|--------------------------------------------------------------------------------------------|----|
| Висновки до розділу 2 .....                                                                | 52 |
| РОЗДІЛ 3 УПРАВЛІННЯ ПРОЄКТОМ РОЗРОБКИ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ B2B-КОМУНІКАЦІЙ..... | 54 |
| 3.1 Визначення основних характеристик проєкту.....                                         | 54 |
| 3.1.1 Статут проєкту та його основні характеристики.....                                   | 54 |
| 3.1.2 Реєстр зацікавлених сторін (стейкхолдерів) .....                                     | 57 |
| 3.1.3 Матриця відповідальності (RACI) та розподіл ролей .....                              | 60 |
| 3.1.4 План управління комунікаціями.....                                                   | 61 |
| 3.1.5. Управління змінами проєкту (PCR) .....                                              | 63 |
| 3.2 Управління роботами та ресурсами проєкту.....                                          | 65 |
| 3.2.1 Ієрархічна структура робіт (WBS) проєкту .....                                       | 66 |
| 3.2.2 Формування команди проєкту та розподіл ролей.....                                    | 68 |
| 3.2.3 Ресурсне планування та матричне завантаження .....                                   | 72 |
| 3.3 Управління розкладом та вартістю проєкту.....                                          | 73 |
| 3.3.1 Визначення тривалості робіт та план за віхами.....                                   | 73 |
| 3.3.2 Оцінка вартості та складання бюджету .....                                           | 74 |
| 3.3.3 Моніторинг та контроль проєкту (Метод освоєного обсягу).....                         | 75 |
| 3.4 Управління ризиками та якістю проєкту .....                                            | 77 |
| 3.4.1 Ідентифікація ризиків та стратегії реагування.....                                   | 77 |
| 3.4.2 Планування якості та причинно-наслідковий аналіз дефектів.....                       | 79 |
| Висновки до розділу 3 .....                                                                | 81 |
| ВИСНОВКИ.....                                                                              | 82 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....                                                            | 84 |
| ДОДАТКИ.....                                                                               | 86 |
| Додаток А. Діаграма Ганта.....                                                             | 86 |
| Додаток Б. Мережева схема .....                                                            | 87 |
| Додаток В. Аркуш ресурсів.....                                                             | 89 |
| Додаток Г. Графічні матеріали .....                                                        | 90 |

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

API (Application Programming Interface) – прикладний програмний інтерфейс.

B2B (Business-to-Business) – бізнес для бізнесу (сегмент ринку).

BDD (Behavior-Driven Development) – розробка на основі поведінки.

BPMN (Business Process Model and Notation) – нотація моделювання бізнес-процесів.

CRM (Customer Relationship Management) – система управління відносинами з клієнтами.

EVM (Earned Value Management) – метод освоєного обсягу.

IDEF0 (Integration Definition for Function Modeling) – методологія функціонального моделювання.

MVP (Minimum Viable Product) – мінімально життєздатний продукт.

RACI (Responsible, Accountable, Consulted, Informed) – матриця розподілу відповідальності.

RBAC (Role-Based Access Control) – управління доступом на основі ролей.

RTM (Requirements Traceability Matrix) – матриця трасування вимог.

WBS (Work Breakdown Structure) – ієрархічна структура робіт (ICP).

## ВСТУП

У сучасному висококонкурентному бізнес-середовищі цифрова трансформація та комплексна автоматизація бізнес-процесів є невіддільними складовими успішного функціонування та масштабування підприємств малого та середнього бізнесу. Перехід від фрагментованого ручного управління до використання єдиних інтегрованих інформаційних екосистем дозволяє суттєво мінімізувати операційні ризики, знизити витрати та підвищити рівень задоволеності клієнтів (Customer Satisfaction). Сьогодні на ринку B2B стрімко зростає попит на розробку корпоративних CRM-систем, індивідуальних веб-інтерфейсів, Telegram-ботів, систем автоматизації та індивідуальних цифрових рішень. Саме на проєктуванні та впровадженні таких технологічних продуктів для малого та середнього бізнесу спеціалізується «KSV Code Studio».

Управління IT-проєктами у сфері індивідуальної розробки вимагає високого рівня зрілості процесів, гнучкості та чіткої формалізації вимог, оскільки потреби замовників є динамічними. Створення системи, яка повинна дозволяти клієнтам подавати заявки через сайт або Telegram-бот, а менеджерам - обробляти їх у єдиній CRM-панелі, вимагає злагодженої роботи крос-функціональної команди. Впровадження автоматичного формування інвойсів та фінансових звітів для внутрішнього обліку, а також розробка адаптивного особистого кабінету клієнта для моніторингу статусу замовлення в режимі реального часу, формують складну архітектуру продукту. Крім того, реалізація надійної рольової моделі з розмежуванням прав (Супер-адмін, Адмін, Менеджер, Розробник, Бухгалтер) та забезпечення інтеграції із зовнішніми сервісами, такими як Google Sheets API та Telegram API, вимагають застосування стандартизованих управлінських практик.

Успішна реалізація подібних проєктів неможлива без імплементації доведених методологій управління (PMBOK, Agile/Scrum), інструментів контролю якості (Traceability Matrix) та технік пріоритизації беклогу

(MoSCoW, Kano). З огляду на це, дослідження підходів до управління проєктами на прикладі розробки власної CRM-екосистеми для фриланс-студії є надзвичайно актуальним науково-практичним завданням, що відображає реальні виклики сучасної IT-індустрії.

Об'єктом дослідження: процеси управління IT-проєктами створення інформаційної системи підтримки B2B-комунікацій.

Предметом дослідження: методи, моделі, інструментальні засоби та практики управління проєктами на прикладі розробки інформаційної системи для «KSV Code Studio».

Метою кваліфікаційної роботи є розробка проєкту створення інформаційної системи підтримки B2B-комунікацій для фриланс-студії «KSV Code Studio».

Для досягнення поставленої мети у роботі вирішуються такі завдання:

- здійснити збір, аналіз та класифікацію системних вимог (поділ на функціональні та нефункціональні) для структурування потреб стейкхолдерів, визначення технічних критеріїв якості та бізнес-цілей продукту;
- формалізувати вимоги та проєктування користувацького досвіду через розробку User Stories (у форматі «Як роль, я хочу можливість, щоб отримати цінність») та Acceptance Criteria (за моделлю BDD: Given-When-Then);
- візуальне моделювати бізнес-процеси та взаємодію акторів із системою з використанням діаграм варіантів використання (Use Case Diagram) у нотації UML для чіткого розмежування зон відповідальності;
- провести пріоритизацію беклогу вимог за допомогою методів MoSCoW та Kano, з метою виділення критичного функціоналу для MVP (Must have / Basic) та визначення додаткової цінності продукту (Excitement);
- забезпечити управління якістю (Quality Management) та змістом (Scope Management) через побудову матриці трасування вимог, що пов'язує

кожну функціональну вимогу з відповідним сценарієм тестування та тест-кейсами, гарантуючи повне покриття системи контролем;

- розробити процесну моделі проєкту за стандартом PMI PMBOK.

Практичне значення одержаних результатів роботи полягає у тому, що розроблена проєктна документація, спроектована архітектура та інструментарій управління життєвим циклом повністю готові до впровадження в операційну діяльність фриланс-студії «KSV Code Studio». Використання запропонованої цільової моделі (TO-BE), інтеграція багатоканального залучення лідів (Telegram API) та автоматизація фінансового документообігу дозволять підприємству зменшити комунікаційну асиметрію, скоротити операційні витрати та мінімізувати вплив людського фактора під час реалізації B2B-проєктів.

## **РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ УПРАВЛІННЯ ПРОЄКТАМИ РОЗРОБКИ ІНФОРМАЦІЙНИХ СИСТЕМ ПІДТРИМКИ B2B- КОМУНІКАЦІЙ**

### **1.1 Аналіз підходів та методів управління IT-проектами**

У сучасній інженерії програмного забезпечення вибір методології управління життєвим циклом (Software Development Life Cycle, SDLC) є стратегічним рішенням, що визначає здатність команди постачати бізнес-цінність в умовах мінливого середовища. Фундаментальний вододіл в управлінні проектами проходить між традиційними (предиктивними) та гнучкими (адаптивними) методологіями.

Традиційний каскадний підхід (Waterfall) базується на строгій лінійній послідовності фаз: збір вимог, проєктування, розробка, тестування, впровадження та підтримка. Ця модель вимагає вичерпної документації на початковому етапі та фіксації меж проєкту (Scope). Однак для проєктів із високим рівнем невизначеності вимог Waterfall демонструє значну неефективність: будь-які зміни на пізніх стадіях призводять до експоненційного зростання вартості та зриву строків. Натомість сімейство гнучких методологій (Agile), зокрема фреймворки Scrum та Kanban, орієнтовані на ітеративно-інкрементальну доставку продукту [14, с. 25; 16, с. 12]. Agile дозволяє адаптуватися до нових бізнес-вимог у процесі розробки, підтримує постійний зворотний зв'язок із замовником та фокусується на працюючому програмному забезпеченні як головному показнику прогресу.

Враховуючи специфіку «KSV Code Studio» - фриланс-студії, що спеціалізується на розробці індивідуальних цифрових рішень, корпоративних CRM-систем, Telegram-ботів та веб-інтерфейсів для малого та середнього бізнесу - використання предиктивного підходу є нежиттєздатним. Клієнти B2B-сектору часто не мають чіткого бачення кінцевого продукту на стадії ініціації, що зумовлює постійну еволюцію вимог. Саме тому для управління

проєктом розробки внутрішньої екосистеми студії було обрано методологію Agile. Вона забезпечує необхідну гнучкість для швидкого прототипування та імплементації інтеграцій із зовнішніми сервісами (Google Sheets API, Telegram API).

Практична реалізація гнучкого управління в «KSV Code Studio» здійснюється за допомогою інструментарію Atlassian Jira. Для структурування обсягу робіт (Scope) використовуються три базові артефакти: Board (дошка для візуалізації робочого процесу за принципами Kanban/Scrum), Епіс (великі блоки робіт) та Story (користувацькі історії). Як видно з Рисунка 1.1, дошка дозволяє команді застосовувати pull-систему (витягування задач), прозоро відстежуючи рух кожної заявки від статусу "To Do" до "Done", що мінімізує ризики простою ресурсів.

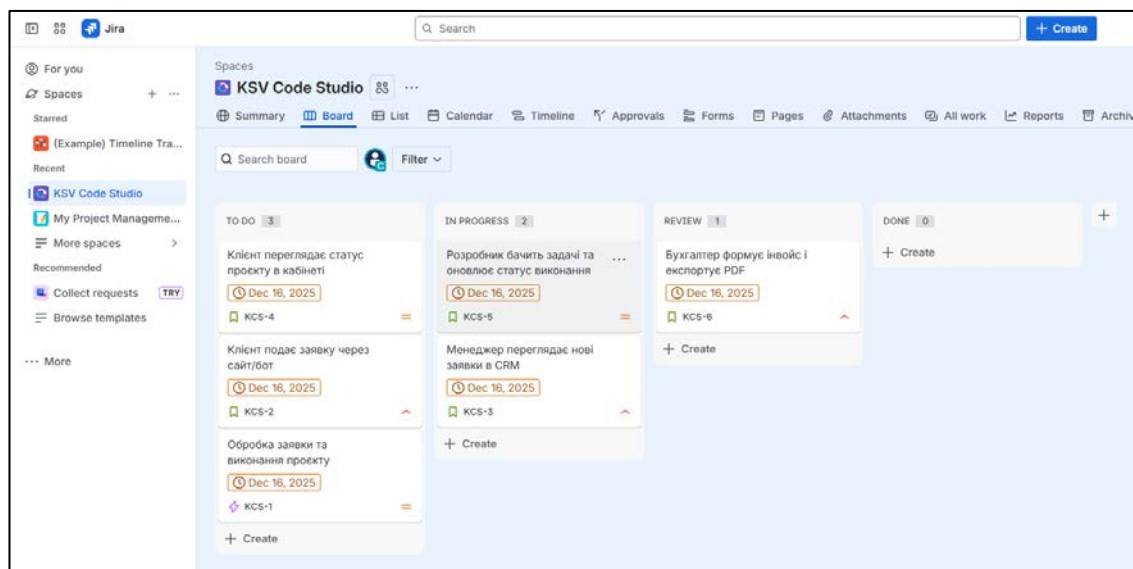


Рисунок 1.1 – Візуалізація робочого процесу (Board) проєкту KSV Code Studio в середовищі Jira

Як приклад, весь життєвий цикл роботи з клієнтом інкапсульовано у глобальний Епіс під назвою «Обробка заявки та виконання проєкту», який об'єднує всі пов'язані завдання в єдиний пул. Рисунок 1.2 демонструє, як Епіс виступає контейнером для всіх операційних процесів. Це дозволяє

проектному менеджеру оцінювати загальний прогрес виконання (через Epic progress) та контролювати стратегічні межі проєкту (Scope Management).

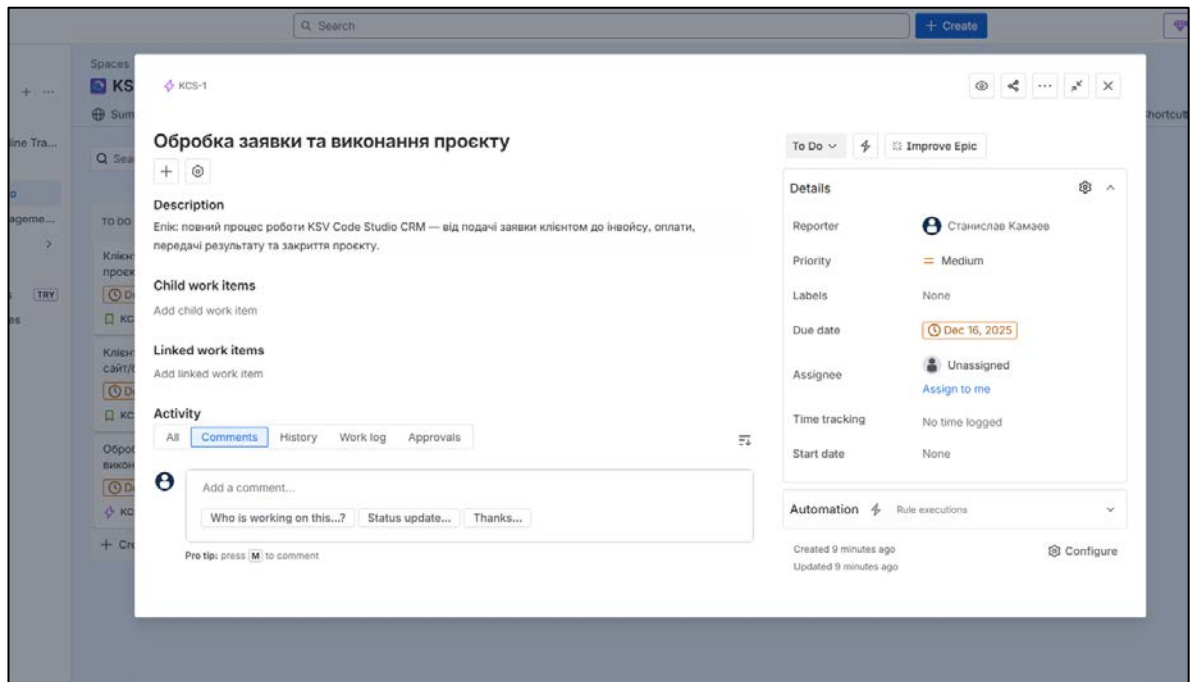


Рисунок 1.2 – Декомпозиція високорівневих вимог через артефакт Epic

Центральним методом формалізації вимог у межах обраного підходу є використання User Stories, які зміщують фокус із технічного опису на бізнес-цінність для конкретного стейкхолдера. У проєкті вони стандартизовані за класичним шаблоном: «Як [роль], я хочу [можливість], щоб [цінність]». Класичним прикладом такої декомпозиції є вимога операційного управління: «Як менеджер, я хочу бачити всі нові заявки в CRM-панелі, щоб швидко призначати відповідальних розробників».

Для того, щоб перевести User Stories з площини бізнес-побажань у площину чітких інженерних метрик якості (Definition of Done), застосовуються критерії прийнятності (Acceptance Criteria), оформлені за моделлю поведінково-орієнтованої розробки (BDD) - Given-When-Then. Це усуває будь-яку двозначність у трактуванні завдань між менеджерами, розробниками та тестувальниками. Наприклад, для перевірки функціоналу обробки заявок визначено наступні критерії: «Given що нова заявка

надходить до системи, When менеджер відкриває CRM-панель, Then заявка відображається у списку з позначкою "Нова"». На Рисунку 1.3 наочно показано застосування BDD-моделі (Given-When-Then) при описі задачі у Jira. Такий формат перетворює абстрактну бізнес-вимогу на конкретний сценарій для тестувальників та розробників, забезпечуючи високу якість (Quality Assurance) вже на етапі постановки задачі.

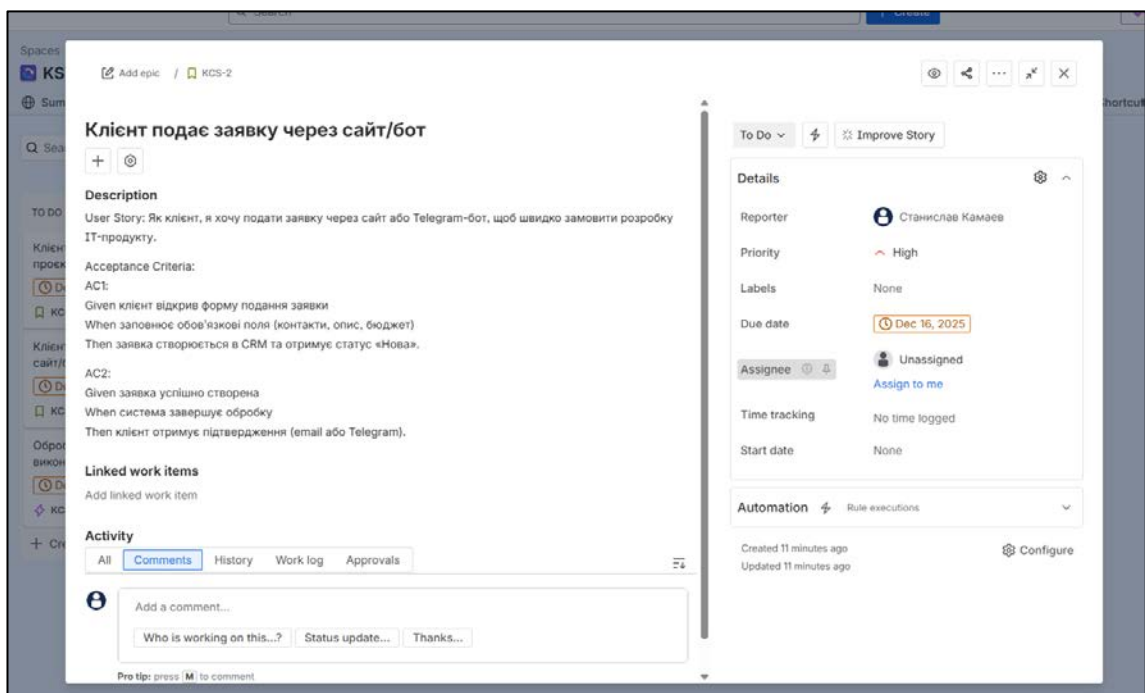


Рисунок 1.3 – Формалізація вимог за допомогою User Story та Acceptance Criteria

Застосування вищеописаних методологій та інструментів дозволило побудувати прозорий і керований життєвий цикл замовлення в системі «KSV Code Studio», що є основним потоком створення цінності (Value Stream). Цей цикл складається з таких регламентованих етапів:

1. Ініціація: Клієнт взаємодіє із системою через веб-форму або Telegram-бот, заповнюючи обов'язкові поля (контакти, опис задачі, бюджет). Після відправки даних система генерує сутність у CRM, яка автоматично отримує статус «Нова», а клієнту надсилається підтвердження.

2. Розподіл ресурсів: Менеджер ідентифікує нову заявку в CRM-панелі, аналізує деталі та використовує функцію «Призначити розробника», переводячи завдання у стан виконання (In Progress).

3. Виконання: Призначений розробник отримує доступ до задачі у персональному модулі «Мої завдання». У процесі розробки він актуалізує статуси (наприклад, «у роботі», «потребує уточнення», «виконано»), які синхронно оновлюються для менеджера та відображаються у кабінеті клієнта.

4. Моніторинг: Завдяки реалізації кабінету клієнта, замовник має змогу в режимі реального часу відстежувати поточний статус виконання (у відсотках або етапах) у розділі «Мій проєкт».

5. Завершення та фінансовий розрахунок: Коли розробка завершена, проєкт переходить у стадію «Оплата». У цей момент у процес інтегрується бухгалтер, який через фінансовий модуль ініціює автоматичну генерацію інвойсу на основі проєктних даних. Після формування документа система дозволяє здійснити його експорт (завантаження у форматі PDF), що містить коректні реквізити студії та клієнта, фіналізуючи тим самим операційний життєвий цикл.

Такий структурований життєвий цикл, підкріплений Agile-практиками та суворим управлінням доступом (де адміністратор гарантує, що кожен актор бачить лише релевантні йому модулі), мінімізує управлінські ризики, забезпечує високу швидкість обробки лідів та формує професійний клієнтський досвід.

## **1.2 Аналіз використання інформаційних технологій у галузі автоматизації B2B**

Після вибору та обґрунтування гнучкої методології управління (Agile), наступним логічним кроком ініціації IT-проєкту є аналіз технологічного ландшафту, у якому буде функціонувати розроблюваний продукт. Для фриланс-студій, таких як «KSV Code Studio», що спеціалізуються на індивідуальній розробці, критично важливо розуміти архітектурні тренди та потреби цільового ринку. У даному підрозділі досліджено специфіку застосування інформаційних технологій в інфраструктурі B2B-комунікацій. Зокрема, увагу зосереджено на трьох стратегічних векторах управління: визначенні базових потреб корпоративного сектору в CRM-екосистемах, аналізі інструментів оптимізації клієнтського досвіду (Customer Experience) та розгляді сучасних інтеграційних сценаріїв (API) для автоматизації внутрішніх бізнес-процесів.

### **1.2.1 Специфіка та потреби B2B сектору в IT-рішеннях**

У сучасному технологічному ландшафті цифрова трансформація малого та середнього бізнесу (SMB) є не просто конкурентною перевагою, а фундаментальною умовою виживання на ринку B2B. Фриланс-студія «KSV Code Studio», що функціонує в місті Харків, фокусується саме на цьому сегменті, пропонуючи розробку корпоративних CRM-систем, веб-інтерфейсів, Telegram-ботів, систем автоматизації бізнес-процесів та індивідуальних цифрових рішень.

Специфіка B2B-сектору (Business-to-Business) суттєво відрізняється від масового ринку (B2C). Вона характеризується складнішими процесами прийняття рішень, тривалими циклами угод, високою вартістю залучення ліда (CAC - Customer Acquisition Cost) та необхідністю глибокої персоналізації послуг. Відповідно, потреби таких компаній в інформаційних технологіях еволюціонують від простих статичних веб-сайтів до комплексних інтегрованих платформ, здатних керувати всім життєвим

циклом взаємодії з клієнтом - від першого контакту до фінансового розрахунку та післяпродажної підтримки.

Аналіз сучасного ринку корпоративних CRM-систем та засобів індивідуальної автоматизації демонструє чітку стратегічну тенденцію: бізнес масово відмовляється від використання розрізнених інструментів на користь єдиних цифрових екосистем. Традиційний підхід, за якого компанія використовує окремий месенджер для комунікації із замовником, електронні таблиці для обліку статусів проєктів та ручне формування фінансових документів у текстових редакторах, неминуче призводить до втрати даних, дублювання інформації та операційної неефективності.

Проєкт власної екосистеми управління замовленнями «KSV Code Studio» є еталонним прикладом такого архітектурного переходу. Замість фрагментованих процесів, система консолідує всі етапи в єдиному цифровому середовищі: прийом заявок через багатоканальні джерела (веб-форма, Telegram-бот), їх маршрутизацію до CRM-панелі менеджера, контроль виконання завдань розробниками та автоматизоване формування інвойсів бухгалтерією. Більше того, для безшовної взаємодії з уже існуючими звичками користувачів реалізована глибока інтеграція із зовнішніми сервісами - Google Sheets та Telegram API, що автоматизує внутрішні робочі процеси без створення штучних бар'єрів. Такий екосистемний підхід гарантує цілісність даних (Data Integrity) та створює єдине джерело істини (Single Source of Truth) для всіх стейкхолдерів проєкту.

У контексті аналізу бізнес-цінності розробленого рішення, ключовий стратегічний ефект від впровадження такої екосистеми полягає у радикальному зниженні комунікаційних та управлінських ризиків. Як показує практика управління IT-проєктами у сфері розробки індивідуальних програмних рішень, нечіткі вимоги, розмиті зони відповідальності або відсутність прозорості часто призводять до конфліктів та неконтрольованого розширення меж проєкту (Scope Creep). Систематизація вимог на функціональні та нефункціональні, а також їх суворий контроль, допомагає

структурувати процес розробки та уникнути непорозумінь із замовниками щодо кінцевого функціоналу продукту.

Для превентивного вирішення проблеми "інформаційного вакууму", з яким часто стикаються B2B-клієнти під час тривалої розробки, в екосистемі KSV Code Studio спроектовано спеціалізований особистий кабінет клієнта. Коли клієнт авторизується в системі та відкриває сторінку «Мій проєкт», він у режимі реального часу бачить актуальний статус виконання робіт (у відсотках або завершених етапах). Це не лише формує високий рівень довіри та задоволеності (Customer Satisfaction), але й розвантажує менеджерів від постійної рутинної комунікації щодо статусу проєкту.

У контексті аналізу бізнес-цінності рішення, важливим аспектом цінності продукту (Value Proposition) є мінімізація часу виконання операцій (Lead Time) за рахунок чіткого структурування функціоналу та підвищення швидкості обробки заявок. Інтеграція інструментів у межах єдиної CRM-панелі дозволяє персоналу миттєво реагувати на нові запити. Згідно з розробленими користувацькими історіями (User Stories), коли клієнт замовляє розробку IT-продукту, вказавши контактні дані, опис задачі та бюджет через форму чи бот, заявка миттєво потрапляє в CRM-систему й автоматично отримує статус «Нова». Це повністю усуває людський фактор при перенесенні даних.

Менеджер, у свою чергу, отримує консолідований список заявок і має змогу одразу ознайомитися з деталями та в один клік («Призначити розробника») передати завдання в роботу. Паралельно, операційний цикл пришвидшується і на етапі фінансового розрахунку. Коли проєкт переходить у стадію "Оплата", бухгалтер не витрачає час на ручний збір реквізитів - система автоматично генерує інвойс на основі наявних у базі даних проєкту, після чого цей документ експортується у PDF для відправки клієнту.

Отже, з управлінської точки зору, впровадження екосистеми управління замовленнями KSV Code Studio не просто забезпечує цифрове збереження інформації, а виступає активним драйвером бізнес-процесів.

Вона усуває "вузькі місця" (bottlenecks), структурує операційну діяльність команди та перетворює фриланс-студію на системний B2B-бізнес із масштабованою архітектурою управління.

### 1.2.2 Технології оптимізації клієнтського досвіду (Customer Experience)

Управління клієнтським досвідом (Customer Experience, CX) у сфері B2B-розробки ІТ-продуктів вийшло за межі простого надання якісного сервісу і стало критичним фактором стратегічного управління проєктами. Відповідно до стандартів PMBOK, це тісно пов'язано з галузями знань Управління комунікаціями (Project Communications Management) та Управління зацікавленими сторонами (Project Stakeholder Management). У проєкті індивідуальної екосистеми для фриланс-студії «KSV Code Studio» оптимізація CX вирішує фундаментальну проблему ІТ-аутсорсингу - усуває комунікаційні бар'єри та інформаційну асиметрію між замовником і командою розробки.

Під час проєктування системи було враховано, що базовим етапом життєвого циклу замовлення є ініціація контакту (Lead Generation), яка в умовах малого та середнього бізнесу вимагає мінімізації «тертя» (friction) на етапі подання заявки. Для вирішення цієї задачі в межах екосистеми «KSV Code Studio» було розроблено омніканальну архітектуру залучення клієнтів. Вона дозволяє одночасно інтегрувати дані з традиційних веб-форм та месенджерів за допомогою Telegram API.

З управлінської точки зору, цей підхід формалізовано в User Story 1: «Як клієнт, я хочу мати можливість подати заявку через сайт або Telegram-бот, щоб швидко замовити розробку потрібного мені ІТ-продукту». Критерії прийнятності (Acceptance Criteria) чітко регламентують, що під час взаємодії із системою клієнт повинен надати ключові артефакти для ініціації проєкту: контакти, опис задачі та бюджет. Щойно клієнт здійснює цю дію (через будь-який із каналів), система миттєво обробляє запит, створює сутність у CRM зі статусом «Нова» та автоматично надсилає клієнту підтвердження (email або повідомлення у Telegram).

Для забезпечення прозорості надання послуг та ліквідації ефекту «чорної скриньки», який часто виникає в управлінні IT-проектами після підписання контракту, у межах «KSV Code Studio» було спроектовано інтерактивний модуль «Кабінет клієнта». Його впровадження дозволяє знизити тривожність стейкхолдерів, мінімізувати мікроменеджмент з боку замовника та зменшити комунікаційне навантаження на проектного менеджера. На етапі проєктування інтерфейсу користувача даний модуль був деталізований у середовищі Figma (макет Project details).

Даний функціонал описується через User Story 3: «Як клієнт, я хочу бачити статус свого проєкту, щоб розуміти прогрес виконання». Управлінська цінність цього рішення розкривається через його Acceptance Criteria. Згідно з AC2, як тільки менеджер або розробник оновлює статус задачі у внутрішній CRM (наприклад, переводить з етапу "проєктування" у "розробку"), ця інформація синхронно та в реальному часі оновлюється в кабінеті клієнта. Клієнт бачить ключові етапи та відсоток готовності продукту без необхідності телефонувати менеджеру чи писати листи.

Аналізуючи цей інструмент через призму методів управління змістом, можна дійти висновку, що за методом MoSCoW особистий кабінет клієнта віднесено до категорії Should have (бажано реалізувати). Однак за моделлю Капо ця функція є яскравим представником категорії Performance (лінійні характеристики). Це означає, що чим детальніше та швидше оновлюється інформація в кабінеті, тим вищим є рівень задоволеності B2B-клієнта. Забезпечення якості цього модуля контролюється вимогою FR-05, яка через матрицю трасування пов'язана з тест-кейсами TC-05.1 (успішна авторизація в кабінеті) та TC-05.2 (коректне відображення статусу).

Інтеграція багатоканального залучення разом із прозорим моніторингом прогресу створює безшовний (seamless) користувацький досвід. Клієнт B2B-сектору отримує відчуття повного контролю над своїми інвестиціями в IT-продукт. Для самої ж студії впровадження кабінету діє як ефективний інструмент Risk Mitigation (зниження ризиків) у сфері

комунікацій, вивільняючи дорогоцінний час команди для фокусування на безпосередній розробці коду, а не на нескінченних статус-мітингах із замовниками.

### 1.2.3 Автоматизація внутрішніх бізнес-процесів (Business Process Automation)

Управління внутрішніми бізнес-процесами є фундаментальною складовою операційної ефективності будь-якої ІТ-організації. Згідно зі стандартами PMBOK, автоматизація бізнес-процесів безпосередньо впливає на процеси управління розкладом [13, с. 175], управління вартістю (Cost Management) та управління якістю (Quality Management), оскільки вона мінімізує ручну працю, усуває так звані «вузькі місця» (bottlenecks) та знижує ризик людської помилки (human error). У проєкті «KSV Code Studio» автоматизація реалізована через глибоку інтеграцію зовнішніх сервісів, цифровізацію документообігу та використання стандартизованих нотацій для моделювання логіки системи.

З теоретичної точки зору, автоматизація бізнес-процесів базується на процесному підході, де кожен процес описується як набір дій щодо перетворення входів (інформації, результатів) у виходи. Для створення комплексних B2B-рішень ключову роль відіграє така функціональна область PMBOK, як управління інтеграцією проєкту. Концепція управління інтеграцією вимагає використання комбінації різних технологій. Організація такого середовища для управління передбачає впровадження спеціалізованих інструментів, таких як Jira, Trello або використання діаграми Ганта. Крім того, процесна автоматизація тісно пов'язана із Загальним управлінням якістю (Total Quality Management, TQM), головними принципами якого є орієнтація на споживача, системний підхід та безперервне покращення інженерних процесів.

При розробці архітектури «KSV Code Studio» значну увагу було приділено інтеграційним сценаріям, зокрема використанню Google Sheets API для експорту проєктних та фінансових даних. В умовах B2B-сектору

здатність системи гнучко взаємодіяти із зовнішнім програмним забезпеченням є критично важливою, оскільки це дозволяє користувачам не відмовлятися від звичних інструментів аналітики. Згідно з проведеним аналізом вимог за методом Кано, автоматичний експорт звітів та глибока інтеграція з Google Sheets API та Telegram API були визначені як функціональні переваги, що суттєво підвищують лояльність кінцевих користувачів (Excitement).

Зокрема, Google Sheets API використовується для автоматичного експорту фінансових і проєктних даних. В управлінській практиці це означає, що менеджмент студії отримує можливість будувати складні аналітичні дашборди або проводити когортний аналіз клієнтів на базі вивантажених даних, не перевантажуючи при цьому саму CRM-систему зайвим функціоналом обчислень. Надійність цієї інтеграції жорстко контролюється через матрицю трасування: вимога FR-08 покривається сценарієм тестування TS-08 та перевіряється через тест-кейс TC-08.1 («Успішний експорт даних у Google Sheets»). Такий архітектурний підхід дозволяє KSV Code Studio автоматизувати рутинні задачі зі збору метрик, звільняючи час команди для створення безпосередньої бізнес-цінності.

Проєктне рішення щодо автоматизації генерації інвойсів та динамічного формування технічних завдань спрямоване на усунення операційних втрат (Waste у термінології Lean/Agile). Як показує аналіз діяльності фриланс-студій, ручне ведення проєктної та фінансової документації є одним із ключових чинників зниження ефективності менеджменту. Впровадження автоматизованого документообігу в «KSV Code Studio» дозволяє перенаправити часовий ресурс команди на виконання безпосередніх завдань проєкту.

По-перше, стандартизація збору вимог (Scope Management) досягається через функціонал динамічного створення документів. Згідно з вимогами, Адміністратор системи має можливість створювати та редагувати шаблони документів і технічних завдань (ТЗ). Цей процес підтверджується тест-

кейсами TC-06.1 («Створення нового шаблону документа») та TC-06.2 («Редагування існуючого шаблону»). Завдяки цьому менеджер не пише кожне ТЗ з нуля, а генерує його на основі затвердженого шаблону, що мінімізує юридичні та технічні ризики при узгодженні меж проєкту із замовником.

По-друге, автоматизовано цикл фінансового розрахунку (Order-to-Cash). Відповідно до User Story 5: «Як бухгалтер, я хочу, щоб система автоматично формувала інвойси, щоб прискорити фінансовий облік та виставлення рахунків клієнтам». Логіка цього процесу, описана через Acceptance Criteria (AC1 та AC2), передбачає, що як тільки проєкт переходить у стадію «Оплата», система самостійно підтягує дані замовлення та генерує інвойс. Бухгалтеру залишається лише перевірити його та натиснути кнопку «Експорт», після чого система генерує PDF-файл із коректними реквізитами студії та клієнта. Візуальне представлення цього фінансового інструменту (екран «Invoice») було спроектовано та затверджено у Figma під час розробки прототипу.

Ця інновація забезпечує миттєве виставлення рахунків, зменшує дебіторську заборгованість та гарантує, що жодна деталь у реквізитах не буде втрачена через неуважність.

З метою формалізації вимог до системи та візуалізації маршрутів проходження заявок у проєкті «KSV Code Studio», було виконано моделювання бізнес-процесів за допомогою нотацій BPMN та UML Activity Diagram. Такий підхід забезпечив точне розуміння того, як інформація розподіляється між учасниками процесу (акторами), що мінімізувало ризики помилок при подальшій інтеграції модулів та налаштуванні API.

Activity Diagram (Діаграма діяльностей UML) використовується для моделювання алгоритмічної логіки окремих варіантів використання на рівні системи. Вона детально описує паралельні процеси. Наприклад, коли розробник змінює статус завдання (у роботі / виконано), діаграма діяльностей показує розгалуження (Fork), де система одночасно оновлює

базу даних, надсилає сповіщення менеджеру та оновлює статус в особистому кабінеті клієнта. Це дозволяє архітекторам побачити технічні залежності між модулями.

BPMN-діаграма розв'язує завдання вищого, управлінського рівня. Вона фокусується на бізнес-процесі в цілому (End-to-End), використовуючи пули (Pools) та доріжки (Swimlanes) для розмежування відповідальності. BPMN наочно демонструє, як заявка, що надійшла через Telegram API (стартова подія), проходить через доріжку Менеджера (обробка, призначення розробника), перетинає доріжку Розробника (виконання, зміна статусу) і завершується в доріжці Бухгалтера генерацією PDF-інвойсу (кінцева подія).

Використання цих нотацій відіграло вирішальну роль у формалізації вимог та виявленні логічних розривів ще до початку етапу програмування. Вони слугують універсальною мовою спілкування між стейкхолдерами: клієнт бачить цінність бізнес-процесу (BPMN), а команда розробки чітко розуміє алгоритм взаємодії компонентів системи (Activity Diagram). Загалом, така ґрунтовна автоматизація та її попереднє графічне проєктування доводять, що розробка KSV Code Studio ведеться з дотриманням найвищих стандартів сучасного IT-менеджменту.

## **Висновки до розділу 1**

У першому розділі було проведено теоретичний аналіз підходів та методів управління IT-проєктами, а також досліджено специфіку використання інформаційних технологій у галузі автоматизації B2B-сектору. На основі проведеного дослідження зроблено наступні висновки:

- доведено, що для розробки індивідуальних інформаційних систем з високим рівнем невизначеності вимог найефективнішою є гнучка методологія (Agile), яка дозволяє ітеративно постачати цінність замовнику. Організація середовища управління вимагає використання сучасних інструментів, зокрема екосистеми Jira (через артефакти Board, Epic, Story);

- аналіз ринку B2B-комунікацій засвідчив стратегічну необхідність переходу від фрагментованих рішень до єдиних інтегрованих цифрових екосистем. Такий підхід дозволяє гарантувати цілісність даних та створити єдине джерело істини (Single Source of Truth) для всіх стейкхолдерів проєкту;
- визначено, що оптимізація клієнтського досвіду (Customer Experience) досягається шляхом імплементації багатоканальних архітектур залучення лідів (зокрема через Telegram API) та створення інтерактивних кабінетів клієнта. Це ліквідує ефект «чорної скриньки», знижує інформаційну асиметрію та мінімізує комунікаційні ризики;
- обґрунтовано, що автоматизація внутрішніх бізнес-процесів повинна спиратися на процесний підхід та ефективне управління інтеграцією проєкту. Впровадження API-інтеграцій та цифровізація документообігу (автоматична генерація технічних завдань та PDF-інвойсів) суттєво усувають операційні втрати, мінімізують вплив людського фактора та підвищують загальну рентабельність фриланс-студії.

## **РОЗДІЛ 2 ПОСТАНОВКА ЗАДАЧІ ТА ПРОЄКТУВАННЯ СИСТЕМИ «KSV CODE STUDIO»**

### **2.1 Розгорнута постановка задачі: аналіз вихідної ситуації (модель AS-IS) та проєктування цільового стану (модель TO-BE)**

Будь-яка розробка комплексної інформаційної системи починається з глибокого аналізу предметної області та поточного операційного стану підприємства. Відповідно до стандартів системної інженерії, перш ніж переходити до написання програмного коду, необхідно формалізувати існуючу проблематику. У даному підрозділі проведено комплексне дослідження операційної діяльності студії «KSV Code Studio». Дослідження структуровано у три етапи: ідентифікація поточних бізнес-процесів, їх функціональне моделювання за принципом AS-IS (як є) із застосуванням інструменту SWOT-аналізу, та розробка цільової моделі TO-BE (як має бути) з формуванням дорожньої карти реінжинірингу.

#### **2.1.1 Ідентифікація та характеристика існуючих бізнес-процесів студії**

Згідно з принципами процесного управління, діяльність будь-якої ІТ-компанії розглядається як мережа пов'язаних процесів, що перетворюють входи (вимоги клієнта) на виходи (готовий програмний продукт). Для проєктування ефективної CRM-екосистеми насамперед необхідно здійснити паспортизацію цих процесів. У Таблиці 2.1 наведено ідентифікацію шести базових бізнес-процесів «KSV Code Studio» та визначено їхніх власників. Для забезпечення можливості подальшого моніторингу, кожному процесу було призначено ключові критерії ефективності (KPI) та цільові значення, які відображено у Таблиці 2.2.

Таблиця 2.1 – Ідентифікація бізнес процесів «KSV Code Studio»

| Назва процесу                     | Опис                                                                           | Власник                               | Проект/компанія |
|-----------------------------------|--------------------------------------------------------------------------------|---------------------------------------|-----------------|
| Управління замовленнями клієнтів  | Прийом заявок від клієнтів, узгодження вимог, формування технічного завдання   | Відділ продажів та акаунт-менеджменту | KSV Code Studio |
| Розробка програмного забезпечення | Проектування архітектури, написання коду, тестування та впровадження продукту  | Відділ розробки (Frontend, Backend)   | KSV Code Studio |
| Управління кадрами                | Найм розробників, адаптація нових співробітників, розподіл завдань             | HR-відділ                             | KSV Code Studio |
| Технічна підтримка клієнтів       | Обслуговування клієнтів після запуску проєкту, вирішення технічних проблем     | Відділ підтримки                      | KSV Code Studio |
| Маркетинг та просування           | Реклама послуг компанії, ведення соцмереж, участь у виставках, SEO-оптимізація | Відділ маркетингу                     | KSV Code Studio |
| Фінансовий облік                  | Контроль витрат, виставлення рахунків, звітність, зарплати співробітникам      | Бухгалтерія                           | KSV Code Studio |

Таблиця 2.2 – Критерії ефективності процесів «KSV Code Studio»

| Назва процесу                     | Класифікація                                                                   | Критерій ефективності                                                               | Цільове значення                                                                                     |
|-----------------------------------|--------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|
| Управління замовленнями клієнтів  | Зовнішній, основний, детальний, здійснення діяльності                          | 1. Час реакції на заявку клієнта<br>2. Відсоток успішно підписаних договорів        | 1. Менше ніж 2 хвилини на першу відповідь<br>2. Не менше 80% заявок переходять у проекти             |
| Розробка програмного забезпечення | Внутрішній, основний, детальний, здійснення діяльності                         | 1. Дотримання термінів спринтів<br>2. Кількість критичних багів після релізу        | 1. Виконання завдань у 90% випадків в межах дедлайну<br>2. Не більше одного критичного багу на реліз |
| Управління кадрами                | Внутрішній, допоміжний, детальний, планування та контроль                      | 1. Середній час закриття вакансії<br>2. Рівень плинності кадрів                     | 1. Менше ніж 30 днів на закриття вакансії<br>2. Не більше 10% плинності на рік                       |
| Технічна підтримка клієнтів       | Зовнішній, основний, детальний, реєстрація інформації та здійснення діяльності | 1. Час реагування на інцидент<br>2. Відсоток вирішених звернень з першого контакту  | 1. Менше ніж 2 години для критичних звернень<br>2. Не менше 70% вирішуються з першої відповіді       |
| Маркетинг та просування           | Зовнішній, допоміжний, верхнього рівня, планування та здійснення               | 1. Кількість лідів, отриманих щомісяця<br>2. Вартість ліда                          | 1. +20% до кількості лідів за квартал<br>2. Зниження вартості ліда на 15% за рік                     |
| Фінансовий облік                  | Внутрішній, допоміжний, детальний, аналіз та контроль                          | 1. Своєчасність підготовки звітів<br>2. Відсутність помилок у фінансових документах | 1. 100% звітів подаються у визначені терміни<br>2. Нуль помилок у звітності                          |

Для того, щоб підготувати базу для подальшого графічного моделювання у нотації IDEF0, кожен із виділених процесів потребує глибокої декомпозиції за методологією ICOM (Inputs, Controls, Outputs, Mechanisms). Ця методологія дозволяє чітко зафіксувати, які дані ініціюють процес, які ресурси його виконують та які регламенти його контролюють. У Таблицях 2.3 – 2.14 наведено деталізований аналіз тривалості, вартості та елементів ICOM для всіх шести ключових операційних циклів компанії – від управління замовленнями клієнтів до фінансового обліку.

Таблиця 2.3 – Процес «Управління замовленнями клієнтів»

| Назва процесу                    | Власник процесу                       | Зацікавлені сторони                                          | Місце в життєвому циклі                                                | Тривалість                                        | Вартість                                                        |
|----------------------------------|---------------------------------------|--------------------------------------------------------------|------------------------------------------------------------------------|---------------------------------------------------|-----------------------------------------------------------------|
| Управління замовленнями клієнтів | Відділ продажів та акаунт-менеджменту | Клієнти, Project Manager, Відділ розробки, Фінансовий відділ | Початковий етап у життєвому циклі проєкту/продукту (збір вимог, старт) | 1–2 тижні (узгодження вимог, підписання договору) | Витрати на роботу менеджерів ( $\approx$ 5–10% бюджету проєкту) |

Таблиця 2.4 – Входи, виходи, елементи управління та механізми процесу «управління замовленнями клієнтів»

| Входи                                                     | Виходи                                        | Елементи управління                             | Механізми                                            |
|-----------------------------------------------------------|-----------------------------------------------|-------------------------------------------------|------------------------------------------------------|
| Заявка клієнта, бриф, початкові вимоги, попередній бюджет | Технічне завдання (ТЗ), договір, план проєкту | CRM-система, чеклісти вимог, правила узгодження | Менеджери з продажу, акаунт-менеджери, документообіг |

Таблиця 2.5 – Процес «Розробка програмного забезпечення»

| Назва процесу                     | Власник процесу                     | Зацікавлені сторони                                    | Місце в життєвому циклі                              | Тривалість  | Вартість                         |
|-----------------------------------|-------------------------------------|--------------------------------------------------------|------------------------------------------------------|-------------|----------------------------------|
| Розробка програмного забезпечення | Відділ розробки (Frontend, Backend) | Клієнти, Project Manager, QA, DevOps, Відділ підтримки | Основний етап у життєвому циклі (створення продукту) | 1-6 місяців | Основна частина бюджету (60–70%) |

Таблиця 2.6 – Входи, виходи, елементи управління та механізми процесу «Розробка програмного забезпечення»

| Входи                                       | Виходи                                                    | Елементи управління                                           | Механізми                                                       |
|---------------------------------------------|-----------------------------------------------------------|---------------------------------------------------------------|-----------------------------------------------------------------|
| Технічне завдання, дизайн, ресурси, команда | Готове ПЗ, документація, вихідний код, тестові результати | Agile/Scrum методологія, планування спринтів, code review, QA | Команда розробників, IDE, GitHub/GitLab, CI/CD, тестові сервери |

Таблиця 2.7 – Процес «Управління кадрами»

| Назва процесу      | Власник процесу | Зацікавлені сторони                                      | Місце в життєвому циклі                    | Тривалість       | Вартість                                      |
|--------------------|-----------------|----------------------------------------------------------|--------------------------------------------|------------------|-----------------------------------------------|
| Управління кадрами | HR-відділ       | Співробітники, Project Manager, Керівництво, Бухгалтерія | Підтримуючий процес (кадрове забезпечення) | Постійний процес | Зарплата HR, витрати на рекрутинг та навчання |

Таблиця 2.8 – Входи, виходи, елементи управління та механізми процесу «Управління кадрами»

| <b>Входи</b>                      | <b>Виходи</b>                                        | <b>Елементи управління</b>                                 | <b>Механізми</b>                                 |
|-----------------------------------|------------------------------------------------------|------------------------------------------------------------|--------------------------------------------------|
| Резюме, вакансії, кадрові потреби | Прийняті співробітники, кадрові документи, онбординг | Політика рекрутингу, стандарти адаптації, оцінка персоналу | HR-менеджери, job-портали, системи обліку кадрів |

Таблиця 2.9 – Процес «Технічна підтримка клієнтів»

| <b>Назва процесу</b>        | <b>Власник процесу</b> | <b>Зацікавлені сторони</b>                | <b>Місце в життєвому циклі</b>                                        | <b>Тривалість</b> | <b>Вартість</b>                               |
|-----------------------------|------------------------|-------------------------------------------|-----------------------------------------------------------------------|-------------------|-----------------------------------------------|
| Технічна підтримка клієнтів | Відділ підтримки       | Клієнти, Project Manager, Відділ розробки | Завершальний етап у життєвому циклі продукту (супровід після запуску) | Постійний процес  | Витрати на зарплату сапарту та інфраструктуру |

Таблиця 2.10 – Входи, виходи, елементи управління та механізми процесу «Технічна підтримка клієнтів»

| <b>Входи</b>                 | <b>Виходи</b>                              | <b>Елементи управління</b>                        | <b>Механізми</b>                                         |
|------------------------------|--------------------------------------------|---------------------------------------------------|----------------------------------------------------------|
| Запити клієнтів, баг-репорти | Вирішені проблеми, оновлення, консультації | SLA (рівень сервісу), база знань, система тикетів | Сапорт-фахівці, система обліку звернень (Jira/Zoho), FAQ |

Таблиця 2.11 – Процес «Маркетинг та просування»

| Назва процесу           | Власник процесу   | Зацікавлені сторони           | Місце в життєвому циклі                                                        | Тривалість | Вартість                                            |
|-------------------------|-------------------|-------------------------------|--------------------------------------------------------------------------------|------------|-----------------------------------------------------|
| Маркетинг та просування | Відділ маркетингу | Клієнти, Керівництво, Продажі | Паралельний процес у життєвому циклі (підтримка бізнесу та залучення клієнтів) | Постійний  | Витрати на рекламу, SEO, маркетинг (10–20% бюджету) |

Таблиця 2.12 – Входи, виходи, елементи управління та механізми процесу «Маркетинг та просування»

| Входи                                                 | Виходи                                            | Елементи управління             | Механізми                                                     |
|-------------------------------------------------------|---------------------------------------------------|---------------------------------|---------------------------------------------------------------|
| Маркетинговий бюджет, стратегія, бренд-позиціонування | Рекламні кампанії, SEO-звіти, заявки від клієнтів | План маркетингу, KPI, аналітика | Маркетологи, SMM-інструменти, SEO-платформи, рекламні сервіси |

Таблиця 2.13 – Процес «Фінансовий облік»

| Назва процесу    | Власник процесу | Зацікавлені сторони                                     | Місце в життєвому циклі                          | Тривалість | Вартість                                             |
|------------------|-----------------|---------------------------------------------------------|--------------------------------------------------|------------|------------------------------------------------------|
| Фінансовий облік | Бухгалтерія     | Керівництво, Інвестори, Податкові органи, Співробітники | Підтримуючий процес (контроль ресурсів компанії) | Постійний  | Зарплата бухгалтера, вартість бухгалтерських програм |

Таблиця 2.14 – Входи, виходи, елементи управління та механізми процесу «Фінансовий облік»

| <b>Входи</b>                                                     | <b>Виходи</b>                                                      | <b>Елементи управління</b>                                        | <b>Механізми</b>                                               |
|------------------------------------------------------------------|--------------------------------------------------------------------|-------------------------------------------------------------------|----------------------------------------------------------------|
| Первинна документація (рахунки, акти, чеки), зарплатні відомості | Фінансові звіти, бюджети, зарплати співробітникам, податкові звіти | Фінансові регламенти, податкове законодавство, внутрішня політика | Бухгалтери, бухгалтерські системи (1С, Zoho Books, QuickBooks) |

Проведена паспортизація та декомпозиція підтверджує наявність складних інформаційних взаємозв'язків між підрозділами. Зібрані дані щодо входів, виходів, механізмів та елементів управління є достатньою аналітичною базою для переходу до наступного етапу – побудови візуальної функціональної моделі підприємства AS-IS та пошуку її структурних недоліків.

### 2.1.2 Моделювання вихідної ситуації (AS-IS) та SWOT-аналіз

Для глибокого розуміння поточного стану управління замовленнями у «KSV Code Studio» необхідно провести функціональне моделювання існуючих бізнес-процесів (модель AS-IS). Як інструмент моделювання обрано методологію IDEF0, яка дозволяє графічно відобразити функції системи, взаємозв'язки між ними, а також інформаційні, керувальні та ресурсні потоки.

Функцією верхнього рівня (контекстна діаграма A-0) є «Управління життєвим циклом розробки ІТ-продукту у KSV Code Studio». На даному етапі входами (Input) системи є:

- заявки клієнтів;
- вимоги та специфікації;
- маркетингові дані;
- первинна бухгалтерська інформація;
- кадрові потреби;

- бюджет проєктів.

Управління (Control) здійснюється через:

- політика компанії;
- договори та юридичні вимоги;
- Agile/Scrum регламенти;
- SLA підтримки;
- податкове законодавство;
- маркетингова стратегія;
- HR-процедури.

Виходами (Output) виступають:

- готове ПЗ;
- ТЗ та проєктна документація;
- звіти (фінансові, маркетингові, технічні);
- найняті співробітники;
- закриті звернення клієнтів;
- рекламні кампанії.

Механізмами (Mechanism) реалізації є:

- команда компанії;
- програмні інструменти: CRM, IDE, Git, CI/CD, Jira;
- сервери та інфраструктура.

Блок А-0 розкладається на 6 основних підпроцесів (модель декомпозиції А0), що відповідають структурі компанії:

А1 – Управління замовленнями клієнтів:

- входи: заявки, бриф;
- виходи: ТЗ, договір;
- управління: CRM-регламенти;
- механізми: Sales/Account менеджери.

А2 – Розробка програмного забезпечення:

- входи: ТЗ, дизайн;

- виходи: готовий продукт, код, документація;
- управління: Agile/Scrum, планування спринтів;
- механізми: команда розробки, IDE, Git, CI/CD.

#### A3 – Управління кадрами:

- входи: кадрові потреби;
- виходи: найняті співробітники, кадрові документи;<sup>4</sup>
- управління: HR-політика;
- механізми: HR-менеджери, job-платформи.

#### A4 – Технічна підтримка клієнтів:

- входи: звернення, баг-репорти;
- виходи: вирішені проблеми, оновлення;
- управління: SLA, база знань;
- механізми: сапорт-команда, Jira/Zoho<sup>4</sup>.

#### A5 – Маркетинг та просування:

- входи: маркетинговий бюджет, стратегія;
- виходи: рекламні кампанії, заявки;
- управління: KPI, маркетинг-план;
- механізми: маркетологи, рекламні інструменти.

#### A6 – Фінансовий облік:

- входи: документи, чеки, зарплатні відомості;
- виходи: фінансові звіти, податкова звітність;
- управління: законодавство, внутрішня політика;
- механізми: бухгалтерія, QuickBooks.

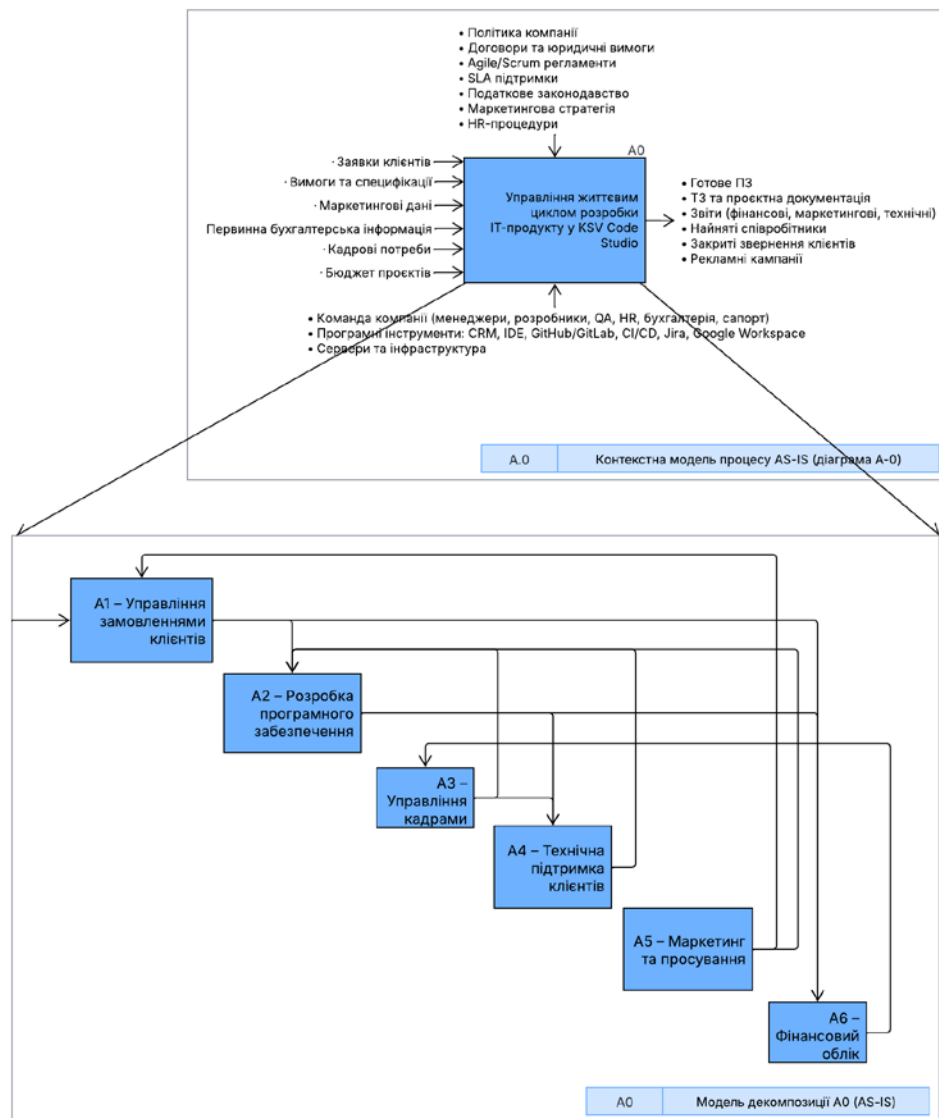


Рисунок 2.1 – Контекстна модель процесу управління життєвим циклом IT-продукту (A-0) та діаграма декомпозиції поточних бізнес-процесів студії (A0)

Побудована модель AS-IS формалізує реальну логіку роботи компанії. Проте, попри наявність визначених процесів, їх поточний операційний стан має ряд критичних обмежень. Для виявлення внутрішніх недоліків та зовнішніх загроз, що уповільнюють масштабування бізнесу, було проведено комплексний SWOT-аналіз. У Таблиці 2.15 зведено результати аналізу всіх шести вищеописаних підпроцесів компанії.

Таблиця 2.15 – Зведений SWOT-аналіз поточних бізнес-процесів KSV Code Studio (модель AS-IS)

| <b>Сильні сторони (Strengths)</b>                                                                                                                                                                                                                                                                                                                                                             | <b>Слабкі сторони (Weaknesses)</b>                                                                                                                                                                                                                                                                                                                                                                       |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Використання гнучких методологій (Agile/Scrum) та наявність базової CRM-системи для обліку заявок.<br>2. Досвідчена крос-функціональна команда розробників та формалізований бухгалтерський облік.<br>3. Наявність системи тикетів для технічної підтримки та формалізованих SLA.<br>4. Визначений маркетинговий бюджет та використання аналітичних інструментів.                          | 1. Значна частка ручної роботи при формуванні технічних завдань (ТЗ) та зведенні фінансових даних.<br>2. Фрагментованість інформації між підрозділами (Sales, Development, Finance).<br>3. Слабке ресурсне планування та нерівномірне навантаження команди розробників.<br>4. Відсутність автоматизованої пріоритизації проєктів та деталізації прибутковості (P&L) по кожному замовленню.               |
| <b>Можливості (Opportunities)</b>                                                                                                                                                                                                                                                                                                                                                             | <b>Загрози (Threats)</b>                                                                                                                                                                                                                                                                                                                                                                                 |
| 1. Впровадження інтегрованої цифрової платформи «єдиного вікна», онлайн-брифів та шаблонів ТЗ.<br>2. Повна автоматизація CI/CD та глибока інтеграція сервіс-деску із загальним беклогом задач.<br>3. Перехід до data-driven маркетингу, заснованого на реальних даних з підтримки та фінансів.<br>4. Впровадження HR-дашбордів для прогнозування кадрових потреб та оптимізації завантаження. | 1. Втрата B2B-клієнтів через повільну реакцію та тривале узгодження вимог (комунікаційні розриви).<br>2. Фінансові ризики та перевищення бюджетів проєктів (Cost Overrun) через помилки в ручних розрахунках.<br>3. Затримки релізів, накопичення технічного боргу та зростання складності проєктів.<br>4. Кадрові ризики: висока плинність та вигорання співробітників через неоптимальне завантаження. |

Висновок за результатами моделювання та SWOT-аналізу: Результати проведеного дослідження підтверджують, що поточна архітектура управління KSV Code Studio вичерпала свій потенціал. Найбільш критичними слабкими сторонами є фрагментованість даних між розрізненими системами (Jira, базові CRM, електронні таблиці) та колосальний обсяг ручної рутини на етапах підготовки ТЗ і генерації

інвойсів. Це генерує загрози зриву термінів та створює комунікаційні розриви із замовниками.

Усунення цих недоліків неможливе шляхом точкових покращень. Виявлені можливості (Opportunities) обґрунтовують об'єктивну необхідність комплексного реінжинірингу бізнес-процесів компанії – переходу від моделі AS-IS до цільового стану (моделі TO-BE) шляхом розробки власної інтегрованої CRM-екосистеми.

### 2.1.3 Проєктування цільової моделі (TO-BE) та дорожня карта реінжинірингу

На основі виявлених під час побудови моделі AS-IS та SWOT-аналізу критичних обмежень (зокрема, фрагментованості інформації, високого рівня ручної роботи та слабкого ресурсного планування), об'єктивною необхідністю є проведення реінжинірингу бізнес-процесів компанії. Метою проєктування цільової моделі (TO-BE) є ліквідація комунікаційних розривів та створення єдиної масштабованої екосистеми управління IT-проєктами.

Для досягнення цільового стану (моделі TO-BE) визначено сім ключових напрямів оптимізації процесів:

- впровадження інтегрованої цифрової платформи «єдиного вікна». Замість розрізнених інструментів передбачається об'єднання CRM, таск-трекера, сервіс-деску, фінансів та HR у єдину систему, що забезпечить прозорий потік даних та підвищить продуктивність команд;
- автоматизація збору вимог та формування ТЗ. Перехід від ручного редагування до автоматизованого ланцюга: онлайн-бриф → шаблон → автоматичне формування чернетки ТЗ та беклогу;
- запровадження портфельного управління (Portfolio Management). Використання матриці пріоритизації проєктів та оцінки прибутковості (P&L) для прийняття обґрунтованих управлінських рішень;

- впровадження CI/CD та автоматичного тестування. Кожен коміт має запускати тести, що прискорить доставку продукту клієнтам та зменшить кількість інцидентів;
- удосконалення ресурсного планування. Впровадження системи обліку трудовитрат (time tracking) та HR-дашбордів для прогнозування ризиків перевантаження співробітників;
- оптимізація служби підтримки (Сервіс-деск). Створення централізованого сервіс-деску з автоматичним перетворенням багів у задачі беклогу;
- впровадження Data-driven маркетингу. Формування стратегії залучення клієнтів на основі фінансової аналітики, швидкості розробки та показників якості (NPS/CSAT).

Трансформація архітектури підприємства від фрагментованого стану до інтегрованої моделі TO-BE є складним проєктним завданням, яке не може бути реалізоване миттєво. Для мінімізації операційних ризиків та забезпечення безперервності бізнесу було розроблено поетапну дорожню карту (Roadmap) переходу.

Дорожня карта реінжинірингу структурована у шість послідовних етапів:

1. Етап 1. Підготовка (1–2 місяці): Аналіз поточних процесів, визначення вимог до інтегрованої платформи, вибір базових інструментів (CRM, таск-трекер, фінанси) та розробка типових шаблонів (бриф, ТЗ, структура backlog, SLA).
2. Етап 2. Процесна стандартизація (1 місяць): Опис та затвердження оновлених процесів (від портфеля до backlog, Scrum-процедури, CI/CD), встановлення KPI для команд щодо продуктивності, SLA, задоволеності клієнтів та фінансів.

3. Етап 3. Впровадження інструментів (2–3 місяці): Технічна інтеграція систем (CRM ↔ таск-трекер, сервіс-деск ↔ backlog), налаштування CI/CD, автоматизація фінансової та HR-аналітики.

4. Етап 4. Навчання персоналу (1 місяць): Проведення тренінгів для відділів Sales, PM, розробки, HR та Support, створення бази знань та проведення тестових спринтів.

5. Етап 5. Пілотний запуск TO-BE (1–2 місяці): Запуск оновленого процесу на 1–2 пілотних проєктах, моніторинг ключових метрик (lead time, cycle time, SLA, P&L) та уточнення регламентів.

6. Етап 6. Масштабування на всю компанію (2–3 місяці): Поширення нового процесу на всі проєкти студії, оптимізація виявлених вузьких місць та остаточне оновлення корпоративних політик.

Згідно з методологією управління проєктами (PMBOK), будь-які організаційні зміни потребують системного моніторингу та контролю. Для об'єктивної оцінки успішності впровадження моделі TO-BE було розроблено план аудиту.

Метою моніторингу є своєчасне виявлення відхилень від плану реінжинірингу та оцінка ефективності змін на основі кількісних метрик. План контролю інтегровано у життєвий цикл проєкту і він включає ключові заходи для кожної стадії (Таблиця 2.16).

Таблиця 2.16 – План контролю та моніторингу переходу до моделі TO-BE

| <b>№</b> | <b>Назва дії з контролю / моніторингу</b>        | <b>Виконавець</b>                 | <b>Час виконання</b>           | <b>Метрики / показники</b>                                       |
|----------|--------------------------------------------------|-----------------------------------|--------------------------------|------------------------------------------------------------------|
| <b>1</b> | Аудит готовності до реінжинірингу (AS-IS аналіз) | Project Manager, Business Analyst | Підготовчий етап               | Наявність опису AS-IS, перелік проблем, затверджена TO-BE модель |
| <b>2</b> | Контроль затвердження процесної моделі TO-BE     | Керівництво, PM                   | Початок проєкту                | Затверджені IDEF0-діаграми, наявність регламентів                |
| <b>3</b> | Моніторинг впровадження CRM та інтеграцій        | IT Lead, PM                       | Етап впровадження інструментів | % інтегрованих систем, кількість автоматизованих процесів        |
| <b>4</b> | Контроль автоматизації збору вимог (A1)          | PM, Sales Lead                    | Етап стандартизації            | Частка проєктів з онлайн-бріфом, час формування ТЗ               |
| <b>5</b> | Аудит впровадження CI/CD (A2)                    | Tech Lead, DevOps                 | Етап розробки                  | % автоматизованих збірок, частота релізів, кількість збоїв       |
| <b>6</b> | Моніторинг ефективності розробки                 | PM, Tech Lead                     | Щомісячно                      | Lead Time, Cycle Time, Velocity команди                          |
| <b>7</b> | Контроль ресурсного планування (A3)              | HR Manager                        | Щомісячно                      | % завантаження персоналу, кількість понаднормових годин          |

Продовження таблиці 2.16

|    |                                               |                   |                     |                                                   |
|----|-----------------------------------------------|-------------------|---------------------|---------------------------------------------------|
| 8  | Аудит роботи сервіс-деску (A4)                | Support Lead      | Після запуску TO-BE | SLA виконання, кількість інцидентів, CSAT         |
| 9  | Моніторинг якості зворотного зв'язку клієнтів | Marketing Manager | Щомісячно           | NPS, CSAT, кількість повторних звернень           |
| 10 | Контроль data-driven маркетингу (A5)          | Marketing Manager | Щоквартально        | Конверсія лідів, вартість залучення клієнта (CAC) |
| 11 | Аудит фінансової аналітики проєктів (A6)      | Finance Manager   | Щомісячно           | P&L проєктів, маржинальність, відхилення бюджету  |
| 12 | Контроль дотримання KPI проєкту реінжинірингу | PM, Керівництво   | Щомісячно           | % досягнення KPI, кількість відхилень             |
| 13 | Моніторинг ризиків реінжинірингу              | PM                | Протягом проєкту    | Кількість ризиків, рівень впливу, статуси         |
| 14 | Аудит навчання персоналу                      | HR Manager        | Етап навчання       | % навчених співробітників, результати тестування  |
| 15 | Оцінка ефективності пілотного запуску         | PM, Stakeholders  | Пілотний етап       | Порівняння AS-IS vs TO-BE метрик                  |
| 16 | Підсумковий аудит TO-BE                       | PM, Керівництво   | Завершення проєкту  | Досягнення цілей, ROI, рівень зрілості процесів   |

Для комплексної оцінки успішності реінжинірингу використовуються такі групи показників:

1. Проєктні метрики: виконання графіка, кількість відхилень від плану, дотримання бюджету.
2. Процесні метрики: середній час від заявки до релізу (Lead Time), кількість ручних операцій, рівень автоматизації.
3. Якісні метрики: кількість дефектів, стабільність релізів, рівень задоволеності клієнтів (CSAT, NPS).
4. Фінансові метрики: прибутковість проєктів (P&L), зниження операційних витрат, ROI від реінжинірингу.

За результатами аналізу очікується, що успішна реалізація дорожньої карти та перехід від моделі AS-IS до TO-BE дозволить компанії «KSV Code Studio» досягти вагомих стратегічних результатів:

- зменшити операційні витрати на 20-30%;
- пришвидшити доставку IT-продукту клієнтам на 30–50%;
- зменшити кількість інцидентів та багів на 40%;
- суттєво підвищити якість сервісу, задоволеність B2B-клієнтів та прибутковість проєктів за рахунок прозорості наскрізної аналітики. Таким чином, розроблена цільова модель формує архітектурний фундамент для створення єдиної цифрової екосистеми управління IT-проєктами підприємства.

## **2.2 Аналіз обмежень, збір вимог та їх пріоритизація**

Управління змістом проєкту (Project Scope Management) згідно зі стандартами PMBOK є одним із найбільш критичних аспектів успішної реалізації програмного забезпечення. Ця галузь знань охоплює процеси, необхідні для гарантування того, що проєкт містить усі необхідні (і лише необхідні) роботи для успішного досягнення цілей. У контексті розробки системи для фриланс-студії «KSV Code Studio» управління змістом відіграє

запобіжну роль проти неконтрольованого розростання меж проєкту (Score Creep) - поширеної проблеми у сфері B2B-розробки.

### 2.2.1 Процес збору та класифікації вимог

Процес управління змістом розпочинається з детального збору та системної класифікації вимог стейкхолдерів. Для того, щоб чітко структурувати майбутній функціонал, задовольнити потреби користувачів та досягти стратегічних бізнес-цілей, усі зібрані вимоги до системи KSV Code Studio були класифіковані на дві фундаментальні категорії: функціональні та нефункціональні.

Функціональні вимоги (Functional Requirements, FR) описують специфічну поведінку системи, тобто те, що система безпосередньо повинна робити. До ключових функціональних вимог системи KSV Code Studio віднесено:

1. Забезпечення клієнтам можливості подавати заявки на розробку через веб-форму або інтегрований Telegram-бот.
2. Отримання менеджерами студії заявок у єдину CRM-панель для їх подальшої обробки та призначення відповідальних виконавців.
3. Впровадження складної рольової моделі доступу, що охоплює такі рівні: Супер-адмін, Адмін, Розробник, Менеджер та Бухгалтер.
4. Наявність особистого кабінету клієнта для моніторингу прогресу розробки замовлення.
5. Автоматичне формування фінансових документів (інвойсів, звітів) та їх експорт, а також генерація технічних завдань на основі шаблонів.

Нефункціональні вимоги (Non-Functional Requirements, NFR) визначають атрибути якості системи, обмеження та стандарти її роботи. У розрізі стратегічного менеджменту саме NFR формують надійність екосистеми. Для даного проєкту були визначені наступні критичні нефункціональні вимоги:

1. Адаптивність та ергономічність: інтерфейс повинен бути простим, інтуїтивним і коректно масштабуватися на мобільних пристроях.

2. Надійність та масштабованість: система повинна гарантувати стабільну роботу без збоїв під час паралельної обробки великої кількості заявок, а також мати архітектурну можливість до масштабування зі зростанням клієнтської бази.

3. Інформаційна безпека: конфіденційні дані користувачів повинні бути захищені алгоритмами аутентифікації та контролю доступу (RBAC), при цьому всі дії користувачів у системі підлягають обов'язковому логуванню для забезпечення можливості подальшого аудиту безпеки.

Ця чітка дихотомія між «що система робить» та «як добре вона це робить» дозволяє інженерній команді сфокусуватися на технічній якості прототипу. Систематизований перелік вимог наведено у Таблиці 2.17.

Таблиця 2.17 – Класифікація системних вимог проєкту KSV Code Studio

| Функціональні вимоги                                   | Нефункціональні вимоги                            |
|--------------------------------------------------------|---------------------------------------------------|
| 1. Подання заявки клієнтом через сайт або Telegram-бот | 1. Інтерфейс має бути зрозумілим та інтуїтивним   |
| 2. Прийом заявок у CRM-панелі менеджера                | 2. Адаптивність інтерфейсу                        |
| 3. Наявність ролей користувачів та управління правами  | 3. Захист даних та безпечне зберігання інформації |
| 4. Збереження історії проєктів та комунікацій          | 4. Надійність роботи під навантаженням            |
| 5. Кабінет клієнта зі статусом виконання проєкту       | 5. Логування всіх дій користувачів                |
| 6. Створення і редагування шаблонів документів адміном | 6. Масштабованість системи                        |
| 7. Автоматичне формування інвойсів та фінзвітів        | -                                                 |
| 8. Інтеграція з Google Sheets та Telegram API          | -                                                 |

Як видно з Таблиці 2.17, формування базового Score проєкту (змісту) здійснюється через балансування між функціональністю системи та її технічною стійкістю. Нефункціональні вимоги виступають у ролі обмежувачів, які гарантують, що розроблений функціонал буде безпечним та масштабованим.

### 2.2.2 Методи пріоритизації беклогу: MoSCoW та Kano

Зібравши пул вимог, проєктна команда стикається з обмеженнями часу та ресурсів, що робить неможливою реалізацію всього обсягу робіт в одній ітерації. Тому для формування MVP (Minimum Viable Product) та планування подальших релізів застосовуються аналітичні методи пріоритизації беклогу - MoSCoW та Kano.

Метод MoSCoW розподіляє вимоги за чотирма векторами критичності: Must have (обов'язкові), Should have (бажані), Could have (можливі) та Won't have (не плануються зараз). Для KSV Code Studio до категорії Must have (без яких запуск системи не має сенсу) віднесено: можливість подання заявки через сайт/бот, перегляд заявок у CRM-панелі, наявність рольової моделі та забезпечення захисту даних. Розподіл усіх вимог за цим методом наведено у Таблиці 2.18. Водночас автоматичне формування інвойсів та особистий кабінет клієнта класифіковані як Should have, оскільки на ранніх етапах ці процеси можна виконувати в напіваавтоматичному режимі. До категорії Could have увійшли функції редагування шаблонів документів та розширені інтеграції з API.

Метод Kano забезпечує інший, клієнтоорієнтований ракурс пріоритизації, аналізуючи вплив функцій на емоційну задоволеність стейкхолдерів.

1. Категорія Basic (базові очікування): подання заявки, ролі та захист даних. Відсутність цього функціоналу викличе крайнє незадоволення клієнтів, проте його наявність не викликає захвату, оскільки сприймається як належне.

Таблиця 2.18 – Пріоритизація беклогу вимог за методом MoSCoW

| Категорія          | Вимога                                       |
|--------------------|----------------------------------------------|
| <b>Must have</b>   | 1. Подання заявки через сайт/Telegram-бот    |
|                    | 2. Перегляд заявок у CRM-панелі              |
|                    | 3. Рольова модель користувачів               |
|                    | 4. Збереження історії проєктів і задач       |
|                    | 10. Захист даних та аутентифікація           |
| <b>Should have</b> | 5. Особистий кабінет клієнта                 |
|                    | 7. Автоматичне формування інвойсів           |
| <b>Could have</b>  | 6. Редагування шаблонів документів           |
|                    | 8. Інтеграція з Google Sheets / Telegram API |
|                    | 9. Адаптивний інтерфейс                      |
| <b>Won't have</b>  | -                                            |

2. Категорія Performance (лінійні характеристики): CRM-панель, історія проєктів, кабінет клієнта та генерація інвойсів. Чим краще та швидше вони працюють, тим вища загальна задоволеність B2B-замовників.

3. Категорія Excitement («вау-ефекти»): розробка шаблонів документів, глибока інтеграція з Google Sheets / Telegram API та висока адаптивність інтерфейсу. Детальний розподіл за моделлю Кано подано у Таблиці 2.19. Користувачі не очікують цих функцій за замовчуванням у стандартних рішеннях, але їх наявність формує значну конкурентну перевагу фриланс-студії на ринку.

Порівняльний аналіз результатів пріоритизації (Таблиці 2.18 та 2.19) демонструє стратегічний управлінський підхід до формування Score проєкту. Варто зазначити, що захист даних (Вимога 10) віднесено до категорії "Must have" за MoSCoW та "Basic" за Кано. Це пояснюється специфікою B2B-сектору: витік інформації (наприклад, фінансових інвойсів чи бази клієнтів) несе критичні юридичні та репутаційні ризики для фриланс-студії. Без базової безпеки система є нежиттєздатною, проте сама по собі безпека не

викликає у клієнта захоплення, оскільки сприймається як обов'язковий стандарт.

Таблиця 2.19 – Класифікація вимог за моделлю задоволеності Kano

| Категорія          | Вимога                                       |
|--------------------|----------------------------------------------|
| <b>Basic</b>       | 1. Подання заявки через сайт/бот             |
|                    | 3. Ролі користувачів                         |
|                    | 10. Захист даних                             |
| <b>Performance</b> | 2. Перегляд заявок у CRM-панелі              |
|                    | 4. Збереження історії проєктів               |
|                    | 5. Кабінет клієнта                           |
|                    | 7. Автоматичне формування інвойсів           |
| <b>Excitement</b>  | 6. Шаблони документів                        |
|                    | 8. Інтеграція з Google Sheets / Telegram API |
|                    | 9. Адаптивний та зручний інтерфейс           |

Натомість інтеграція з Telegram API (Вимога 8) віднесена до категорії "Excitement" за Kano та лише "Could have" за MoSCoW. З технічної точки зору MVP може функціонувати і без неї (використовуючи лише веб-форми), проте можливість безшовної омніканальної комунікації через бот перевищує стандартні очікування клієнтів малого бізнесу. Це створює потужний "вау-ефект", формуючи унікальну конкурентну перевагу (USP) продукту на ринку індивідуальної розробки.

### 2.3 Опис сценаріїв діалогу з системою та матриця трасування

Після збору та пріоритизації системних вимог наступним кроком є формалізація сценаріїв взаємодії користувачів з майбутньою екосистемою «KSV Code Studio». Для цього застосовується комплексний інженерний

підхід: від візуального моделювання меж системи (Use Case Diagram) до деталізації користувацьких історій (User Stories за моделлю BDD) та забезпечення повного покриття коду тестами через матрицю трасування (RTM).

### 2.3.1 Моделювання взаємодії акторів із системою (Use Case Diagram)

Метою побудови діаграми варіантів використання (Use Case Diagram) у нотації UML є візуальний опис ключових сценаріїв роботи продукту, демонстрація взаємодії зовнішніх акторів із системою та визначення меж відповідальності кожного учасника. Оскільки KSV Code Studio впроваджує власну CRM-систему для управління клієнтами, проектами, завданнями та фінансами, у моделі враховано всі основні ролі.

У спроектованій системі ідентифіковано наступних зовнішніх та внутрішніх акторів:

- Клієнт – подає заявку, переглядає статус проекту, отримує звіти.
- Менеджер – приймає заявки, призначає виконавців, оновлює статуси.
- Розробник – виконує задачі та оновлює прогрес розробки.
- Бухгалтер – формує інвойси, здійснює фінансовий контроль.
- Адміністратор системи – керує користувачами та правами доступу.
- Зовнішні сервіси (Telegram API, Google Sheets API) – здійснюють прийом заявок та експорт даних.

Для забезпечення складної логіки роботи CRM-екосистеми на діаграмі використано зв'язки включення (Include) та розширення (Extend). Наприклад, базовий сценарій «Подати заявку» обов'язково включає (Include) перевірку даних та відправку заявки у CRM.

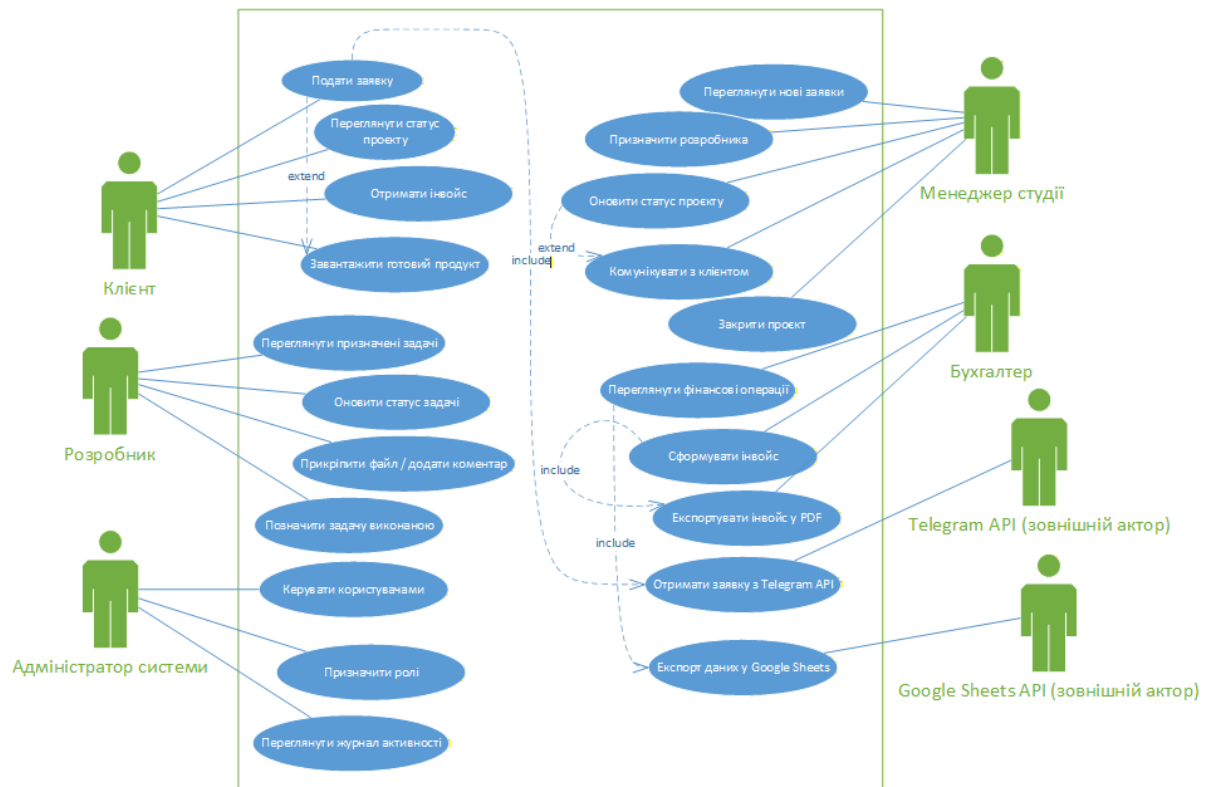


Рисунок 2.2 – Діаграма варіантів використання екосистеми «KSV Code Studio»

Водночас сценарій «Оновити статус проекту» може розширюватися (Extend) функцією надсилення повідомлення клієнту, що є опціональним. Таке структурне розділення дозволяє уникнути хаосу в командній роботі та усуває конфлікти інтересів: розробник не може випадково надіслати клієнту неправильний рахунок, а бухгалтер не може змінити технічний статус завдання.

### 2.3.2 Формалізація користувацьких сценаріїв (User Stories та BDD)

Центральним методом формалізації вимог у межах обраного Agile-підходу є використання User Stories, які зміщують фокус із технічного опису на бізнес-цінність для конкретного стейкхолдера. Вони подані у стандартизованому форматі: «Як [роль], я хочу [можливість], щоб [цінність]».

Для того, щоб перевести User Stories з площини бізнес-поважань у площину чітких інженерних метрик якості (Definition of Done),

застосовуються критерії прийнятності (Acceptance Criteria), оформлені за моделлю поведінково-орієнтованої розробки (BDD) — Given-When-Then. Це усуває будь-яку двозначність у трактуванні завдань між менеджерами, розробниками та тестувальниками.

Основні сценарії діалогу з системою формалізовано наступним чином:

Сценарій 1. Подання клієнтом заявки на розробку. User Story: Як клієнт, я хочу мати можливість подати заявку через сайт або Telegram-бот, щоб швидко замовити розробку потрібного мені IT-продукту. Acceptance Criteria (AC1): Given що клієнт відкрив форму подання заявки, When він заповнює всі обов'язкові поля (контакти, опис задачі, бюджет), Then заявка успішно надсилається у CRM студії.

Сценарій 2. Обробка заявки менеджером. User Story: Як менеджер, я хочу бачити всі нові заявки в CRM-панелі, щоб швидко призначати відповідальних розробників. Acceptance Criteria (AC2): Given що менеджер переглядає деталі заявки, When він натискає кнопку "Призначити розробника", Then система дозволяє вибрати виконавця зі списку доступних розробників.

Сценарій 3. Робота розробника із завданнями. User Story: Як розробник, я хочу бачити список призначених мені задач, щоб ефективно планувати свою роботу. Acceptance Criteria (AC2): Given що розробник відкриває картку задачі, When він змінює статус (у роботі, виконано, потребує уточнення), Then зміни зберігаються та стають доступними менеджеру і клієнту.

Сценарій 4. Автоматичне формування інвойсів. User Story: Як бухгалтер, я хочу, щоб система автоматично формувала інвойси, щоб прискорити фінансовий облік та виставлення рахунків клієнтам. Acceptance Criteria (AC1): Given що проєкт переходить у стадію "Оплата", When бухгалтер відкриває фінансовий модуль, Then система автоматично генерує інвойс на основі даних проєкту.

### 2.3.3 Забезпечення якості змісту: Матриця трасування вимог (RTM)

Для підтвердження того, що заявлені вимоги та описані вище сценарії дійсно реалізовані у продукті, використовується інструмент контролю якості - Матриця трасування вимог (Requirements Traceability Matrix) [3, с. 45]. Вона гарантує повне тестове покриття (Test Coverage) системи, прокладаючи безперервний зв'язок між функціональними бізнес-вимогами (FR), сценаріями тестування (TS) та конкретними тест-кейсами (TC).

Застосування матриці трасування вимог у проєкті KSV Code Studio не лише спрощує процес приймально-здавальних випробувань (UAT), але й відіграє роль стратегічного запобіжника від неконтрольованого розростання меж проєкту (Scope Creep). Якщо розробник пише код, який не покривається жодним із зазначених у матриці тест-кейсів, цей функціонал ідентифікується як зайвий і підлягає вилученню з поточного спринту.

Зведена матриця трасування, розроблена для ключових функціональних вимог системи (від подання заявки до налаштування безпеки RBAC та генерації інвойсів), наведена у Таблиці 2.20.

Таблиця 2.20 – Матриця трасування вимог

| Вимога (FR) | Сценарій (TS) | Тест-кейси (TC)  |
|-------------|---------------|------------------|
| FR-01       | TS-01         | TC-01.1, TC-01.2 |
| FR-02       | TS-02         | TC-02.1, TC-02.2 |
| FR-03       | TS-03         | TC-03.1, TC-03.2 |
| FR-04       | TS-04         | TC-04.1, TC-04.2 |
| FR-05       | TS-05         | TC-05.1, TC-05.2 |
| FR-06       | TS-06         | TC-06.1, TC-06.2 |
| FR-07       | TS-07         | TC-07.1, TC-07.2 |
| FR-08       | TS-08         | TC-08.1, TC-08.2 |
| FR-09       | TS-09         | TC-09.1, TC-09.2 |
| FR-10       | TS-10         | TC-10.1, TC-10.2 |

## Висновки до розділу 2

Другий розділ кваліфікаційної роботи присвячено дослідницько-аналітичному обґрунтуванню та безпосередньому проектуванню архітектури інформаційної системи «KSV Code Studio». За результатами проведених досліджень зроблено наступні висновки:

1. Функціональне моделювання поточної операційної моделі (AS-IS) та проведений SWOT-аналіз довели, що існуюча архітектура процесів студії має низку критичних вразливостей. Основною проблемою є фрагментованість інформації та значна частка ручної праці. Це обґрунтовує необхідність стратегічного реінжинірингу та переходу до моделі TO-BE шляхом розробки власної CRM-екосистеми.

2. Процес управління змістом (Scope Management) дозволив чітко класифікувати вимоги на функціональні та нефункціональні. Використання методів пріоритизації MoSCoW та Капо забезпечило формування оптимального беклогу для створення мінімально життєздатного продукту (MVP). Виявлено, що базові очікування B2B-клієнтів лежать у площині безпеки та прозорості (RBAC, ролі), тоді як додаткову цінність (Excitement) формує інтеграція з Telegram API та Google Sheets.

3. Спроектована діаграма варіантів використання (Use Case Diagram) чітко розмежувала зони відповідальності між зовнішніми акторами (Клієнт, API) та внутрішньою командою.

4. Формалізація користувацьких сценаріїв за моделлю BDD (Given-When-Then) та розробка Матриці трасування вимог (RTM) гарантують 100% тестове покриття майбутнього програмного коду. Цей інструментарій надійно захищає проєкт від неконтрольованого розширення меж (Scope Creep) і створює бездоганну інженерну базу для переходу до етапу безпосередньої розробки та розподілу проєктних ресурсів.

## РОЗДІЛ 3 УПРАВЛІННЯ ПРОЄКТОМ РОЗРОБКИ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ B2B-КОМУНІКАЦІЙ

### 3.1 Визначення основних характеристик проєкту

Ініціація та планування ІТ-проєкту вимагають чіткої формалізації його базових параметрів, обмежень та правил взаємодії між учасниками. Відповідно до міжнародних стандартів управління проєктами, розробка та впровадження внутрішньої CRM-екосистеми для ІТ-студії «KSV Code Studio» базується на розробці комплексного проєктного інструментарію, що включає Статут проєкту, реєстр зацікавлених сторін, матрицю розподілу відповідальності (RACI), план комунікацій та регламент управління змінами.

#### 3.1.1 Статут проєкту та його основні характеристики

Згідно з класичною теорією управління проєктами, Статут проєкту (Project Charter) забезпечує безпосередній зв'язок унікального інженерного завдання зі стратегічними цілями ІТ-організації. У межах даного дослідження Статут відіграє роль фундаментального документа, який фіксує бізнес-потребу, визначає межі проєкту (Project Scope), унікальність продукту, аналіз ресурсів та ключові обмеження (проєктний трикутник: час, вартість, зміст).

Як видно зі Статуту, унікальність проєкту полягає в тому, що архітектура платформи створюється ексклюзивно під специфіку процесів KSV Code Studio, реалізуючи міграцію даних за концепцією "Zero Downtime" (без зупинки поточної комерційної діяльності).

Таблиця 3.1 – Статут проєкту розробки інформаційної екосистеми

| <b>Проект</b>                   |                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Назва проєкту:                  | Розробка внутрішньої екосистеми управління замовленнями «KSV Code Studio»                                                                                                                                                                                                                                                                                                                                          |
| Коротка назва:                  | CRM KSV Code Studio                                                                                                                                                                                                                                                                                                                                                                                                |
| Ініціатори:                     | Спонсор проєкту / Директор студії (Іванов А. П.)                                                                                                                                                                                                                                                                                                                                                                   |
| Дата подання:                   | Квітень 2026 р.                                                                                                                                                                                                                                                                                                                                                                                                    |
| Підготував:                     | Менеджер проєкту Камаєв С. В.                                                                                                                                                                                                                                                                                                                                                                                      |
| Обговорення та доповнення:      | Статут Проєкту може змінюватись у міру розвитку проєкту. Порядок та зміст змін визначається процедурою інтегрованого управління змінами (Change Management) та погоджується безпосередньо зі Спонсором.                                                                                                                                                                                                            |
| <b>Зміст проєкту</b>            |                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Обґрунтування ініціації проєкту | Високі операційні витрати (Waste) та комунікаційні бар'єри при ручному веденні проєктів. Наявність інформаційної асиметрії між клієнтом та командою призводить до збільшення кількості мікроменеджменту. Бізнес вимагає створення єдиного джерела істини (Single Source of Truth) для консолідації даних та прискорення циклу виконання замовлення (Lead Time).                                                    |
| Цілі проєкту                    | <ol style="list-style-type: none"> <li>1. Автоматизувати прийом заявок через омніканальні джерела (Telegram API, веб-форми).</li> <li>2. Підвищити задоволеність B2B-клієнтів (CX) через впровадження кабінету моніторингу статусу.</li> <li>3. Скоротити час на операційний облік завдяки автоматизації виставлення інвойсів.</li> <li>4. Впровадити надійну архітектуру інформаційної безпеки (RBAC).</li> </ol> |
| Результати проєкту              | Розгорнутий, протестований та впроваджений в операційну діяльність програмний комплекс; налаштовані та активні інтеграції з Google Sheets та Telegram API; навчений персонал студії.                                                                                                                                                                                                                               |
| Продукт проєкту                 | Програмна екосистема управління замовленнями (включає веб-інтерфейс, серверну логіку та підключеного Telegram-бота).                                                                                                                                                                                                                                                                                               |
| Структура продукту проєкту      | <ol style="list-style-type: none"> <li>1. Модуль прийому заявок (Клієнтський інтерфейс).</li> <li>2. CRM-панель та Модуль Розробника («Мої задачі»).</li> <li>3. Особистий кабінет клієнта (Project details).</li> <li>4. Фінансовий модуль (генерація Інвойсів).</li> <li>5. Адміністративний модуль управління доступами та шаблонами документів.</li> </ol>                                                     |

Продовження таблиці 3.1

|                                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Учасники та зацікавлені сторони        | Інвестор/Спонсор, Менеджер проєкту, Бізнес-аналітик, Розробники, Бухгалтер, Адміністратор інфраструктури, Зовнішні B2B-замовники послуг.                                                                                                                                                                                                                                                                                                                                                   |
| Основні потреби (очікування) учасників | Спонсор: зростання прибутку за рахунок оптимізації процесів.<br>Клієнт: безпека даних, прозорість етапів розробки.<br>Менеджер: відсутність рутини, зручний розподіл задач (Board/Jira).<br>Розробник: чіткі формалізовані ТЗ (Acceptance Criteria), відсутність мікроменеджменту.<br>Бухгалтер: автоматичний та безпомилковий експорт фінансових документів у PDF.                                                                                                                        |
| <b>Обмеження проєкту</b>               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Обмеження за часом                     | Тривалість проєкту обмежена 6 календарними місяцями. Реліз мінімально життєздатного продукту (MVP) заплановано на 16 грудня поточного року.                                                                                                                                                                                                                                                                                                                                                |
| Обмеження на витрати                   | Бюджет обмежено виділеним фінансуванням на внутрішню розробку (In-house) та лімітами на оплату зовнішніх комерційних API-сервісів і хмарного хостингу.                                                                                                                                                                                                                                                                                                                                     |
| Організаційні та інші вимоги           | Процес міграції зі старих табличних систем у нову CRM має відбутися за концепцією "Zero Downtime", не зупиняючи поточну комерційну діяльність фриланс-студії.                                                                                                                                                                                                                                                                                                                              |
| Обмеження на штат                      | На 100% завантаження виділено лише менеджера проєкту. Решта крос-функціональної команди (розробники, аналітики, адміністратор) працюють за матричною організаційною структурою та залучаються до створення CRM паралельно з виконанням комерційних B2B-замовлень студії.                                                                                                                                                                                                                   |
| Інші обмеження                         | Інтерфейс має бути повністю адаптивним під мобільні пристрої. Технологічний стек зобов'язаний підтримувати безшовну інтеграцію з Telegram API.                                                                                                                                                                                                                                                                                                                                             |
| Критерії оцінки успішності проєкту     | (з урахуванням очікувань основних учасників):<br>1. 100% покриття затверджених бізнес-вимог сценаріями тестування згідно з Матрицею трасування (RTM).<br>2. Автоматична генерація PDF-інвойсу за умови переходу статусу в "Оплата" не перевищує 2-х секунд.<br>3. Система повністю витримує навантаження паралельної обробки заявок без збоїв.<br>4. Аудит безпеки підтверджує абсолютну ізоляцію модулів (користувач без прав не може переглянути чужий проєкт або фінансову статистику). |

### 3.1.2 Реєстр зацікавлених сторін (стейкхолдерів)

Успішна реалізація екосистеми критично залежить від розуміння очікувань усіх зацікавлених сторін. Для виявлення інтересів, потенційних конфліктів та шляхів оптимізації комунікацій було проведено аналіз стейкхолдерів проєкту.

Аналіз зацікавлених сторін проводиться з метою виявлення їхніх інтересів, потенційних конфліктів та шляхів оптимізації комунікацій. Розробка власної екосистеми для фриланс-студії безпосередньо впливає як на внутрішні операційні процеси команди, так і на зовнішній клієнтський досвід. На основі архітектури системи та спроектованої матриці доступу (RBAC) ідентифіковано наступні ключові групи зацікавлених сторін:

#### 1. Внутрішні стейкхолдери (Основна команда та Інвестор):

- Спонсор (Директор студії) – суб'єкт, який забезпечує фінансування та визначає стратегічні цілі оптимізації бізнес-процесів.

- Керівник проєкту (Менеджер) – операційне ядро системи, зацікавлений в автоматизації рутини (прийом заявок, розподіл задач).

- Розробник – технічний виконавець, який потребує ізольованого робочого середовища (модуль «Мої задачі») та чітких Acceptance Criteria.

- Бухгалтер – користувач фінансового модуля, який очікує безпомилкової автоматичної генерації PDF-інвойсів.

- Адміністратор інфраструктури – відповідає за технічну стабільність, безпеку даних та управління ролями.

#### 2. Зовнішні стейкхолдери:

- Замовник (B2B Клієнт) – кінцевий споживач послуг студії. Його головний інтерес полягає у зручності подання заявок (через Telegram-бот) та прозорості відстеження статусу проєкту через інтерактивний кабінет.

Таблиця 3.2 – Реєстр стейкхолдерів проєкту

| <b>ID</b> | <b>Стейкхолдер (Ім'я, Посада, Відділ)</b>  | <b>Роль у проєкті</b> | <b>Переважні канали комунікації</b>             | <b>Головні очікування та вимоги</b>                                                                                  | <b>Матриця оцінки(Вплив / Інтерес / Ставлення)</b>     | <b>Коментар / Стратегія залучення</b>                                                              |
|-----------|--------------------------------------------|-----------------------|-------------------------------------------------|----------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|----------------------------------------------------------------------------------------------------|
| st-1      | Іванов А. П.<br>Директор (Керівництво)     | Спонсор проєкту       | Наради, Email (тел..., email...)                | Очікування: Оптимізація роботи студії, зростання прибутку. Вимоги: Дотримання бюджету проєкту та строків релізу MVP. | Вплив: 10<br>Інтерес: Високий<br>Ставлення: Прихильник | Затверджує фінансування інтеграцій (Google Sheets, Telegram API). Стратегічний контроль.           |
| st-2      | B2B Клієнти<br>Власники бізнесу (Зовнішні) | Замовник проєкту      | Telegram-бот, Кабінет клієнта (Дані з веб-форм) | Очікування: Прозорість розробки, швидкий фідбек. Вимоги: Адаптивний інтерфейс, безпека конфіденційних даних.         | Вплив: 8<br>Інтерес: Високий<br>Ставлення: Прихильник  | Зацікавлені сторони, що висувають функціональні вимоги до проєкту, формуючи базовий Score (зміст). |

Продовження таблиці 3.2

|      |                                                      |                    |                                           |                                                                                                                              |                                                        |                                                                                           |
|------|------------------------------------------------------|--------------------|-------------------------------------------|------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|-------------------------------------------------------------------------------------------|
| st-3 | Камаєв С. В.<br>Project Manager<br>(Управління)      | Керівник проєкту   | Jira, Telegram, Наради (тел..., Telegram) | Очікування: Швидка обробка нових лідів, контроль задач. Вимоги: Безперебійна робота CRM-панелі, усунення людського фактора.  | Вплив: 9<br>Інтерес: Високий<br>Ставлення: Прихильник  | Є операційним ядром системи; потребує консолідованих аналітичних даних.                   |
| st-4 | Шевченко І. І.<br>Senior Developer<br>(Розробка)     | Член команди       | Jira, Slack (email..., Jira)              | Очікування: Мінімізація рутини, відсутність мікроменеджменту. Вимоги: Коректне відображення ТЗ, зручна зміна статусів задач. | Вплив: 7<br>Інтерес: Середній<br>Ставлення: Прихильник | Працює виключно у модулі «Мої задачі» згідно з принципом найменших привілеїв.             |
| st-5 | Петренко М. О.<br>Головний бухгалтер<br>(Фінансовий) | Користувач системи | Email, CRM (тел..., CRM)                  | Очікування: Автоматизація виставлення рахунків. Вимоги: Абсолютна коректність фінансових даних у згенерованих PDF-інвойсах.  | Вплив: 6<br>Інтерес: Високий<br>Ставлення: Прихильник  | Не має доступу до коду та інших модулів; повністю ізольована фінансова роль (RBAC).       |
| st-6 | Лисенко Д. В.<br>DevOps / SysAdmin<br>(IT)           | Адміністратор      | Jira, Email (email...)                    | Очікування: Стабільність системи під навантаженням. Вимоги: Наявн. логування дій користувачів та гнучкого налаштування RBAC. | Вплив: 7<br>Інтерес: Середній<br>Ставлення: Нейтрал    | Відповідає за безпеку інтеграції з Telegram API та налаштування серверної інфраструктури. |

### 3.1.3 Матриця відповідальності (RACI) та розподіл ролей

Враховуючи, що команда проєкту працює за матричною організаційною структурою (де розробники поєднують внутрішній проєкт із комерційними B2B-замовленнями), критично важливим є чіткий розподіл повноважень. Для цього застосовано методологію RACI (Responsible, Accountable, Consult, Inform).

Таблиця 3.3 – Матриця розподілу відповідальності RACI

| Процедура / Роль                                       | Клієнт | Спонсор<br>(Директор) | Адміністратор | Бухгалтер | Розробник | Бізнес-аналітик | Менеджер<br>проєкту |
|--------------------------------------------------------|--------|-----------------------|---------------|-----------|-----------|-----------------|---------------------|
| <b>1. Ініціація проєкту та стратегічне планування</b>  | I      | A                     |               |           |           | C               | R                   |
| <b>2. Збір вимог, написання User Stories та ТЗ</b>     | C      | I                     |               |           | C         | R               | A                   |
| <b>3. Налаштування ролей доступу (RBAC) та безпеки</b> |        | C                     | R, A          |           | I         |                 | I                   |
| <b>4. Прийом заявок та призначення розробників</b>     | I      |                       |               |           | I         |                 | R, A                |
| <b>5. Написання програмного коду (виконання задач)</b> | I      |                       |               |           | R, A      | C               | C                   |
| <b>6. Автоматичне формування інвойсів та оплат</b>     | I      | I                     |               | R, A      |           |                 | C                   |
| <b>7. Тестування, аудит логів та контроль якості</b>   |        |                       | C             |           | R         | R               | A                   |
| <b>8. Закриття проєкту та передача результату</b>      | C      | I                     |               | I         |           |                 | R, A                |

Аналіз ключових процедур:

1. Прийом заявок та призначення розробників (Процедура 4): Менеджер проєкту виступає як виконавець і єдиний відповідальний (R, A), що повністю відповідає його Use Case («Призначити розробника»). Розробник лише отримує сповіщення (I) про нове завдання у модулі «Мої задачі».

2. Написання коду (Процедура 5): Розробник несе повну відповідальність за технічну реалізацію (R, A). Менеджер та Бізнес-аналітик виступають у ролі консультантів (C), щоб гарантувати відповідність коду початковим вимогам (Acceptance Criteria).

3. Фінансовий контроль (Процедура 6): Бухгалтер є єдиним виконавцем і відповідальним (R, A) за генерацію PDF-інвойсів. Менеджер лише консультує щодо статусу готовності проєкту (C), а Клієнт отримує фінальний документ (I). Це ідеально відображає принцип ізоляції фінансового модуля.

4. Управління доступом (Процедура 3): Адміністратор одноосібно виконує і відповідає (R, A) за налаштування системи безпеки. Інші учасники команди лише інформуються (I) про зміну їхніх прав.

#### 3.1.4 План управління комунікаціями

Спираючись на Реєстр стейкхолдерів та матрицю RACI, було розроблено План управління комунікаціями. Його головна мета — усунути інформаційну асиметрію між замовником та командою розробки, використовуючи омніканальні системи (Telegram API, Jira) та інтерактивний «Кабінет клієнта».

Таблиця 3.4 – План (матриця) комунікацій проєкту

| Відправник / Одержувач | Спонсор (Директор)                              | Керівник проєкту (Менеджер)                                            | Бізнес-аналітик                                              | Розробник                                                | Бухгалтер                                         | Адміністратор                                       | Замовник (Клієнт)                                                                  |
|------------------------|-------------------------------------------------|------------------------------------------------------------------------|--------------------------------------------------------------|----------------------------------------------------------|---------------------------------------------------|-----------------------------------------------------|------------------------------------------------------------------------------------|
| Спонсор (Директор)     |                                                 | Стратегічні цілі, затвердження бюджетів інтеграцій (Наради, Email)     | Погодження ключових вимог (Наради)                           |                                                          | Затвердження фінансових планів (Email, Наради)    | Затвердження інфраструктурних витрат (Email)        |                                                                                    |
| Керівник проєкту       | Звіти про статус розробки, метрики (CRM, Email) |                                                                        | Передача вимог клієнта для формалізації (Jira, Slack)        | Призначення задач, контроль виконання (Jira, CRM-панель) | Передача статусу проєкту "Готово до оплати" (CRM) | Запити на зміну прав доступу команди (Jira)         | Оновлення статусів проєкту, запити на уточнення (Кабінет клієнта, Telegram, Email) |
| Бізнес-аналітик        |                                                 | Узгоджені User Stories та Acceptance Criteria (Jira)                   |                                                              | Передача ТЗ та критеріїв BDD (Jira, Slack)               |                                                   |                                                     | Уточнення бізнес-вимог та логіки (Email, Наради)                                   |
| Розробник              |                                                 | Оновлення статусів («у роботі», «виконано») (Jira, Мої задачі)         | Запити на уточнення логіки Acceptance Criteria (Jira, Slack) |                                                          |                                                   | Запит на доступ до серверів/баз даних (Slack, Jira) | - (пряма комунікація ізольованою системою)                                         |
| Бухгалтер              | Фінансові звіти (Google Sheets експорт, Email)  | Підтвердження отримання оплат (CRM)                                    |                                                              |                                                          |                                                   |                                                     | PDF-інвойси, акти виконаних робіт (Email)                                          |
| Адміністратор          | Звіт з аудиту безпеки (Email)                   | Логи активності користувачів в (CRM)                                   | Шаблони документів для роботи (CRM)                          | Надання технічних доступів (Jira)                        |                                                   |                                                     |                                                                                    |
| Замовник (Клієнт)      |                                                 | Подання заявки (опис, бюджет), узгодження ТЗ (Telegram-бот, веб-форма) | Затвердження алгоритмів роботи (Email)                       |                                                          | Надання юридичних реквізитів (Веб-форма)          |                                                     |                                                                                    |

Таблиця 3.5 – План комунікацій щодо подій проєкту

| <b>Подія</b>                                              | <b>Засіб комунікації</b>        | <b>Відправник інформації</b> | <b>Одержувач інформації</b> | <b>Технологія комунікації</b>      |
|-----------------------------------------------------------|---------------------------------|------------------------------|-----------------------------|------------------------------------|
| Запуск проєкту                                            | Проведення наради (Kick-off)    | Менеджер проєкту / Спонсор   | Команда проєкту             | Google Meet / Zoom                 |
| Завершення етапу проєкту                                  | Автоматичне сповіщення / Звіт   | Менеджер проєкту             | Замовник (Клієнт) / Спонсор | Кабінет клієнта / Telegram / Email |
| Настання планової віхи                                    | Зведення (Статус-репорт)        | Менеджер проєкту             | Спонсор                     | Jira / Експорт у Google Sheets     |
| Затвердження ключового проєктного документа (ТЗ, Матриця) | Надсилання на узгодження        | Бізнес-аналітик              | Менеджер / Замовник         | Електронна пошта / Jira            |
| Виникнення проблеми                                       | Текстове повідомлення (Блокер)  | Розробник                    | Менеджер проєкту            | Slack / Jira (коментар до задачі)  |
| Необхідність внесення зміни до проєкту                    | Запит на зміну (Change Request) | Замовник / Менеджер          | Спонсор / Бізнес-аналітик   | Електронна пошта / Telegram        |
| Завершення проєкту                                        | Передача інвойсу та продукту    | Менеджер / Бухгалтер         | Замовник (Клієнт)           | CRM (Кабінет клієнта) / Email      |

Як видно з плану комунікацій, взаємодія технічних виконавців (розробників) та фінансового сектору (бухгалтерів) із замовником здійснюється опосередковано через Менеджера та автоматизований «Кабінет клієнта». Це повністю усуває ризики мікроменеджменту та витоку конфіденційної комерційної інформації.

### 3.1.5. Управління змінами проєкту (PCR)

В умовах розробки програмного забезпечення за гнучкими (Agile) методологіями еволюція вимог є природним процесом. Однак безконтрольне додавання нового функціоналу призводить до явища «Scope Creep»

(розмиття меж проєкту). Для забезпечення інженерної дисципліни в проєкті «KSV Code Studio» імплементовано процедуру Project Change Request (PCR).

У процесі бета-тестування MVP-версії CRM-системи було виявлено вузьке місце (bottleneck) в операційному циклі: клієнти B2B-сектору витрачають від 1 до 3 днів на формальне погодження надісланих на email технічних завдань (ТЗ) та інвойсів, що затримує старт розробки. Для вирішення цієї проблеми ініційовано Запит на зміну CR-001 (табл. 3.6).

Таблиця 3.6 – Формалізований запит на внесення змін (PCR) CR-001

| Найменування поля         | Опис                                                                                                                                                                                                                                                                                                                                                                                           |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Номер запиту на зміну     | CR-001                                                                                                                                                                                                                                                                                                                                                                                         |
| Назва                     | Інтеграція модуля інтерактивного погодження документів (ТЗ та інвойсів) через Telegram-бот.                                                                                                                                                                                                                                                                                                    |
| Пріоритет                 | Високий (High)                                                                                                                                                                                                                                                                                                                                                                                 |
| Інформація про ініціатора | Камаєв С. В. (Менеджер проєкту) на основі зворотного зв'язку від фокус-групи B2B-клієнтів.                                                                                                                                                                                                                                                                                                     |
| Детальний опис зміни      | Розширення існуючої інтеграції з Telegram API. Замість простого надсилання текстових нотифікацій про статус (що було затверджено у базовому плані), бот повинен надсилати згенерований PDF-інвойс або ТЗ разом із вбудованими інтерактивними кнопками Inline Keyboard («Погодити» / «Відхилити з коментарем»). Відповідь клієнта має автоматично оновлювати статус документа в базі даних CRM. |

Продовження таблиці 3.6

|                                       |                                                                                                                                                                                                                                                                                                                                                  |
|---------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Причини, що викликали потребу у зміні | Неефективність електронної пошти як каналу швидкого реагування. Аналіз процесу виявив, що очікування email-погодження збільшує загальний час обробки замовлення (Lead Time) в середньому на 48 годин. Клієнти вимагають можливості керувати процесами безпосередньо зі смартфона в один клік.                                                    |
| Вплив зміни на проєкт                 | Зміст (Score): Збільшується (потрібна розробка обробників webhook для Telegram та оновлення архітектури БД для збереження цифрових підписів/погоджень). Якість (Quality): Зростає задоволеність клієнтів (CX). Ризики: Зростає ризик інформаційної безпеки (необхідно налаштувати додаткову авторизацію в боті для підтвердження особи клієнта). |
| Вплив змін на інші проєкти            | Потребує внесення змін до Плану комунікацій - заміна електронної пошти на Telegram як основного каналу для погодження документації. Оновлення Use Case діаграм.                                                                                                                                                                                  |

Формалізація запитів на зміну (PCR) спільно зі Статутом та матрицею комунікацій забезпечує Спонсору проєкту прозорість інвестицій, а Керівнику проєкту – легітимну підставу для коригування базового розкладу та обсягу робіт розробників, надійно захищаючи архітектуру CRM-системи від деградації.

### 3.2 Управління роботами та ресурсами проєкту

Відповідно до стандартів управління проєктами (PMBOK), успішна реалізація інформаційної системи вимагає чіткої декомпозиції загального обсягу робіт (Score) на керовані елементи та обґрунтованого розподілу людських і матеріальних ресурсів. Управління роботами у проєкті «KSV

Code Studio» базується на побудові ієрархічної структури робіт (ICP/WBS), а управління ресурсами – на формуванні крос-функціональної команди з використанням матричної організаційної структури.

### 3.2.1 Ієрархічна структура робіт (WBS) проєкту

Для ефективного управління розкладом та вартістю проєкту весь життєвий цикл розробки CRM-системи було декомпозовано на 49 керованих операцій. Ієрархічна структура робіт (ICP) структурована за п'ятьма логічними фазами: Ініціація (AS-IS), Проєктування, Реалізація (Код), Контроль якості та Перехід до цільової моделі (TO-BE). Для кожної операції визначено тривалість, залежності (Predecessors) та прив'язку до календарного графіка за принципом автоматичного планування (Auto Scheduled). Детальний перелік проєктних завдань, згенерований у середовищі Microsoft Project, наведено у Таблиці 3.7.

Таблиця 3.7 – Ієрархічна структура робіт (WBS) проєкту розробки CRM-системи

| Ідентифікатор | Ім'я                         | Тривалість | Початок              | Завершення            | Попередники |
|---------------|------------------------------|------------|----------------------|-----------------------|-------------|
| 0             | MVP Development Project      | 91 днів    | 29 Червень 2026      | 02 Листопад 2026      |             |
| 1             | Початок проєкту              | 0 днів     | 29 Червень 2026      | 29 Червень 2026       |             |
| 2             | ФАЗА 1: ІНІЦІАЦІЯ (AS-IS)    | 16 днів    | 29 Червень 2026 8:00 | 20 Липень 2026 17:00  |             |
| 3             | Збір бізнес-вимог            | 3 днів     | 29 Червень 2026 8:00 | 01 Липень 2026 17:00  | 1           |
| 4             | Аналіз AS-IS процесів        | 4 днів     | 02 Липень 2026 8:00  | 07 Липень 2026 17:00  | 3           |
| 5             | Розробка концепції (TO-BE)   | 5 днів     | 08 Липень 2026 8:00  | 14 Липень 2026 17:00  | 4           |
| 6             | Ідентифікація стейкхолдерів  | 1 день     | 15 Липень 2026 8:00  | 15 Липень 2026 17:00  | 5           |
| 7             | Написання Статуту проєкту    | 2 днів     | 16 Липень 2026 8:00  | 17 Липень 2026 17:00  | 5;6         |
| 8             | Закупівля ліцензій ПЗ (Jira) | 1 день     | 20 Липень 2026 8:00  | 20 Липень 2026 17:00  | 7           |
| 9             | Затвердження Статуту (Віха)  | 0 днів     | 20 Липень 2026 17:00 | 20 Липень 2026 17:00  | 8           |
| 10            | ФАЗА 2: ПРОЄКТУВАННЯ         | 70 днів    | 21 Липень 2026 8:00  | 26 Жовтень 2026 17:00 | 9           |
| 11            | Побудова BPMN діаграм        | 3 днів     | 27 Липень 2026 8:00  | 29 Липень 2026 17:00  | 9           |

## Продовження таблиці 3.7

|        |                               |         |                        |                        |          |
|--------|-------------------------------|---------|------------------------|------------------------|----------|
| 1<br>2 | Пріоритизація (MoSCoW, Капо)  | 1 день  | 30 Липень 2026 8:00    | 30 Липень 2026 17:00   |          |
| 1<br>3 | Розробка UML Use Case         | 4 днів  | 21 Липень 2026 8:00    | 24 Липень 2026 17:00   |          |
| 1<br>4 | Написання User Stories (BDD)  | 4 днів  | 31 Липень 2026 8:00    | 05 Серпень 2026 17:00  | 11;12    |
| 1<br>5 | Матриця трасування (RTM)      | 3 днів  | 22 Липень 2026 8:00    | 24 Липень 2026 17:00   | 13FF     |
| 1<br>6 | UI/UX дизайн Кабінету клієнта | 3 днів  | 12 Жовтень 2026 8:00   | 14 Жовтень 2026 17:00  | 14       |
| 1<br>7 | UI/UX дизайн CRM-панелі       | 4 днів  | 27 Липень 2026 8:00    | 30 Липень 2026 17:00   | 15SS     |
| 1<br>8 | Архітектура БД та RBAC        | 5 днів  | 20 Жовтень 2026 8:00   | 26 Жовтень 2026 17:00  | 15;16    |
| 1<br>9 | Завершення проєктування       | 0 днів  | 30 Липень 2026 17:00   | 30 Липень 2026 17:00   | 17       |
| 2<br>0 | ФАЗА 3: РЕАЛІЗАЦІЯ (КОД)      | 63 днів | 31 Липень 2026 8:00    | 27 Жовтень 2026 17:00  | 19       |
| 2<br>1 | Оренда Cloud Hosting          | 1 день  | 27 Жовтень 2026 8:00   | 27 Жовтень 2026 17:00  | 18       |
| 2<br>2 | Налаштування CI/CD            | 4 днів  | 31 Липень 2026 8:00    | 05 Серпень 2026 17:00  |          |
| 2<br>3 | Створення БД та ролей RBAC    | 5 днів  | 31 Липень 2026 8:00    | 06 Серпень 2026 17:00  |          |
| 2<br>4 | Модуль автентифікації         | 4 днів  | 13 Серпень 2026 8:00   | 18 Серпень 2026 17:00  | 22       |
| 2<br>5 | Інтеграція Telegram API       | 4 днів  | 07 Серпень 2026 8:00   | 12 Серпень 2026 17:00  | 22SS+2 д |
| 2<br>6 | Логіка кнопок (CR-001)        | 5 днів  | 19 Серпень 2026 8:00   | 25 Серпень 2026 17:00  | 24       |
| 2<br>7 | Фронтенд CRM-панелі           | 7 днів  | 02 Вересень 2026 8:00  | 10 Вересень 2026 17:00 | 23;25    |
| 2<br>8 | Інтерактивний Кабінет клієнта | 5 днів  | 26 Серпень 2026 8:00   | 01 Вересень 2026 17:00 | 26       |
| 2<br>9 | Генерація PDF-інвойсів        | 5 днів  | 17 Вересень 2026 8:00  | 23 Вересень 2026 17:00 | 27       |
| 3<br>0 | Інтеграція Google Sheets API  | 4 днів  | 11 Вересень 2026 8:00  | 16 Вересень 2026 17:00 | 28SS     |
| 3<br>1 | Злиття модулів (Integration)  | 5 днів  | 30 Вересень 2026 8:00  | 06 Жовтень 2026 17:00  | 28;29    |
| 3<br>2 | Розробка MVP завершена        | 0 днів  | 16 Вересень 2026 17:00 | 16 Вересень 2026 17:00 | 30       |

## Закінчення таблиці 3.7

|        |                                     |            |                          |                           |           |
|--------|-------------------------------------|------------|--------------------------|---------------------------|-----------|
| 3<br>3 | ФАЗА 4: КОНТРОЛЬ ЯКОСТІ             | 26<br>днів | 17 Вересень 2026<br>8:00 | 22 Жовтень 2026<br>17:00  | 32        |
| 3<br>4 | Code Review<br>бекенду/фронтенду    | 3 днів     | 15 Жовтень 2026<br>8:00  | 19 Жовтень 2026<br>17:00  | 31        |
| 3<br>5 | Модульне тестування за RTM          | 4 днів     | 24 Вересень 2026<br>8:00 | 29 Вересень 2026 17:00    |           |
| 3<br>6 | Оренда тестових серверів            | 1 день     | 17 Вересень 2026<br>8:00 | 17 Вересень 2026 17:00    |           |
| 3<br>7 | High Load тестування API            | 3 днів     | 07 Жовтень 2026<br>8:00  | 09 Жовтень 2026<br>17:00  | 35        |
| 3<br>8 | Аудит інформаційної безпеки         | 3 днів     | 18 Вересень 2026<br>8:00 | 22 Вересень 2026<br>17:00 | 36        |
| 3<br>9 | UAT тестування інвойсів             | 3 днів     | 20 Жовтень 2026<br>8:00  | 22 Жовтень 2026<br>17:00  | 37        |
| 4<br>0 | MVP протестовано (Віха)             | 0 днів     | 19 Жовтень 2026<br>17:00 | 19 Жовтень 2026<br>17:00  | 34;3<br>8 |
| 4<br>1 | ФАЗА 5: ПЕРЕХІД ДО TO-BE            | 7 днів     | 23 Жовтень 2026<br>8:00  | 02 Листопад 2026<br>17:00 | 40        |
| 4<br>2 | Підготовка відео-гайдів             | 2 днів     | 23 Жовтень 2026<br>8:00  | 26 Жовтень 2026<br>17:00  | 39        |
| 4<br>3 | Тренінг для Sales та PM             | 2 днів     | 26 Жовтень 2026<br>8:00  | 27 Жовтень 2026 17:00     |           |
| 4<br>4 | Тренінг для Бухгалтерії             | 1 день     | 23 Жовтень 2026<br>8:00  | 23 Жовтень 2026<br>17:00  | 42S<br>S  |
| 4<br>5 | Розгортання MVP на продакшн         | 2 днів     | 28 Жовтень 2026<br>8:00  | 29 Жовтень 2026<br>17:00  | 42;4<br>3 |
| 4<br>6 | Оплата розширених квот API          | 1 день     | 26 Жовтень 2026<br>8:00  | 26 Жовтень 2026<br>17:00  | 44        |
| 4<br>7 | Відключення старих таблиць<br>AS-IS | 1 день     | 28 Жовтень 2026<br>8:00  | 28 Жовтень 2026<br>17:00  | 44S<br>F  |
| 4<br>8 | Пілотний запуск (2 замовлення)      | 3 днів     | 29 Жовтень 2026<br>8:00  | 02 Листопад 2026<br>17:00 | 44        |
| 4<br>9 | Закриття проєкту (Віха)             | 0 днів     | 28 Жовтень 2026<br>17:00 | 28 Жовтень 2026<br>17:00  | 47        |

## 3.2.2 Формування команди проєкту та розподіл ролей

Розробка ІТ-продуктів за гнучкими (Agile) методологіями вимагає залучення крос-функціональної команди, здатної самостійно вирішувати завдання на кожному етапі SDLC. Для організації роботи впроваджено жорстку рольову модель (Role-Based Access Control), яка чітко розмежовує зони відповідальності між стейкхолдерами. Розподіл ключових ролей та посадових обов'язків наведено у Таблиці 3.8.

Таблиця 3.8 – Розподіл ролей і відповідальності в команді проєкту

| №  | Роль                                              | Функції та обов'язки                                                                                                                                                                                                                                                           | ПІБ             |
|----|---------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|
| 1. | Менеджер проєкту (Project Manager)                | Здійснює керівництво проєктом на всіх етапах, забезпечує комунікацію із замовником. Є операційним ядром системи: обробляє нові заявки в CRM, комунікує з B2B-клієнтами, аналізує деталі та делегує задачі, використовуючи функцію «Призначити розробника».                     | Камаєв С. В.    |
| 2. | Бізнес-аналітик (Business Analyst)                | Займається аналізом предметної області, з'ясуванням вимог та обмежень. Відповідає за класифікацію вимог, розробку матриці трасування (Traceability Matrix), формалізацію User Stories та Acceptance Criteria (BDD).                                                            | Коваленко О. М. |
| 3. | Розробник (Developer)                             | Займається проєктуванням, написанням програмного коду та початковим тестуванням. Є технічним виконавцем. Працює в ізольованому модулі системи «Мої задачі» (згідно з принципом найменших привілеїв), оновлює статус виконання робіт, який синхронізується з кабінетом клієнта. | Шевченко І. І.  |
| 4. | Менеджер з фінансів (Finance Manager) / Бухгалтер | Контролює рух грошових потоків, бюджетування та управління вартістю. У системі здійснює виключно фінансовий контроль: ініціює автоматичне формування інвойсів на основі проєктних даних (коли проєкт переходить у стадію «Оплата») та експортує PDF-документи.                 | Петренко М. О.  |

Продовження таблиці 3.8

|    |                                                       |                                                                                                                                                                                                                                           |                               |
|----|-------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------|
| 5. | Адміністратор проекту (Project Administrator) / Адмін | Здійснює загальне керівництво інфраструктурою, забезпечує команду необхідним середовищем. В ІТ-системі керує ролями користувачів, налаштовує права доступу, створює та редагує шаблони документів (ТЗ). Проводить аудит дій користувачів. | Лисенко Д. В.                 |
| 6. | Спонсор (Куратор) проекту                             | Надає матеріальну підтримку проекту. Здійснює стратегічний контроль за реалізацією екосистеми, затверджує цілі (MVP) та виділяє ресурси на інтеграцію (оплату серверів, Google Sheets API, Telegram API).                                 | Директор студії (Іванов А.П.) |

Визначення ролей є лише першим кроком кадрового забезпечення. Для гарантування якості продукту необхідно залучити фахівців із відповідними компетенціями. Враховуючи специфіку B2B-сектору, проект вимагає управлінських (Soft Skills), інженерних (інтеграція API, проектування баз даних) та операційних (фінансовий облік, кібербезпека) компетенцій. Комплексний план забезпечення проекту людськими ресурсами, що регламентує джерела залучення кадрів та методику їх відбору, наведено у Таблиці 3.9.

Таблиця 3.9 – План забезпечення проєкту людськими ресурсами

| Посада                | ПІБ             | План забезпечення персоналом (джерело)           | Вимоги до кандидата (прецеденти та компетенції)                                                                                                                          | Методика залучення та відбору персоналу                                                                      | Посадові інструкції                                                                                                                                    |
|-----------------------|-----------------|--------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| Менеджер проєкту (PM) | Камаєв С. В.    | Внутрішній ресурс (існуючий співробітник студії) | Знання методологій Agile (Scrum/Kanban), високі комунікаційні навички B2B-взаємодії, досвід роботи з Jira. Лідерство, націленість на кінцевий результат.                 | Оцінка результативності у попередніх проєктах, тестування на стресостійкість, аналіз soft skills.            | Організація роботи команди, прийом нових заявок від клієнтів, призначення завдань розробникам у CRM, контроль строків та комунікація із замовником.    |
| Бізнес-аналітик (BA)  | Коваленко О. М. | Внутрішній резерв / Переведення з іншого відділу | Аналітичне мислення, вміння формалізувати вимоги (User Stories у форматі Given-When-Then), побудова UML-діаграм (Use Case, Activity), володіння методами MoSCoW та Kanb. | Аналіз резюме, тестове завдання зі складання матриці трасування (RTM) та проєктування бізнес-процесу у BPMN. | З'ясування бізнес-обмежень, складання ТЗ, перетворення бізнес-вимог на Acceptance Criteria, погодження архітектурних рішень із командою розробки.      |
| Розробник (Developer) | Шевченко І. І.  | Зовнішній найм (рекрутинг) або внутрішній резерв | Технічна експертиза у веб-розробці. Знання архітектури баз даних, досвід роботи з Telegram API та Google Sheets API. Вміння писати чистий код, що легко тестується.      | Технічна співбесіда (Live Coding), аналіз портфоліо на GitHub, оцінка архітектурного мислення, рекомендації. | Проєктування та написання програмного коду, оновлення технічних статусів у Jira («Мої задачі»), початкове тестування інтегрованих компонентів системи. |

Продовження таблиці 3.9

|                                 |                |                                                     |                                                                                                                                                                                |                                                                                                                                       |                                                                                                                                                |
|---------------------------------|----------------|-----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| Менеджер з фінансів (Бухгалтер) | Петренко М. О. | Аутсорсинг (фінансова компанія)                     | Глибоке знання фінансового обліку B2B сектору, уважність до деталей, досвід генерації інвойсів та роботи з цифровим документообігом (експорт у PDF).                           | Перевірка кваліфікаційних сертифікатів, аналіз відгуків з попередніх місць роботи, інтерв'ю.                                          | Контроль бюджету проєкту, автоматичне формування інвойсів при переході заявки у статус "Оплата", виставлення рахунків клієнтам.                |
| Адміністратор інфраструктури    | Лисенко Д. В.  | Зовнішній найм (спеціаліст з кібербезпеки / DevOps) | Компетенції у сфері інформаційної безпеки. Знання систем керування доступом на основі ролей (RBAC), вміння працювати з серверною інфраструктурою та здійснювати логування дій. | Технічне інтерв'ю, перевірка знань стандартів кібербезпеки (напр. ISO 27001), випробувальний термін (конфігурація тестового сервера). | Налаштування рівнів доступу користувачів, створення та адміністрування шаблонів документів, проведення щомісячного аудиту безпеки CRM-системи. |

### 3.2.3 Ресурсне планування та матричне завантаження

Економічна ефективність внутрішньої (In-house) розробки досягається завдяки використанню матричної організаційної структури. На 100% завантаження (повний робочий день) до проєкту залучено лише Менеджера проєкту та Бізнес-аналітика. Решта фахівців (Розробник, Адміністратор, Бухгалтер) мають часткове завантаження (від 20% до 80%), паралельно виконуючи зовнішні комерційні замовлення студії «KSV Code Studio». Окрім трудових ресурсів, для реалізації проєкту залучено матеріальні активи (серверна інфраструктура) та витратні ресурси (ліцензії на ПЗ, квоти API). Зведений аркуш ресурсів (Resource Sheet) із визначенням максимального завантаження (Max. Units) та погодинних ставок (Std. Rate) наведено у Таблиці 3.10.

Таблиця 3.10 – Аркуш ресурсів проєкту (Resource Sheet)

| Ідентифікатор | Ім'я                              | Тип      | Макс. одиниці | Звичайна ставка |
|---------------|-----------------------------------|----------|---------------|-----------------|
| 0             |                                   | Робота   | 100%          | 0,00 ₴/г        |
| 1             | Камаєв С. В. (Проектний менеджер) | Робота   | 100%          | 400,00 ₴/г      |
| 2             | Шевченко І. І. (Розробник)        | Робота   | 100%          | 450,00 ₴/г      |
| 3             | Коваленко О. М. (Бізнес-аналітик) | Робота   | 100%          | 400,00 ₴/г      |
| 4             | Лисенко Д. В. (Адміністратор)     | Робота   | 100%          | 350,00 ₴/г      |
| 5             | Петренко М. О. (Бухгалтер)        | Робота   | 100%          | 250,00 ₴/г      |
| 6             | Іванов А. П. (Спонсор)            | Робота   | 100%          | 0,00 ₴/г        |
| 7             | Cloud Hosting (AWS)               | Матеріал | 4400 ₴        |                 |
| 8             | ПЗ Ліцензії (Jira, Figma)         | Витрати  |               |                 |
| 9             | Тестові сервери (High Load)       | Витрати  |               |                 |
| 10            | Аудит безпеки                     | Витрати  |               |                 |
| 11            | Квоти API (Telegram/Google)       | Витрати  |               |                 |

Після визначення пулу ресурсів було здійснено їх призначення на конкретні завдання ієрархічної структури робіт (Resource Assignment). Такий підхід дозволив розрахувати точні трудовитрати (Work) для кожної операції, уникнути перевантаження членів команди (Overallocation) та сформувати базову вартість проєкту (Cost Baseline), яка буде детально проаналізована у наступному підрозділі.

### 3.3 Управління розкладом та вартістю проєкту

Згідно зі стандартами PMBOK, ефективна реалізація IT-проєкту вимагає чіткої синхронізації між розкладом (Schedule Management) та бюджетом (Cost Management). Після побудови ієрархічної структури робіт та матричного розподілу ресурсів було проведено оптимізацію календарно-мережевого графіка та фінансове планування розробки CRM-системи «KSV Code Studio».

#### 3.3.1 Визначення тривалості робіт та план за віхами

Завдяки використанню методології Agile та розпаралелюванню ключових інженерних завдань (наприклад, одночасна розробка модулю

автентифікації та інтеграція Telegram API), загальну тривалість проєкту розробки MVP вдалося оптимізувати до 91 робочого дня. Критичний шлях проєкту (Critical Path) проходить через найбільш ресурсомісткі задачі: «Фронтенд CRM-панелі» (7 днів), «Інтерактивний Кабінет клієнта» (5 днів) та «Злиття модулів» (5 днів).

Для забезпечення стратегічного контролю з боку Спонсора проєкту (Директора) розклад структуровано за ключовими контрольними точками – віхами (Milestones), які мають нульову тривалість та сигналізують про завершення важливих фаз життєвого циклу. План за віхами наведено у Таблиці 3.11.

Таблиця 3.11 – План проєкту за ключовими віхами

| Назва контрольної віхи         | Запланована дата | Пов'язана фаза проєкту    |
|--------------------------------|------------------|---------------------------|
| 1. Затвердження Статуту (Віха) | 20.07.2026       | Фаза 1: Ініціація (AS-IS) |
| 2. Завершення проєктування     | 30.07.2026       | Фаза 2: Проєктування      |
| 3. Розробка MVP завершена      | 16.09.2026       | Фаза 3: Реалізація (Код)  |
| 4. MVP протестовано (Віха)     | 19.10.2026       | Фаза 4: Контроль якості   |
| 5. Закриття проєкту (Віха)     | 28.10.2026       | Фаза 5: Перехід до TO-BE  |

### 3.3.2 Оцінка вартості та складання бюджету

Оскільки екосистема розробляється як внутрішній інфраструктурний проєкт (In-house) для потреб самої студії «KSV Code Studio», оцінка вартості ресурсів базується на концепції «внутрішньої собівартості» (Prime Cost). Замість комерційних (Billable) ставок до розрахунку закладено базові погодинні ставки членів команди (від 250 до 450 €/год), що дозволило уникнути штучного роздування бюджету.

З урахуванням часткового матричного завантаження команди, а також витрат на хмарну інфраструктуру (Cloud Hosting), ліцензії та аудит безпеки, консолідований бюджет проєкту (Budget at Completion, BAC) склав 454 640,00 €. Такий обсяг інвестицій є економічно обґрунтованим і гарантує

високий показник рентабельності (ROI) за рахунок ліквідації ручної праці та прискорення циклу розробки для майбутніх B2B-замовлень студії.

### 3.3.3 Моніторинг та контроль проєкту (Метод освоєного обсягу)

Для об'єктивної оцінки ефективності виконання проєкту на етапі реалізації застосовується методика освоєного обсягу (Earned Value Management, EVM) [10, с. 68]. Цей інструмент дозволяє інтегровано аналізувати динаміку виконання розкладу та освоєння бюджету, виявляючи відхилення на ранніх стадіях.

У межах контролю проєкту розробки CRM-системи «KSV Code Studio» було змодельовано ситуацію проміжного аудиту на екваторі життєвого циклу (орієнтовно 45-й день розробки).

Вихідні дані для аналізу (на момент перевірки):

1. Загальний бюджет проєкту (BAC): 454 640,00 ₴
2. Планова вартість запланованих робіт (BCWS або PV): 227 320,00 ₴ (заплановано виконати 50% робіт).
3. Фактична вартість виконаних робіт (ACWP або AC): 240 000,00 ₴ (реально витрачені кошти на зарплати та сервери).
4. Освоєний обсяг (BCWP або EV): 250 052,00 ₴ (фактично виконано 55% від загального обсягу робіт).

Для проведення аудиту було розраховано показники відхилень та індекси ефективності:

1. Відхилення від календарного плану (SV):  $SV = BCWP - BCWS = 250\,052 - 227\,320 = +22\,732$  ₴ (Оскільки  $SV > 0$ , проєкт випереджає графік виконання робіт).

2. Відхилення за вартістю (CV):  $CV = BCWP - ACWP = 250\,052 - 240\,000 = +10\,052$  ₴ (Оскільки  $CV > 0$ , на поточному етапі наявна економія бюджету).

3. Індекс виконання розкладу (SPI):  $SPI = BCWP / BCWS = 250\,052 / 227\,320 = 1,10$  (Значення  $SPI > 1$  підтверджує, що темпи виконання робіт на 10% вищі за планові).

4. Індекс виконання вартості (CPI):  $CPI = BCWP / ACWP = 250\,052 / 240\,000 = 1,04$  (Показник  $CPI > 1$  демонструє ефективне використання коштів: на кожен витрачений гривню проєкт генерує 1,04 грн цінності).

5. Попередня оцінка по завершенню (EAC):  $EAC = ACWP + (BAC - BCWP) / CPI = 240\,000 + (454\,640 - 250\,052) / 1,04 \approx 436\,719$  ₴ (Очікувана фінальна вартість проєкту буде меншою за початковий бюджет).

Таблиця 3.12 – Аналіз показників освоєного обсягу (EVM) проєкту KSV Code Studio

| Показник | Формула розрахунку          | Результат  | Управлінське трактування             |
|----------|-----------------------------|------------|--------------------------------------|
| SV       | $BCWP - BCWS$               | + 22 732 ₴ | Випередження плану розкладу          |
| CV       | $BCWP - ACWP$               | + 10 052 ₴ | Економія проєктних коштів            |
| SPI      | $BCWP / BCWS$               | 1,10       | Темпи робіт вищі за планові на 10%   |
| CPI      | $BCWP / ACWP$               | 1,04       | Висока ефективність освоєння бюджету |
| EAC      | $ACWP + (BAC - BCWP) / CPI$ | 436 719 ₴  | Проєкт обійдеться дешевше за BAC     |
| VAC      | $BAC - EAC$                 | + 17 921 ₴ | Очікувана фінальна економія          |

Висновки за результатами моніторингу: Розраховані індекси SPI (1,10) та CPI (1,04) науково доводять, що обрана гібридна методологія управління (Agile + PMBOK) разом із матричним ресурсним плануванням забезпечують високу ефективність реалізації. За термінами проєкт демонструє позитивну динаміку, випереджаючи календарний графік, а за вартістю – спостерігається чітка економія бюджету. На основі розрахованого прогнозу (EAC), проєкт буде завершено успішно з очікуваною фінальною економією у розмірі 17 921 ₴ від початкового бюджету, що створює управлінський резерв для розширення функціоналу системи в майбутньому (наприклад, реалізації вимог із категорії Excitement за моделлю Kano).

### 3.4 Управління ризиками та якістю проєкту

Відповідно до стандартів PMBOK, управління ризиками (Project Risk Management) та якістю (Project Quality Management) є нерозривними безперервними процесами. Невиявлені інфраструктурні та управлінські загрози на етапі розробки безпосередньо призводять до деградації продукту та зростання "вартості невідповідності якості" (Cost of Poor Quality) – тобто збільшення витрат на виконання додаткових компенсаційних робіт через дефекти.

#### 3.4.1 Ідентифікація ризиків та стратегії реагування

Для фриланс-студії «KSV Code Studio» успішна реалізація внутрішньої CRM-екосистеми критично залежить від стабільності архітектури, безпеки даних B2B-сегменту та дотримання розкладу в умовах матричної організаційної структури. На основі розробленої архітектури було проведено ідентифікацію ключових ризиків, визначено їхні причини та сформовано комплексний План управління ризиками (Таблиця 3.13). Для управління загрозами застосовуються класичні стратегії: зниження, уникнення, розподіл та самострахування (прийняття).

Таблиця 3.13 – План управління ризиками проєкту «KSV Code Studio»

| Тип ризику                                                        | Причини та вірогідні наслідки                                                                                                                    | Стратегія управління               | План запобігання та реагування                                                                                                                            |
|-------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Технічний ризик: Відмова системи під навантаженням (High Load) | Причини:<br>Одночасна обробка великої кількості запитів з Telegram-бота.<br>Наслідки:<br>Зависання CRM-панелі, втрата нових заявок від клієнтів. | Зниження ризику та Самострахування | Запобігання:<br>Впровадження асинхронних черг для API та оптимізація бази даних. Реагування:<br>Автоматичне горизонтальне масштабування хмарних серверів. |

Продовження таблиці 3.13

|                                                                  |                                                                                                                                                           |                                     |                                                                                                                                                                                                                                               |
|------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2. Ризик безпеки:<br>Компрометація конфіденційних даних клієнтів | Причини:<br>Помилки в налаштуванні рольової моделі (RBAC), людський фактор.<br>Наслідки: Витік комерційної таємниці, судові позови за порушення NDA.      | Уникнення та Розподіл ризику        | Запобігання: Суворе дотримання політик RBAC. Регулярний аудит логів активності користувачів.<br>Реагування: Негайне блокування скомпрометованих акаунтів, залучення DevOps-фахівців хостинг-провайдера за SLA.                                |
| 3. Управлінський:<br>Зрив розкладу проєкту (Schedule Overrun)    | Причини:<br>Матрична структура: розробники відволікаються на комерційні B2B замовлення студії. Наслідки: Відтермінування релізу MVP, збільшення вартості. | Самострахування (збереження ризику) | Запобігання:<br>Уникнення неконтрольованого розширення меж (Scope Creep) через обов'язкові запити на зміну (PCR).<br>Реагування:<br>Залучення заздалегідь закладених часових буферів. Тимчасове переведення команди на 100% завантаження CRM. |
| 4. Ризик якості:<br>Наявність критичних дефектів у релізі        | Причини: Поспіх перед дедлайном, неповне тестове покриття коду.<br>Наслідки: Некоректні суми в PDF-інвойсах, втрата лояльності замовників.                | Зниження ризику                     | Запобігання: 100% покриття функціоналу тест-кейсами згідно з Матрицею трасування (RTM).<br>Реагування:<br>Терміновий випуск патчу (hotfix) або відкат фінансового модуля до стабільної версії (rollback).                                     |

### 3.4.2 Планування якості та причинно-наслідковий аналіз дефектів

Забезпечення якості (Quality Assurance) проєкту базується на циклі безперервного вдосконалення (PDCA: Plan-Do-Check-Act). Основні контрольні процедури в проєкті спираються на використання Матриці трасування вимог (RTM), регулярний аудит інформаційної безпеки та навантажувальне тестування ключових вузлів. Комплексний план контролю якості наведено у Таблиці 3.14.

Таблиця 3.14 – План управління якістю проєкту «KSV Code Studio»

| Захід з управління якістю                           | Відповідальний         | Строк виконання   | Критерії оцінки якості                                                                                         |
|-----------------------------------------------------|------------------------|-------------------|----------------------------------------------------------------------------------------------------------------|
| 1. Затвердження Матриці (RTM) та UI/UX дизайну      | Бізнес-аналітик        | Етап проєктування | 100% покриття бізнес-вимог сценаріями тестування.<br>Відповідність макетів (Figma) логіці BPMN.                |
| 2. Code Review та модульне тестування бекенду       | Розробник (Tech Lead)  | Протягом розробки | Відсутність критичних блокерів у коді.<br>Синхронна зміна статусів у Jira та «Кабінеті клієнта».               |
| 3. Навантажувальне тестування (High Load API)       | Адміністратор (DevOps) | Перед релізом     | Час автоматичної генерації PDF-інвойсу < 2 сек. Безперебійна маршрутизація заявок з Telegram API.              |
| 4. Аудит кібербезпеки та перевірка матриці RBAC     | Адміністратор          | Щомісяця          | Неможливість доступу користувача без ролі (наприклад, Розробника) до фінансового модуля.<br>Успішне логування. |
| 5. Приймально-здавальні випробування (UAT) інвойсів | Бухгалтер              | Перед релізом     | Абсолютна точність реквізитів та розрахунків сум у тестових PDF-документах.                                    |

Для глибинного аналізу потенційних дефектів (зокрема, ризику появи багів у MVP-релізі) застосовано інструмент Загального управління якістю (TQM) – діаграму Ісікави («риб'яча кістка»). На Рисунку 3.1 візуалізовано причинно-наслідкові зв'язки факторів, що спричиняють деградацію продукту.

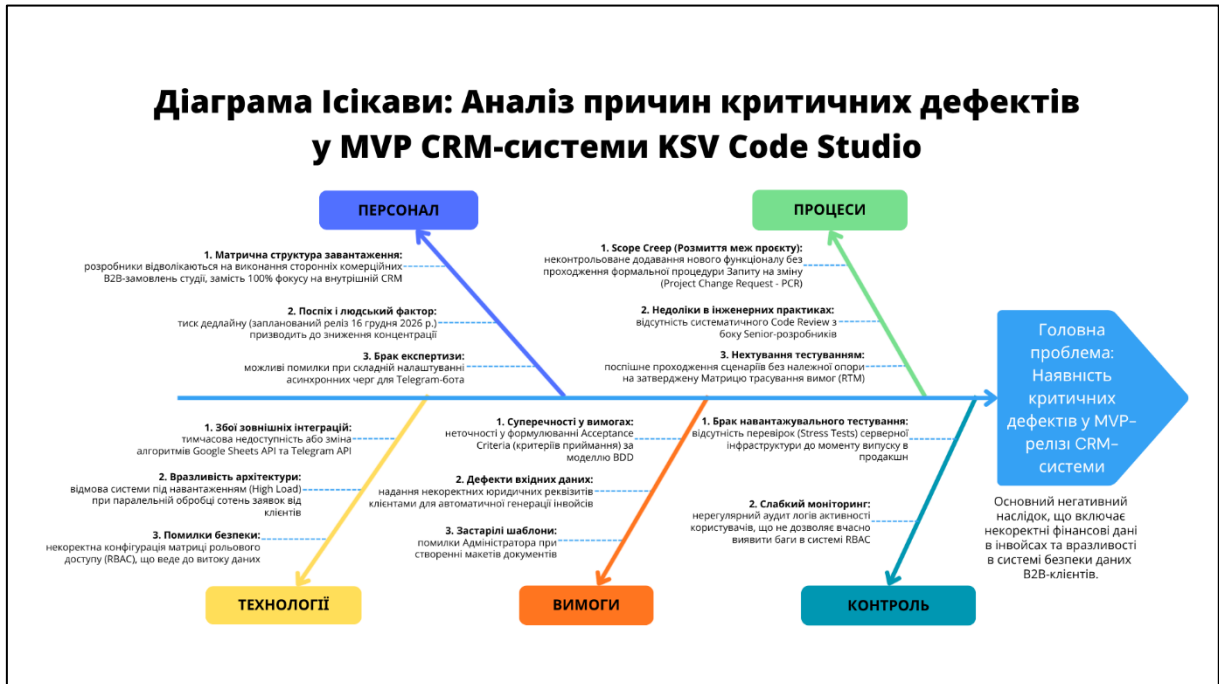


Рисунок 3.1 – Причинно-наслідкова діаграма Ісікави для дефектів CRM-системи

Побудована структура доводить, що деградація якості є наслідком не лише суто технічних збоїв (відмова серверів чи API), але й комплексних управлінських недоліків. Найбільш вагомими факторами виступають матрична структура завантаження персоналу, яка створює конфлікт пріоритетів і провокує поспіх, а також неконтрольоване розширення меж проєкту (Score Creep). Застосування діаграми Ісікави спільно з розробленими реєстрами дозволяє Керівнику проєкту перейти від реактивного виправлення багів до проактивного забезпечення надійності екосистеми, блокуючи першопричини їх виникнення.

### Висновки до розділу 3

У третьому розділі було розроблено комплексний проектно-рекомендаційний інструментарій для управління життєвим циклом розробки CRM-екосистеми «KSV Code Studio». На основі отриманих результатів зроблено такі висновки:

1. Формалізація базових параметрів проекту через Статут та Реєстр стейкхолдерів забезпечила чітке визначення цілей та меж продукту. Побудова матриці відповідальності RACI та плану комунікацій ліквідувала інформаційну асиметрію між крос-функціональною командою студії та зовнішніми B2B-замовниками.

2. Розроблена Ієрархічна структура робіт (WBS), що містить 49 керованих операцій, дозволила ефективно декомпонувати життєвий цикл на 5 послідовних фаз. Матричний розподіл ресурсів забезпечив раціональне завантаження команди без відриву від їхньої основної комерційної діяльності.

3. Оптимізація мережевого графіка дозволила скоротити час реалізації мінімально життєздатного продукту (MVP) до 91 робочого дня. Бюджет проекту, розрахований за внутрішньою собівартістю (Prime Cost), склав 454640 грн. Застосування методу освоєного обсягу (EVM) науково довело високу ефективність виконання робіт ( $SPI = 1,10$ ;  $CPI = 1,04$ ) з прогнозованою економією коштів по завершенню.

4. Розроблений інтегрований план управління ризиками та якістю, підкріплений причинно-наслідковим аналізом Ісікави, сформував проактивні стратегії запобігання технічним збоям та витокам конфіденційних даних клієнтів. Це гарантує стабільну роботу системи під навантаженням та відповідність фінального IT-продукту найвищим індустріальним стандартам.

## ВИСНОВКИ

Кваліфікаційна робота присвячена вирішенню актуального інженерно-управлінського завдання – розробці та впровадженню інтегрованої цифрової CRM-екосистеми для IT-студії «KSV Code Studio». За результатами проведеного дослідження мету роботи повністю досягнуто, а всі поставлені завдання виконано. Узагальнення результатів роботи дозволяє зробити наступні висновки:

1. За результатами теоретичного дослідження обґрунтовано, що для індивідуальної B2B-розробки з високим рівнем невизначеності вимог найбільш ефективним є застосування гнучкої методології (Agile). Доведено стратегічну необхідність переходу від фрагментованих інструментів до єдиних цифрових екосистем, що дозволяє усунути комунікаційну асиметрію (через впровадження інтерактивного «Кабінету клієнта» та Telegram-бота) і забезпечити автоматизацію внутрішнього документообігу та фінансової звітності.

2. За результатами дослідницько-аналітичного етапу проведено функціональне моделювання операційної діяльності студії за методологією IDEF0. SWOT-аналіз моделі AS-IS підтвердив наявність критичних вузьких місць: значна частка ручної праці, фрагментованість даних та слабе ресурсне планування. Для подолання цих обмежень розроблено цільову модель TO-BE та дорожню карту реінжинірингу. Системні вимоги були пріоритизовані за методами MoSCoW та Kano, що дозволило визначити базовий функціонал MVP (безпека, ролі RBAC) та фактори додаткової цінності (інтеграція з Google Sheets та Telegram API). Якість майбутнього продукту забезпечено формалізацією користувацьких сценаріїв (за моделлю BDD) та розробкою Матриці трасування вимог (RTM).

3. За результатами проектно-рекомендаційного етапу розроблено повноцінний інструментарій управління життєвим циклом системи згідно зі стандартами PMBOK. Затверджено Статут проекту та план комунікацій,

розмежовано зони відповідальності крос-функціональної команди (матриця RACI). Структурна декомпозиція робіт (WBS) дозволила оптимізувати мережевий графік: тривалість розробки MVP склала 91 робочий день, а розрахований за внутрішньою собівартістю бюджет склав 454 640 грн.

4. Аналіз ефективності управління методом освоєного обсягу (EVM) підтвердив високу результативність обраних підходів: індекси виконання розкладу ( $SPI = 1,10$ ) та вартості ( $CPI = 1,04$ ) доводять випередження графіка та економію коштів. Комплексний план управління ризиками та причинно-наслідковий аналіз (діаграма Ісікави) забезпечили проактивний захист системи від відмов під навантаженням та витоку конфіденційних даних B2B-клієнтів.

Практичне значення отриманих результатів полягає в тому, що розроблена ІТ-екосистема та проєктна документація повністю готові до впровадження в операційну діяльність компанії «KSV Code Studio». Її реалізація дозволить підприємству скоротити операційні витрати, пришвидшити цикл доставки програмних продуктів (Lead Time), усунути ручну паперову роботу та створити масштабований фундамент для подальшого зростання на ринку ІТ-аутсорсингу.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бібліографічне посилання (загальні правила складання) відповідно до ДСТУ 8302:2015, запровадженого в дію в Україні 01.07.2016 : метод. рек. / Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова ; уклад. : С. В. Карпенко. – Харків : ХНУМГ ім. О. М. Бекетова, 2017. – 19 с.
2. Бушуєв С. Д., Бушуєва Н. С., Неизвестный С. І. Сучасні технології управління ІТ-проєктами: гнучкі методології та гібридні підходи. Управління проєктами та розвиток виробництва, 2021. № 2 (78). С. 15–28.
3. ДСТУ ISO/IEC/IEEE 29148:2018 (ISO/IEC/IEEE 29148:2018, IDT). Системна та програмна інженерія. Процеси життєвого циклу. Вимоги до інженерії. Київ: ДП «УкрНДНЦ», 2019.
4. Загальні вимоги до підготовки, оформлення та видання наукової і навчально-методичної літератури : метод. рек. / Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова ; уклад. О. В. Михаленко. – 3-тє вид., зі змін. та доп. – Харків : ХНУМГ ім. О. М. Бекетова, 2023. – 78 с.
5. Косенко Н. В., Чумаченко І. В. Оптимізація ресурсного планування в ІТ-проєктах з використанням матричних організаційних структур. Комп'ютерні науки та інженерія, 2022. Т. 5, № 1. С. 45–54.
6. Лисенко О. В. Впровадження CRM-систем у B2B секторі: управління змінами та подолання опору стейкхолдерів. Економіка та управління підприємствами, 2023. Вип. 42. С. 112–119.
7. Морозов В. В., Череднік О. О. Управління ІТ-проєктами: підручник. Київ: Київський національний університет імені Тараса Шевченка, 2021. 245 с.
8. Новіков О. А. Сучасні методології розробки програмного забезпечення: адаптація Agile в українському B2B секторі. Вісник Національного університету «Львівська політехніка». Серія: Інформаційні системи та мережі, 2022. Вип. 10. С. 45-53.

9. Рекомендації щодо запобігання академічному плагіату та його виявлення в наукових роботах. URL: <https://zakon.rada.gov.ua/rada/show/v8681729-18#Text> (дата звернення: 10.09.2023).

10. Рибальченко І. А. Оцінка ефективності розробки програмного забезпечення за методом освоєного обсягу (EVM). Інформаційні технології в управлінні, 2021. № 3. С. 67–75.

11. Теслюк В. М., Мельник Н. В., Зелінський А. В. Проєктування інформаційних систем: навчальний посібник. Львів: Видавництво Львівської політехніки, 2020. 312 с.

12. International Project Management Association (IPMA). Individual Competence Baseline for Project, Programme & Portfolio Management (ICB4). 4th ed. IPMA, 2015. 431 p.

13. Project Management Institute (PMI). A Guide to the Project Management Body of Knowledge (PMBOK® Guide). 7th ed. Newtown Square, PA: Project Management Institute, 2021. 370 p.

14. Project Management Institute (PMI). Agile Practice Guide. Newtown Square, PA: Project Management Institute, 2017. 168 p.

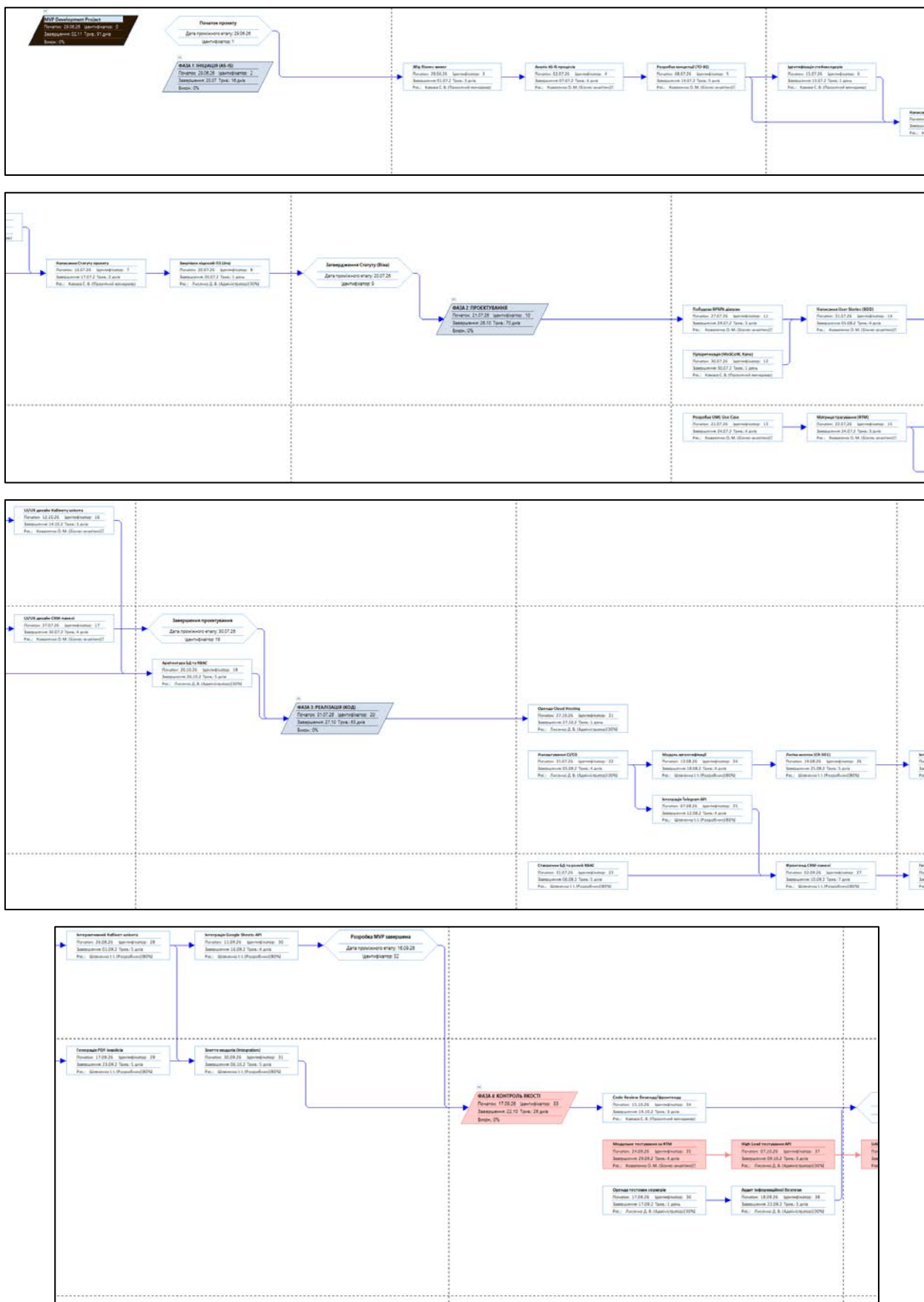
15. Schwaber K., Sutherland J. The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game. 2020. URL: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>

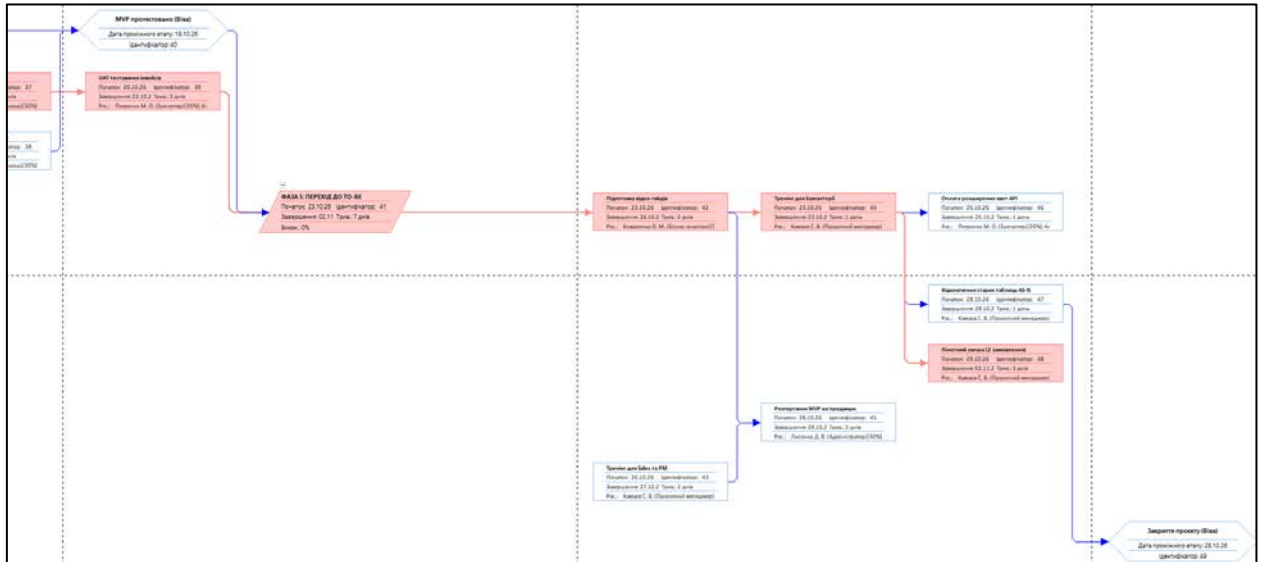
16. Sutherland J. Scrum: The Art of Doing Twice the Work in Half the Time. London : Random House Business, 2019. 256 p.

## ДОДАТКИ

## Додаток А. Діаграма Ганта

[illegible]





## Додаток В. Аркуш ресурсів

[illegible]

**Додаток Г. Графічні матеріали**