

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет інформаційних технологій

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

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА на тему:

“Дослідження процесів управління проєктом з розробки інформаційної системи електронної комерції для підприємців Amazon”

Студент 2-го курсу групи УП-21

Науковий керівник:

Владислав ГАВРИЛЮК

к.т.н., доцент

(прізвище, ім'я)

(науковий ступінь, вчене звання)

Любов КУБЯВКА

(прізвище, ім'я)

(підпис студента)

(дата)

(підпис)

Попередній захист:

(Висновок: "До захисту в Екзаменаційній комісії")

Завідувач кафедри
технологій управління

Віктор МОРОЗОВ

(підпис)

(прізвище, ініціали)

(дата)

ЗАТВЕРДЖУЮ
Завідувач кафедри
професор Морозов В.В.

“ _____ ” _____ 2024 року

ЗАВДАННЯ НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Студент: Гаврилюк Владислав Якович

Група: УП-21

1. Тема кваліфікаційної роботи: «Дослідження процесів управління проектом з розробки інформаційної системи електронної комерції для підприємців Amazon». Затверджена на засіданні кафедри технологій управління, протокол №6 від 06.11.2023 року.
2. Строк подання студентом готової роботи “10” травня 2024 р.
3. Цільова установка та вихідні дані до роботи: дослідження характеристик об'єкту управління, планування робіт проекту включаючи календарне планування, цілей, віхи, WBS, бюджету й робочих ресурсів. Робота зі стейкхолдерами проекту.
4. Зміст роботи: аналіз структури проекту, предметної галузі, документів й артефактів, проведення ринкових досліджень, виконання інвестиційних досліджень, постановка SMART цілей продукту, аналіз конкурентів, опис мети, розробка життєвого циклу проекту, побудова ієрархічної структури робіт проекту, побудова організаційної структури компанії, формування складу команди проекту та розподіл ролей і відповідальностей, дослідження БД програмної інформаційної архітектури ПЗ, календарне планування проекту, планування ресурсів проекту, обрахування вартості проекту, планування управління якістю, визначення й управління ризиками проекту, моніторинг проекту.
5. Перелік графічного матеріалу (слайдів): титульна сторінка, тема, мета, інноваційність проекту, інформація про проект, життєвий цикл проекту, вимоги та цілі проекту, деструктуризація робіт, віхи проекту, вибір інструментів комунікації, дизайну, розробки та документації, структура агенції та проекту, концептуальна та математична модель проекту, ризик

менеджмент проєкту, навігаційна архітектура, клієнт-серверна архітектура додатку, приклад Amazon Seller API інтеграції, база даних проєкту, фіксовані затрати проєкту, оцінка вартості проєкту, графік планових та виконаних показників робіт, інвестиційна привабливість проєкту, висновки, перелік посилань, дякую за увагу.

6. Календарний план виконання роботи:

№ з/п	Назва частин роботи	План виконання роботи
1	Вивчення інформації з предмету дослідження.	15.11.23 – 06.12.23
2	Збір і вивчення матеріалів досліджуваної теми.	07.12.24 – 31.01.24
3	Складання плану кваліфікаційної роботи магістра.	01.02.24 – 14.03.24
4	Ознайомлення наукового керівника з планом кваліфікаційної роботи магістра. Внесення змін.	15.03.24 – 21.03.24
5	Підготовка розділу 1	22.03.24 – 28.03.24
6	Підготовка розділу 2	29.03.24 – 04.04.24
7	Підготовка розділу 3	05.04.24 – 11.04.24
8	Підготовка розділу 4	11.04.24 – 03.05.24
9	Оформлення кваліфікаційної роботи.	12.04.24 – 17.05.24
10	Передача роботи на рецензування.	10.05.24
11	Передача роботи науковому керівнику.	11.05.24
12	Попередній захист кваліфікаційної роботи.	11.05.24
13	Захист роботи.	21.05.24 – 23.05.21

Дата видачі завдання: “09” листопада 2023 р.

Керівник роботи кандидат технічних наук, доцент Кубявка Л.Б. _____

(підпис)

Завдання прийняла до виконання:

Студента групи УП-21 Гаврилюк В.Я. _____

(підпис)

ЗМІСТ

АНОТАЦІЯ.....	6
ВИЗНАЧЕННЯ ТЕРМІНІВ ТА АБРЕВІАТУР.....	8
ВСТУП.....	10
РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПРОЄКТУ.....	14
1.1 Бізнес аналіз проєкту.....	14
1.1.1 Дослідження сфери діяльності продукту.....	14
1.1.2 Аналіз ринку та конкурентів.....	16
1.1.3 Визначення вимог та цілей проєкту.....	21
1.1.4 Дослідження стейкхолдерів проєкту.....	28
1.2 Дослідження документів та артефактів проєкту.....	32
1.3 Вибір стандартів та технологій менеджменту.....	36
1.4 Оцінка інвестиційної привабливості проєкту.....	37
1.5 Концептуальна та математична модель проєкту.....	40
РОЗДІЛ 2. ПЛАНУВАННЯ ПРОЄКТУ.....	44
2.1 Створення життєвого циклу проєкту.....	44
2.2 Побудова структури проєкту та проєктної команди.....	50
2.2.1 Візуалізація структури проєкту.....	54
2.2.2 Розподіл компетентностей, ролей та зон відповідальності.....	57
2.3 Формування бюджету та оцінювання ресурсів.....	60
2.4 Планування віх та WBS проєкту.....	62
2.5 Календарний план та таймлайни проєкту.....	66
РОЗДІЛ 3. АНАЛІЗ ІНСТРУМЕНТІВ ТА ПРОЦЕСІВ ДІЯЛЬНОСТІ ЧЛЕНІВ КОМАНДИ.....	70
3.1 Вибір інструментів комунікації, дизайну, розробки та документації.....	70
3.2 Вибір технологічного стеку.....	73

3.3 Дослідження процесів виконання дизайну.....	75
3.3.1 Дослідження процесів прототипування користувацького досвіду.....	76
3.3.2 Елементи візуалізації дизайну.....	78
3.4 Дослідження процесів розробки.....	79
3.5 Проектування бази даних.....	82
3.6 Тестування програмного продукту.....	88
РОЗДІЛ 4. МОНІТОРИНГ ВИКОНАННЯ ПРОЄКТУ ТА ЗВІТУВАННЯ.....	90
4.1 Постановка задач, робіт та дедлайнів.....	90
4.2 Контроль виконання робіт.....	92
4.3 Управління ризиками проєкту.....	94
4.3.1 Загальні положення про ризики та фреймворк MSF.....	94
4.3.2 Дослідження та ідентифікація ризиків.....	96
4.3.3 Розробка стратегії управління ризиками.....	102
4.4 Оцінювання якості виконаних робіт.....	102
4.5 Аналіз фінансових витрат та показників ефективності.....	103
ВИСНОВКИ.....	108
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	110
ДОДАТОК А.....	114
ДОДАТОК Б.....	117

АНОТАЦІЯ

Темою поданої кваліфікаційної магістерської роботи є «Дослідження процесів управління проектом з розробки інформаційної системи електронної комерції для підприємців Amazon». В свою чергу робота присвячена дослідженню та аналізу сучасного проекту в основу якого входить розробка веб додатка у сфері електронної комерції під платформу продажів Amazon.

Метою роботи постає пошук та аналіз проблем сучасних підприємців, пошук потенційних ринкових конкурентів. А також створення зручного програмного рішення яке дозволить вести ефективно та виконувати весь цикл бізнес процесів на платформі Amazon починаючи від закупки товарів, закінчуючи звітуванням про успішність продажів товарів у вигляді звітності фінансової та маркетингової аналітики.

Ціль проекту – полягає у створенні інноваційної інформаційної системи з електронної комерції, що забезпечить підприємців платформи Amazon зручним та потужним інструментом зі збору, обробки та автоматизації більшості бізнес процесів.

Наукова новизна – інноваційність B2B проекту полягає в експериментуванні з новими практиками у заліяних сферах продакт менеджменту, дизайну та розробки для досягнення максимально зручного та комплексно-автоматизованого програмного рішення з безшовними API інтеграціями сторонніх сервісів для ведення бізнесу FBA та FBM типів. Такий додаток дозволить зекономити підприємцям великий обсяг часу, масштабувати їх бізнес та зменшити кількість використання сторонніх інструментів для його ведення. Не менш важливим є новаторська розробка III модулю з аналізу користувачьких регіональних магазинів Amazon. Що дозволяє на основі досліджених ринкових патернів та практик згенерувати рекомендації щодо ведення бізнесу та попередити про певну низку ризиків власника бізнесу. Ця комбінація технологій дозволяє навіть недосвідченим підприємцям успішно вести бізнес на платформі Amazon та максимально використовувати її потенціал.

В ході роботи виконано дослідження сучасної літератури у сфері проектного менеджменту, проаналізовано ринок конкурентів, а також було закладено загальну наукову новизну в проєкт.

Робота включає визначення ключових термінів, вступ, чотири основні розділи, висновки та перелік використаних джерел.

У першому розділі проведено бізнес-аналіз проєкту, що охоплює дослідження сфери діяльності продукту, аналіз ринку та конкурентів, визначення вимог та цілей проєкту, а також дослідження стейкхолдерів.

Другий розділ присвячений плануванню проєкту, включаючи створення життєвого циклу, побудову структури проєкту та проєктної команди, формування бюджету, оцінювання ресурсів та планування таймлайнів.

Третій розділ зосереджений на аналізі інструментів та процесів діяльності членів команди, включаючи вибір інструментів комунікації, дизайну, розробки, документації та технологічного стеку.

Четвертий розділ описує моніторинг виконання проєкту та звітування, включаючи постановку задач, контроль виконання робіт, оцінювання якості та аналіз фінансових витрат.

У висновках виконано загальну оцінку ефективності досліджуваного проєкту, загальних методів управління, підсумування його процесів, ресурсів та перспектив розвитку у сучасному світі.

Ключові слова: проєкт, електронна комерція, автоматизація, архітектура програмного забезпечення, вебдодаток, ризик менеджмент, аналіз продукту.

ВИЗНАЧЕННЯ ТЕРМІНІВ ТА АБРЕВІАТУР

Абревіатура	Визначення
API	Application Programming Interface - інтерфейс програмування додатків, набір правил взаємодії між компонентами програмного забезпечення.
GIT	Розподілена система керування версіями для відстеження змін у файлах та координації роботи над ними серед кількох користувачів.
MVP	Minimum Viable Product - продукт з мінімальним набором функцій, достатнім для задоволення перших користувачів та збору зворотного зв'язку для подальшого розвитку.
Scrum	Гнучка технологія управління проектами, яка використовує ітеративний підхід для оптимізації розробки та доставки продукту.
Sprint	Спринт – ітеративний проміжок часу в проєктів зазвичай тривалістю у 2 тижні.
Retrospective	Ретроспектива – зустріч проєктної команди після закінчення спринту.
WBS	Work Breakdown Structure – ієрархічна структура розбивки робіт в проєкті.
Sprint planning	Планування спринту – зустріч з командою в розрізі якого набираються задачі на поданий спринт.
Daily meeting	Щоденні зустрічі скрам команд.
Scrum points	Одиниці вимірювання часу та складності виконання задач.
Project Lifecycle (ЖЦ)	Життєвий цикл проєкту.

ПЗ	Програмне Забезпечення.
БД	База Даних.
ТЗ	Технічне Завдання.
SDLC	Життєвий цикл програмного забезпечення.
QA	Quality Assurance – забезпечення якості програмного забезпечення, продуктів та послуги.
SaaS	Software as a Service – програмне забезпечення як сервіс, модель надання програмного забезпечення на основі передплати за функціонал.
B2B	Business to Business – вид інформаційної та економічної діяльності взаємодії коли одна бізнес організація надає послуги іншій бізнес організації.
UI	User Interface – користувацький інтерфейс програмного забезпечення.
UX	User Experience – користувацький досвід.
ACWP	Actual Cost of Work Performed – фактична вартість виконаної роботи.
BCWP	Budgeted Cost of Work Performed – бюджетована вартість виконаної роботи.
BCWS	Budgeted Cost of Work Scheduled – бюджетована вартість запланованої роботи.
KPI	Key Performance Indicators – ключові показники ефективності, застосовуються в бізнесі для вимірювання й оцінки ведення процесів та справ.

ВСТУП

У контексті стрімкого розвитку глобального ринку електронної комерції, ця магістерська робота зосереджується на аналізі його динаміки та різноманітності моделей взаємодії. Електронна комерція, яка охоплює широкий спектр форматів від B2C до P2P, є важливим елементом сучасної економіки, пропонуючи зручність та доступність для користувачів різних соціальних верств. Цифрові платформи комерції, стають все більш інтегрованими в повсякденне життя, забезпечуючи важливий канал для обміну інформацією та транзакцій.

З огляду на ці тенденції, робота прагне визначити ефективні стратегії для підприємців, що зіштовхуються з викликами, пов'язаними з високими витратами на оренду та необхідністю автоматизації бізнес-процесів. Вивчення успішних моделей, таких як Amazon, та їхніх інноваційних підходів до B2B онлайн платформ, надає цінні інсайти для розробки нових рішень у сфері електронної комерції.

Наукова новизна роботи полягає у ряді процесів проектного продукту направлених на високий рівень автоматизації та швидку роботу з великими об'ємами даних підприємців зі всього світу. Процеси створення проекту такого рівня охоплюють високорівневе програмування, набір архітектурних та інформаційних рішень, дизайну інтерфейсів, бізнес-аналітики, а також маркетингу та соціальної інженерії, які відкривають новий рівень користувацького досвіду і нові перспективи для розвитку галузі електронної комерції, уособлюючи ведення бізнесу лише з декількох пов'язаних платформ. Перша платформа у нашому випадку Amazon виступає в ролі об'єкта ведення бізнесу та отримання доходу на основі збуту товарів за своїми політиками, а друга платформа є поточним проектним рішенням яке виконує весь збір даних, проводить аналітику, дає поради ведення бізнесу, модифікує дані та автоматизує більшість аспектів ведення бізнесу в розрізі однієї платформи замінюючи цілий набір програм відразу та забезпечуючи економію часу користувачів. Завдяки аналізу актуальної літератури та інформаційних джерел,

робота робить великий вклад у вектор розвитку програмних рішень на динамічному ринку електронної комерції.

Головною метою роботи постає створення B2B SaaS платформи, яка сприятиме автоматизації та оптимізації процесів продажу, закупівель, маркетингу та фінансової аналітики. Проектна платформа, орієнтована на міжнародних підприємців, пропонує модель підписки, що забезпечує фінансову вигоду замовнику та зручність користувачам.

Об'єктом дослідження є розробка інформаційної системи процеси якої направлені на розвиток електронної комерції та створення інноваційних практик в сфері.

Предметом дослідження поточної роботи є набір виконання послідовних процесів, дій та практик в сфері проектного менеджменту направлених на створення концепту продукту, прототипування, розробку інформаційної системи, керування ресурсами проекту як матеріальними так і робочими, управління ризиками, термінами і фінансовими аспектами.

В перелік завдань до виконання направлених на дослідження відібрано наступні роботи:

- аналіз ринку сучасної електронної комерції та пошук конкурентів й ринкових альтернатив;
- визначення вимог та цілей проекту;
- дослідження стейкхолдерів;
- дослідження документів та артефактів;
- вибір технологій та стандартів менеджменту;
- оцінка інвестиційної привабливості продукту;
- створення концептуальної моделі проекту;
- відтворення життєвого циклу проекту;
- вобудова структури проекту й проектної команди;
- візуалізація структури проекту;

- розподіл компетентностей, ролей та зон відповідальності;
- формування бюджету та оцінка ресурсів;
- розробка віх й WBS проєкту;
- відтворення календарного плану та таймлайнів проєкту;
- відбір інструментів комунікації, дизайну, розробки та документації;
- вибір технологічного стеку розробки проєкту;
- детальне дослідження процесів дизайну, розробки та тестування продукту;
- постановка задач та робіт;
- контроль виконання робіт;
- управління ризиками включаючи дослідження та ідентифікацію ризиків, розробку ефективної стратегії їх управління;
- оцінювання якості виконаних робіт;
- аналіз витрат та показників ефективності проєкту.

В ході роботи застосовано наступні *методи дослідження*: евристичні, емпіричні, теоретичні, математичні та методи комп'ютерного моделювання й візуалізації.

Досліджена новизна отриманих результатів в порівнянні з поточним ринком станом на 2024 рік рішень демонструє, що проєкт удосконалює та в рази збільшує автоматизацію сучасних процесів ведення бізнесу в умовах взаємодії підприємців з однією основної платформи, як основним програмним рішенням для повного циклу ведення бізнесу. Серед досліджених конкурентів на ринку не було знайдено альтернатив на основі штучного інтелекту, які б надавали рекомендації та застерезували підприємців про проблеми з їх регіональними магазинами Amazon на основі досліджених патернів ведення бізнесу.

Практичне значення дослідження демонструє, що модулі розробленої інформаційної системи забезпечують повноцінну інвентаризацію товарів користувачів платформи починаючи з внесення контактних даних постачальників послуг зі поставки, зберігання та доставки товарів. Закінчуючи питанням створення

та звітування замовлень з детальний моніторингом дій з товарами на кожному з етапів. Фінансовий модуль в свою чергу надає можливість збирати фінансові дані та показники з платформи Amazon структурувати їх у формі фінансової звітності та встановлювати додаткові витрати і моніторити повернення товарів від незадоволених покупців. Розділ покупців та замовлень демонструє зібрані та залоговані замовлення покупців в деталях замовлення. А також дозволяє виконувати моніторинг фідбеку користувачів про покупки товарів й виконувати запити від бізнесмена до клієнта на відгуки користувачів. Модуль домашньої сторінки знайшов практичне примінення для швидкого моніторингу й звітування підприємця про стан його бізнесу через найбільш важливі модулі, графіки, віджети та таблиці зібрані по всій архітектурі сервісу. Останній але один найбільш важливий модуль штучного інтелекту асистенту виконує аналіз та звітування про проблеми бізнесу на основі досліджених ринкових патернів, показників та політик Amazon, що потребують уваги й прийняття рішень з боку підприємця.

РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПРОЄКТУ

1.1 Бізнес аналіз проєкту

Бізнес аналіз проєкту відбувається від початку життєвого циклу продукту з формуванням концепції бізнес-кейсу для виявлення життєздатності продукту. Отже, таким чином, бізнес-аналітик на чолі з проєктним менеджером та стейкхолдерами проєкту приділяють достатню кількість часу опису бізнес-моделі проєкту, визначають сферу та проблеми які вирішує продукт для потенційних користувачів.

Наступним кроком відбувається аналіз оточення проєкту в який входить перегляд зовнішніх та внутрішніх впливів, а також проєціювання таких впливів на проєкт та його діяльність занурюючись в кожен аспект. При детальному вивченні зовнішнього середовища велика кількість часу витрачається на безпосередній аналіз конкурентів, який прямо впливає на функціональні та нефункціональні вимоги до продукту та тривалість його реалізації в умовах фінансування. Не менш важливим постає виконання аналізу ризиків для оточення проєкту та генерація та впровадження протиризикових стратегій.

Додатково бізнес-аналітик визначає основні ключові метрики якими користуються сучасні підприємці Amazon для досягнення своїх цілей у сфері міжнародної електронної комерції. Варто зазначити, оскільки таких метрик достатньо велика кількість і вони збираються протягом великого проміжку часу, репрезентація ключових показників візуалізується та закладається бізнес-аналітиком у логіці продукту у вигляді графіків.

1.1.1 Дослідження сфери діяльності продукту

Проєкт зосереджується на діяльності підприємців Amazon, які потребують якісне програмне рішення для ведення свого бізнесу у сфері продажів. Їх проблематика полягає в тому, що вони потребують потужний інструмент який буде виконувати аналітику даних та звітувати їх у разі потреби, індикуючи проблеми

ведення продажів. Водночас до головних потреб таких підприємців входить бажання отримати зручну систему менеджменту стоку продуктів які знаходяться на складах, розв'язання проблем поповнення та менеджменту ресурсів на складах, а також закриття питань їх логістики та повернень товарів у разі гарантійних випадків. Не менш важливим на думку опитуваних бізнесменів є можливість отримання зворотної комунікації від покупців товарів у разі отримання негативних відгуків.

При дослідженні функціонала та політики майданчика Amazon виявлено стек певних ключових поділів ведення бізнесу з продажів. Першочергово бізнес на майданчику Amazon поділяється на два формати:

- FBM – (Fullfilment By Merchant) означає виконання замовлень продавцем. Це означає, що продавець самостійно займається зберіганням, пакуванням та відправленням своїх товарів покупцям від початку покупки до моменту продажу товару. Продавці можуть використовувати Amazon для продажу своїх товарів, але зберігати їх власними зусиллями та здійснювати доставлення до покупця самостійно.
- FBA – (Fulfillment by Amazon) виконання замовлень Amazon. Це означає, що продавець зберігає свої товари на складах Amazon, персонал якого бере на себе пакування та доставлення товарів покупцям. Крім того, Amazon також надає підтримку клієнтів та повернення товарів. Продавці можуть відправляти свої товари до складів Amazon, а потім підключати їх до своїх Amazon акаунтів для продажу. За це майданчик бере оплату за послуги FBA та має додаткову прописані особливості зберігання та транспортування товару.

Як основні коди для ідентифікації товарів та визначення їх особливостей використовують ASIN та SKU коди.

- ASIN (Amazon Standard Identification Number) – це унікальний ідентифікатор, який присвоюється кожному товару, що продається на Amazon. ASIN складається з 10 символів і може бути знайдений на сторінці товару в Amazon.

- SKU (Stock Keeping Unit) – це унікальний ідентифікатор, що використовується продавцями на Amazon для ідентифікації конкретного товару або варіанту товару. SKU може мати в собі інформацію про розмір, колір, матеріал, модель та інші характеристики товару. SKU допомагає продавцям відстежувати запаси товарів, керувати інвентарем, звітувати про продажі та відстежувати виконання замовлень.

1.1.2 Аналіз ринку та конкурентів

Метою аналізу ринку електронної комерції є пошук рішень аналогів, які намагаються зацікавити частину підприємців Amazon у своїх програмних рішеннях. Функціонал таких продуктів має концептуально розв'язувати проблеми користувачів та перекривати їх потреби при використанні набору функціональних інструментів для ведення бізнесу. Варто врахувати, що прогрес функціоналу ПЗ конкурентів не стоїть на місці, тому найкращим рішенням є можливість також ознайомитись з дорожніми картами функціоналу створених для інвесторів проєктів конкурентів.

Пошук конкурентів починається з пошуку рішень на основі ключових пошукових запитів в мережі інтернет використовуючи оператори уточнення запитів. Такі оператори надають можливість зробити пошуковий запит максимально точним та відповідним до шуканого. Дуже часто можна знайти рішення за запитом інформаційного характеру які ведуть на інформаційні портали та блоги, на яких зазвичай закупають веб трафік компанії та проєкти. Аналогічним чином інформаційні платформи та форуми було використано для пошуку реальних користувачів програмного забезпечення такого класу і наступного опитування реальних підприємців.

Під час проведення аналізу була сформовано наступну таблицю 1.1 яка відображає прямих конкурентів під платформу Amazon та їх функціональні можливості, на основі ведення B2B моделі бізнесу.

Аналіз конкурентів проєкту

Назва продукту	Бізнес-модель	Призначення	Монетизація продукту	Інструменти та функціонал продукту
1	2	3	4	5
Helium 10	B2B	Платформа ведення та оптимізації Amazon бізнесу.	Цифрові підписки: Diamond 229\$/міс., Platinum 79\$/міс., Starter 29\$/міс.	Дешборд інсайдів, система пошуку продуктів для продажів, система пошуку ключових слів, оптимізатор лістингів, функціонал менеджменту інвентарю, аналітика ринку, проведення маркетингових PPC кампаній.
Jungle Scout	B2B	Платформа для дослідження продуктів та ключових слів на Amazon.	Цифрові підписки: Basic 29\$/міс., Suite 49\$/міс., Professional 84\$/міс.	Розширення для браузера, трекер продуктів, аналізатор можливостей, продуктова БД, БД постачальників, менеджер інвентарю, FBA відшкодування, пошук ключових слів, трекер ранку, будівельник лістингів, перегляд автоматизації, аналітика продажів, рекламна аналітика, сповіщення, категорії трендів.

1	2	3	4	5
AMZScout	B2B	Платформа аналізу продуктів для Amazon.	Цифрові підписки: Advanced Booster Pack 25\$/міс., Starter Kit 100\$/міс, Seller's Bundle for Amazon 50\$/міс.	Інструменти дослідження продуктів, аналізатор ключових слів, будівельник лістингів, оцінювач продажів, розширення статистики стоку.
Teikametrics	B2B	Платформа для оптимізації продажів на Amazon і Walmart.	Цифрові підписки: Basic безплатний, AI-Powered 100\$/міс.	Менеджер реклами, оптимізатор лістингів.
Viral Launch	B2B	Платформа оптимізатор продажів для Amazon.	Цифрові підписки: Essentials 69\$/міс., Pro 99\$/міс., Pro Plus Ads 199\$/міс.	Інструмент дослідження продуктів, аналізатор ринку, аналізатор ключових слів, досліджувач конкурентів, аналізатор лістингів, менеджер ключових слів, будівельник лістингів, рекламні та PPC кампанії, генератор відгуків, функціонал A/B тестування, FBA відшкодування, бухгалтерські послуги Amazon.

1	2	3	4	5
SellerApp	B2B	Платформа аналітики продажів та проведення рекламних кампаній.	Цифрові підписки: Freemium безплатний, DIY Professional Plan 99\$/міс., Smart Plan 149\$/міс.	Сервіси PPC менеджменту, оптимізатор лістингів, сервіс менеджменту попиту Amazon DSP, рекламний менеджер, аналіз конкуренції, рішення для корпорацій та агенцій.
FeedbackWhiz	B2B	Платформа управління відгуками та продажами на Amazon.	Цифрові підписки: Starter 20\$/міс., Basic 40\$/міс., Professional 80\$/міс., Ultimate 140\$/міс.	Менеджер замовлень, функціонал збору відгуків користувачів, email кампанії, автоматичні запити на відгуки, функція сповіщення проблем, інструмент зворотного зв'язку.
eComEngine	B2B	Система лістингу та інвентаризації Amazon продуктів.	Цифрові підписки: Listing Health & Brand Protection 19\$/міс., Feedback & Reviews 24\$/міс., Access All Features 140\$/міс.	Менеджер інвентарю, протектор Amazon лістингів.

1	2	3	4	5
Zonguru	B2B	ПЗ моніторингу та дослідження Amazon продуктів.	Цифрові підписки: Researcher 29\$/міс., Seller 49\$/міс.	Інструмент дослідження продуктів, оптимізатор лістингів, моніторинг та сповіщення, аналіз бізнес метрик, система комунікації з покупцями на основі email.
Seller Legend	B2B	Розгорнута система аналітики та PPC кампаній для Amazon.	Цифрові підписки: Starter 50\$/міс., Advanced 60\$/міс., Professional 60\$/міс., Enterprise 100\$/міс.	Ринкові графіки, панель інструментів продукту, інструменти аналізу покупців, інструменти моніторингу продажів та дохідності, менеджер замовлень, менеджер інвентарю, менеджер покупців, менеджер продуктів, відстеження продуктивності PPC.

На основі проаналізованого списку конкурентів, можна побачити що вони пропонують, широкий спектр послуг та програмних рішень для підприємців, але не дивлячись на кількість поданих інструментів варто враховувати їх якість та вартість.

Деякі рішення мають занадто високу вартість за підписки з повним доступом до необхідного функціоналу. Враховуючи той фактор що це підписки щомісячного характеру до формування ціни при виході на ринок варто враховувати кількість профіту та можливий середній оберт бізнесменів, щоб покупка такої підписки була вигідна обом сторонам.

Також якщо уважно пройтись по відеоматеріалам, інструкціям та документаціям з користування та тестовим підпискам, дуже часто можна помітити недоліки таких програм. Одні ведуть маркетингове направлення бізнесу, інші надають аналітичні інструменти, але дуже мала кількість з поданих конкурентів закриває питання всього спектру ведення бізнесу включаючи менеджмент складів, стоку товарів та логістики. При всіх перелічених вище факторах, такі програмні рішення не завжди є швидкими у вивантаженні даних при клієнт-серверній комунікації та вивантаженню великих об'ємів даних, а також дружніми та зрозумілими для новачків у сфері.

Отже, таким чином продукт дипломної роботи може вирішити цю низку проблем зробивши ПЗ швидким, зручним та інтуїтивним у користуванні з додатковим функціоналом який не представлений у конкурентів і за доступну вартість.

1.1.3 Визначення вимог та цілей проєкту

Кожен проєкт, має низку цілей які прямопропорційно впливають на його майбутнє задоволення стейкхолдерів. Паралельно з цілями також генеруються та відбираються вимоги, аби проєкт відповідав потребам користувачів і мав максимальну конкурентоспроможність.

Визначення основних вимог та цілей до проєкту є одним з найголовніших питань яке надає розвиток подій та пріоритетів виконання проєкту. Сформований список вимог до продукту виглядає наступним чином:

- програмний продукт має бути розроблений в часовому розрізі від 27.03.2023 до 27.09.2024;
- програмне забезпечення має розроблюватись на основі вебдодатка і поширюватись за допомогою платних підписок;
- програмний продукт має мати блок Finances з Dashboard аналітикою продажів товарів в певному розрізі часу;

- програмний продукт має мати сторінку Home після авторизації акаунту для простежування останніх важливих подій які відбулись на майданчику Amazon;
- програмний продукт має мати блок сторінок Inventory в яких можна простежити процеси пов'язані з інвентаризацією товарів;
- програмний продукт має мати блок сторінок Orders для відстеження замовлень та зворотного зв'язку з користувачами;
- продукт має бути зручним у використанні на персональних комп'ютерах та планшетах;
- програмний продукт має мати в собі асистента який буде рекомендувати користувачу які дії краще впровадити з певним продуктом на майданчику Amazon, щоб уникнути проблем з продажами або додаткових витрат;
- документація має відображати сутність та принципи роботи готового програмного продукту.

Після того як були визначені головні вимоги до проєктного продукту найкращим чином потрібно сформувавши цілі в основу яких увійдуть головні роботи. Найкращим методом перевірки при формуванні цілей слугує SMART методика.

Ціль: Сформувавши проєктну команду для розробки продукту за перший місяць від початку роботи проєкту.

- S – Specific (конкретна): Формування проєктної команди для розробки продукту, яка складається з Back-End розробника, Business Analyst спеціаліста, UI дизайнера та UX дизайнера, Front-End двох розробників, QA тестувальника, за перший місяць від початку роботи проєкту.
- M – Measurable (вимірювана): Кількість та тип професій, які повинні бути у команді визначені, а часові рамки для досягнення цілі – перший місяць від початку роботи проєкту.
- A – Achievable (досяжна): Сформувавши проєктну команду з вищезазначеними спеціалістами протягом першого місяця є реальною задачею, оскільки на

цьому етапі проєкту важливо забезпечити належну комунікацію та взаємодію між різними частинами команди.

- R – Relevant (відповідна): Сформування проєктної команди є важливою складовою для успішного виконання проєкту розробки продукту, оскільки вона відповідальна за розробку та тестування програмного забезпечення.
- T – Time-bound (термінова): Кінцевий строк досягнення цілі – перший місяць від початку роботи проєкту, що дозволяє організувати час та працю таким чином, щоб виконати ціль вчасно.

Ціль: З допомогою Back-End розробника зарезервувати домен та орендувати сервер для розробки програмного забезпечення на період в 5 років.

- S – Specific (конкретна): Зарезервувати домен та орендувати сервер для розробки програмного забезпечення на період в 5 років за допомогою Back-End розробника.
- M – Measurable (вимірювана): Можна виміряти, чи була ціль досягнута, перевіривши наявність зарезервованого домену та орендованого сервера для розробки програмного забезпечення на період в 5 років.
- A – Achievable (досяжна): Резервування домену та оренда сервера можуть бути досягнуті з допомогою Back-End розробника, оскільки він зазвичай займається налаштуванням сервера та інфраструктурою, необхідною для розробки програмного забезпечення.
- R – Relevant (відповідна): Резервування домену та оренда сервера є важливими етапами в розробці програмного забезпечення, оскільки це дозволяє забезпечити надійний та безперебійний доступ до сервера.
- T – Time-bound (термінова): Кінцевий строк досягнення цілі – протягом періоду в 5 років. Це означає, що необхідно зарезервувати домен та орендувати сервер якомога швидше, щоб забезпечити надійну інфраструктуру для розробки програмного забезпечення на тривалий період часу.

Ціль: Провести бізнес-аналітику головних показників якими користуються бізнесмени Amazon для інтеграції в програмне забезпечення, починаючи з продуктових кодів, фінансових показників, закінчуючи маркетинговими показниками рейтингу продажів продуктів. Обмеження часу виконати до 27.05.2023.

- S – Specific (конкретна): Провести бізнес-аналітику головних показників, якими користуються бізнесмени Amazon, для інтеграції в програмне забезпечення починаючи з продуктових кодів, фінансових показників, закінчуючи маркетинговими показниками рейтингу продажів продуктів.
- M – Measurable (вимірювана): Можна виміряти, чи була ціль досягнута, зіставивши результати бізнес аналізу з головними показниками, якими користуються бізнесмени Amazon.
- A – Achievable (досяжна): Проведення бізнес-аналітики головних показників, якими користуються бізнесмени Amazon, є досяжним завданням, оскільки є відповідні ресурси та інструменти для збору та аналізу даних.
- R – Relevant (відповідна): Проведення бізнес-аналітики головних показників, якими користуються бізнесмени Amazon, є важливим етапом в інтеграції програмного забезпечення з Amazon, оскільки дозволяє збільшити ефективність та прибутковість продажів.
- T – Time-bound (термінова): Кінцевий строк досягнення цілі – до 27.05.2023. Це означає, що необхідно провести бізнес-аналітику головних показників якомога швидше, щоб мати достатньо часу на інтеграцію програмного забезпечення з Amazon.

Ціль: Створити бекенд розробником базу даних та серверну архітектуру для сервісу, враховуючи взаємодію API та парсерів з Amazon. Покрити питання розробки за термін 4 місяці від початку проекту.

- S – Specific (конкретна): створити базу даних та архітектуру для бекенду сервісу, враховуючи взаємодію API та парсерів з Amazon.

- M – Measurable (вимірювана): створення бази даних та архітектури бекенду сервісу.
- A – Achievable (досяжна): ціль можливо досягнути з використанням наявних ресурсів та знань.
- R – Relevant (відповідна): ціль відповідає потребам проєкту, оскільки вона необхідна для розробки бекенду сервісу та взаємодії з Amazon.
- T – Time-bound (обмежена в часі): покрити питання розробки за термін 4 місяці від початку проєкту.

Ціль: С кооперувати дію UI дизайнера та UX дизайнера і на основі проведеної бізнес-аналітики створити прототипи інтерфейсів за третій місяць та дизайн користувачького інтерфейсу додатка за 3 та 4 місяці.

- S – Specific (Конкретна): С кооперувати дію UI дизайнера та UX дизайнера і на основі проведеної бізнес-аналітики створити дизайн.
- M – Measurable (Вимірювана): Ціль можна виміряти через кількість відпрацьованих днів від початку проєкту до закінчення робіт.
- A – Achievable (Досяжна): Створення прототипів і дизайну користувачького інтерфейсу за 3 та 4 місяці є досяжною метою, якщо в команді є відповідні спеціалісти та не буде несподіваних труднощів.
- R – Relevant (Відповідна): Ціль пов'язана зі створенням програмного забезпечення та необхідна для забезпечення комфортної роботи користувачів.
- T – Time-bound (Обмежена в часі): Ціль має обмеження часу – створення прототипів за третій місяць, а дизайну користувачького інтерфейсу за 3 та 4 місяці від початку проєкту.

Ціль: Зверстати користувачький інтерфейс продукту під персональні комп'ютери, планшети та налаштувати запити на взаємодії користувача з інтерфейсом, обравши API та бібліотеки для реалізації графіків. Почати розробку фронт-енду з 4-го місяця закінчуючи 8-го місяця.

- S – Specific (Конкретна): створення користувацького інтерфейсу продукту під різні пристрої та налаштування запитів.
- M – Measurable (Вимірювана): успішним виконанням цієї мети буде створення відповідного робочого користувацького інтерфейсу.
- A – Achievable (Досяжна): дана мета досяжна, оскільки вже є плани та розроблені відповідні API та бібліотеки для реалізації графіків.
- R – Relevant (Актуальна): ця мета є актуальною, оскільки вона є складовою частиною розробки продукту та відповідає вимогам та потребам користувачів.
- T – Time-bound (Має часові рамки): дана мета повинна бути виконана протягом 5 місяців.

Ціль: Покривати QA-тестувальником програмне забезпечення продукту на наявність функціональних проблем та помилок після кожної ітерації підготовки програмного забезпечення і злиття на тестову гілку GIT системи. Виконання тестування проводити протягом всього життєвого циклу проєкту.

- S – Specific (конкретна): Покрити програмне забезпечення продукту QA тестувальником на наявність функціональних проблем та помилок після кожної ітерації.
- M – Measurable (вимірювана): Перевірка на наявність помилок після кожної ітерації підготовки програмного забезпечення і злиття на тестову гілку GIT системи.
- A – Achievable (досяжна): Тестування проводити протягом всього життєвого циклу проєкту.
- R – Relevant (відповідна): Забезпечення якості продукту через виявлення та виправлення помилок.
- T – Time-bound (термінова): Виконати під час всього життєвого циклу проєкту.

Для розбиття ієрархії цілей використовують наступний інструмент. Дерево цілей – це наочне графічне зображення підпорядкованості та взаємозв'язку цілей, що

демонструє розподіл загальної мети або місії на підцілі, завдання та окремі дії. Дерево цілей можна визначити, як цільовий каркас організації, явища чи діяльності [1]. За допомогою цієї інформації створено структуру дерева цілей й продемонстровано його ієрархію на (рис. 1.1).

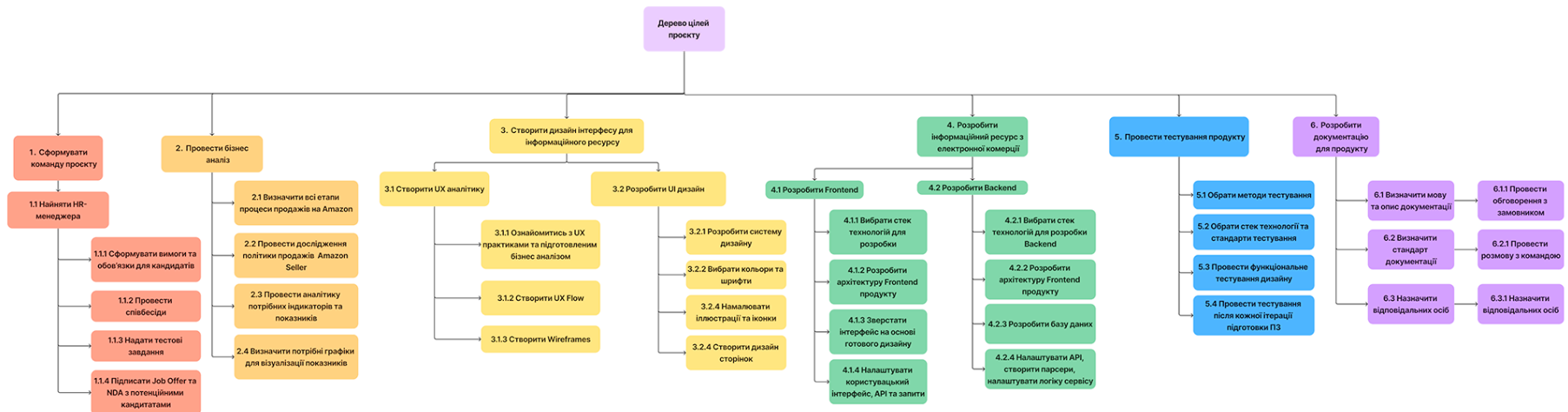


Рис. 1.1. Візуалізація дерева цілей проекту

Ілюстрація дерева цілей формує та пояснює мету й об'єктивні умови виконання проекту, а також коригування структури об'єкта і напрямку його робіт. А декомпозиція допомагає розкриттю структурності цілей системи проекту формуючи основу для створення деструктуризації робіт проекту. Таким чином було розглянуто найбільш важливі аспекти, пов'язані з визначенням вимог та цілей проекту. Вимоги є основою для подальшого проектування та розробки, а цілі визначають напрямок роботи. Основним чином враховано потреби замовника, технічні обмеження та можливості проекту в умовах обмеженості ресурсів при створенні вимог і цілей.

1.1.4 Дослідження стейкхолдерів проєкту

При створенні будь-якого проєкту в його зовнішнє та внутрішнє оточення входять нові групи осіб, їх найменують стейкхолдерами. Стейкхолдери – це зацікавлені сторони у виконанні проєкту, вони їх увага та вплив мають високу цінність та вплив на успішність проєкту [4].

За такою тенденцією вони мають певний вплив на внутрішні зміни проєкту. Головними стейкхолдерами проєкту з розвитку електронної комерції стали наступні сторони:

- Конкуренти – організації або групи осіб, які створюють продукти аналоги на їх основі й таким чином формується конкурентоспроможність на ринку.
- Замовник бізнесу – однойменна особа за бажанням якої був створений бізнес проєкт. Головним чином формує вимоги, корегує цілі фінансує проєкт та визначає часові рамки його виконання.
- Керівник проєкту – особа яка бере на себе відповідальність за створення проєкту.
- Інвестори – особи зацікавлені у майбутньому проєкті, головним чином додатково фінансують проєкт або надають окремі послуги які можуть посприяти розробці.
- Команда проєкту – наймана та сформована група осіб з однією ціллю створенням якісного проєктного продукту.
- Партнери – компанії або організації, які надають послуги та технології які сприяють отриманню певних ресурсів у нашому випадку програмних;
- Уряд – орган виконавчої влади в державі. В нашому випадку уряд сприяє вступу до Євросоюзу. Таким чином він надасть змогу розширити маркетплейс Amazon на Україну забезпечивши громадян новою ланкою бізнесу.
- Користувачі – кінцеві особи, в нашому випадку бізнесмени Amazon Seller які отримують доступ до нашого платного продукту. Метою користування їх є можливості автоматизації та спрощення ведення бізнесу, отримання фінансової

звітності, інформація зі зберігання та транспортування товарів і отримання маркетингових показників просування товарів маркетплейсі.

- Маркетингові агенції – компанії які надають послуги з просування рішень та бізнесів у цифровому форматі.
- Медіа – особи які мають масовий інформаційний вплив на велику аудиторію потенційних користувачів.
- Органи регулювання – зазвичай цілі організації які контролюють низку стандартів індустрії, та надають дозволи на низку дій, наприклад Google яка контролює політику GDPR в сучасних пошукових системах.

На основі поданих проаналізованих типів стейкхолдерів було сформовано матрицю впливу (рис. 1.2) та розібрано, як потрібно впливати на певні касти стейкхолдерів головним чином.

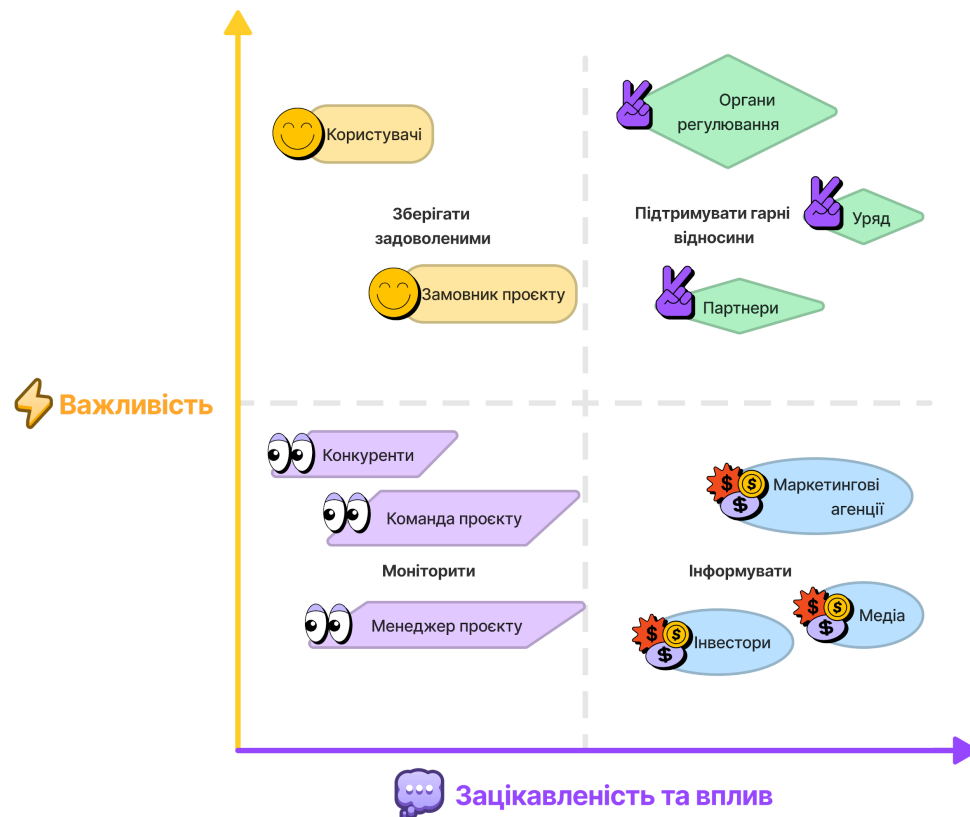


Рис. 1.2. Матриця впливу стейкхолдерів

Як можна побачити з попередньої ілюстрації матриця впливу несе в собі розділення на 4 когорти стейкхолдерів особи яких потрібно: зберігати задоволеними, моніторити, інформувати, підтримувати гарні відносини. Також для частини найбільш важливих стейкхолдерів створено реєстр стейкхолдерів в таблиці 1.2., щоб зберігати їх персональну інформацію та завжди мати можливість підтримання активної комунікації.

Таблиця 1.2

Реєстр стейкхолдерів проєкту

Посада, та повне ім'я особи або організації	Контакти стейкхолдерів	Роль на проєкті	Локація	Нотатки
1	2	3	4	5
Проектний менеджер Гаврилюк Владислав Якович	Електронна пошта: Gavryliuk_vlad@knu.ua Моб. телефон: + 38(093)**-**-***	Впровадження управлінських рішень, менеджмент робочих ресурсів.	Україна, м. Київ.	Онлайн у будь-який час.
Замовник проєкту Jonh Doe	Електронна пошта: John_Doe@gmail.com Моб. телефон: + 1(415)**-**-***	Постачання вимог до проєкту, фінансування.	США, Каліфорнія	Турбувати після 23:00.
Бізнес-аналітик Піддубний Дмитро Миколайович	Email: Dmytro_Analyst@gmail.com Моб. телефон: + 1(415)**-**-***	Збір та аналітика вхідних даних для застосунку. Моніторинг ринку конкурентів.	Україна, м. Київ	Працює асинхронно вночі.

1	2	3	4	5
Back-end розробник Винник Петро Олександрович	Електронна пошта: Petro_Vynnik@gmail.com Моб. телефон: + 38(067)**_**_***	Розробка серверної архітектури та бази даних вебзастосунку.	Україна, м. Київ	Закінчує працювати о 19:00.
Front-end розробник Артеменко Максим Глібович	Електронна пошта: Maks_dev@gmail.com Моб. телефон: + 38(067)**_**_***	Верстка дизайну інтерфейсу, налаштування користувацького інтерфейсу.	Україна, м. Київ	-
QA-тестувальник Коваленко Оксана Вікторівна	Електронна пошта: Kovalenko Oksana1254@gmail.com Моб. телефон: + 38(097)**_**_***	Тестування ПЗ, документації.	Україна, м. Київ	-
UI дизайнер Баккан Андрій Батькович	Електронна пошта: Bakkan Andrew@gmail.com Моб. телефон: + 38(066)**_**_***	Розробка екранів дизайну інтерфейсу.	Україна, м. Київ	Працює на вихідних
UX дизайнер Ярмак Гліб Петрович	Електронна пошта: Yarmak_Gleb@gmail.com Моб. телефон: + 38(066)**_**_***	UX-аналітика інтерфейсу користувача, розробка прототипів інтерфейсу.	Україна, м. Київ	Тимчасово закордоном
HR-менеджер Полянська Поліна Юрїївна	Електронна пошта: Polyanskaya_HR@gmail.com Моб. телефон: + 38(066)**_**_***	Наймання людських ресурсів та організація івентів.	Україна, м. Харків	Фігурує в проєкті перших 2 місяці

1	2	3	4	5
Конкурент Helium 10	https://www.helium10.com/	Потенційний конкурент.	США.	Має занадто складний інтерфейс та дорогий тариф
Конкурент Seller legend	https://sellerlegend.com/ Електронна пошта: contact@sellerlegend.com	Потенційний конкурент.	Британія	Проблеми з графіками, не просте підключення магазинів.
Партнер Amazon Seller	https://sellercentral.amazon.com/	Партнер який надає платформу для продажів та вихідні дані через API.	Головний офіс США	-
Стратегічний інвестор Andrew Hale	Електронна пошта: a.hale@gmail.com	Бажає інвестувати додаткові кошти.	Румунія	Часто на зв'язку.

1.2 Дослідження документів та артефактів проєкту

В розділі розглядаються документи проєкту пов'язаних з розробкою B2B SaaS проєкту під потреби бізнесменів під платформу Amazon. Першопочатково варто зауважити, що дані проєкту надійно зберігаються, обробляються та поширюються на основі веб продуктів Google Drive та Atlassian Confluence з якими достатньо зручно працювати в одному корпоративному середовищі.

Умовно документація проєкту поділяється на декілька розділів: документація з бізнес-процесів та менеджменту, документація з користувацького досвіду та дизайну, документація з розробки.

В документацію з бізнесу та менеджменту входять наступні документи:

- **паспорт проекту** – документ який засвідчує початок проекту, зазначаючи в ньому головних чинних осіб проекту, його умови, цілі та ресурси;
- **вимоги до програмного продукту** – містять в собі збір функціональних та нефункціональних вимог, які потрібно впровадити програмний продукт задля задоволення замовників тощо.
- **реєстр стейкхолдерів** – документ який демонструє дані усіх заохочених осіб які мають вплив на процеси успіху проекту;
- **документація конкурентів** – список з потенційними конкурентами їх продуктами, особливостями, загрозами та потенційними можливостями в умовах конкуренції на ринку покупців;
- **дорожня карта** – чудовий інструмент для демонстрації цілей проекту які команда має покрити в довготривалому життєвому циклі проекту;
- **фінансові формули та графіки** – для демонстрації команді, які бізнес показники важливі для покупців майбутнього продукту і типу візуалізації вихідних даних на основі стану бізнесу;
- **розрахункова вартість проекту** – орієнтовна вартість яку проектна команда затратить на повне виконання проекту.
- **бізнес діаграми** – документи які демонструють бізнес-процеси взаємодії майбутніх користувачів з програмним продуктом.

Документи з дизайну та користувацького досвіду:

- **ментальна карта** – документ та інструмент візуалізації взаємодії користувача з продуктом його побажань та потреб в певний момент часу відносно певної сторінки вебресурсу;
- **стандарти дизайну** – технічні умови виконання дизайну в які входять обрані девайси під які розроблюється дизайн, кольори, візуальні типи даних, шрифти, мова анімацій;

- **інформаційна архітектура та структура** – сюди входить візуалізація навігації майбутнього програмного забезпечення, та умови взаємодії користувача зі сторінками вебресурсу;
- **прототипи** – артефакти які дозволяють візуалізувати ідеї замовника та проаналізувати та відвалідувати ідеї для розвитку концепту дизайну перед розробкою;
- **керівництва для користувачів** – документи які включають опис сторінок ресурсу, елементів їх функціонального призначення для того, щоб швидше ознайомити користувачів зі всім майбутнім функціоналом проекту.
Документація для розробників та тестувальників:
- **артефакт з доступами** – документ який уособлює в собі список збір від доступів до різних інформаційних ресурсів які потрібні для успішної розробки програмного продукту.
- **стандарти тестування** – документ в якому прописано, яким чином відбувається тестування програмного забезпечення та валідація дизайнів;
- **звіти з тестувань** – документ з ітераційними описами знайдених проблем та помилок в програмному забезпеченні, які потребують виправлень;
- **технологічний стек** – артефакт зі списком всіх необхідних технологій на основі яких буде розроблений програмний ресурс;
- **технічне завдання** – встановлює основне призначення, показники якості, техніко-економічні та спеціальні вимоги до виробу, обсягу, стадії розроблення та складу конструкторської документації;
- **аудити з кібербезпеки** – артефакти які охоплюють в собі технології та умови експлуатації програмного забезпечення, щоб попередити ризики втручання третіх осіб в інформаційну систему, які можуть нашкодити їй.

Після уточнення поточних документів проекту продемонстровано паспорт проекту який засвідчує формування та початок робіт на проєкті таблиця 1.3.

Таблиця 1.3

Паспорт проєкту

1	2
Назва проєкту	«B2B проєкт з розробки інформаційної системи електронної комерції для підприємців Amazon»
Керівник проєкту (Project Manager)	Гаврилюк Владислав Якович Email: Gavryliuk_vlad@knu.ua Phone: +38(093)22-88-070
Замовник проєкту (Business Owner)	John Doe Email: John_Doe@gmail.com Phone: + 1(415)**-**-***
Цілі проєкту	<ul style="list-style-type: none"> ● Створити вебплатформу SaaS типу яка виконує аналітику та автоматизацію бізнес-процесів на основі даних з Amazon Seller; ● Розробити документацію для готового продукту; ● Сформувати компетентну команду для розробки проєкту; ● Провести контроль якості готового продукту; ● Отримати знання з розробки e-commerce проєктів та ознайомитись з політикою продажів Amazon; ● Вкластись в час розробки півтора року.
Задачі проєкту	<ul style="list-style-type: none"> ● Провести бізнес-аналітику ключових індикаторів бізнесу; ● Покрити потреби користувачів у необхідній аналітиці з продажів на платформі Amazon; ● Виділити сервер створити БД та бекенд для кінцевого продукту;

1	2
	<ul style="list-style-type: none"> ● Розробити та зверстати зручний користувацький інтерфейс; ● Інтегрувати функціонал з логістики товарів, типів їх переміщення та зберігання; ● Створити зручні маркетингові інструменти для відстежування подій з товарами; ● Створити документацію за стандартами AgilePM та ISO21500.
Офіс проєкту, зайнятість	Відсутній (Аутсорсинг)
Обладнання проєкту	<ul style="list-style-type: none"> ● Орендований сервер; ● Персональні комп'ютери співробітників; ● Закупка ліцензованого ПЗ та підписки на сервіси.
Дата початку проєкту	27.03.2023
Дата закінчення проєкту	27.09.2024
Орієнтовний бюджет	282 тис. доларів.

Підсумовуючи поданий розділ можна зрозуміти наскільки подані документи та артефакти є важливими для проєктної діяльності. В свою чергу вони забезпечують організацію інформації на проєкті, надають основну інформацію та канали комунікації з діючими особами. Документація дозволяє виконати контроль та моніторинг великих обсягів робіт, а також звітувати про виконані роботи.

1.3 Вибір стандартів та технологій менеджменту

Для розробки ресурсу який допомагає користувачам автоматизувати бізнес-процеси було обрано ітеративну технологію Scrum для розробки проєкту під

формат Time & Materials. Саме такий варіант був найбільш зручним для обох сторін, оскільки замовник міг достатньо гнучко керувати зміною вимог до продукту проєктної діяльності.

Вибір технології управління є необхідним етапом для кожного проєкту, оскільки вона допомагає менеджеру ефективно керувати ресурсами, такими як час виконання, фінанси, робочі та матеріальні ресурси. У нашому випадку найбільш доцільним є гнучкий підхід Agile, зокрема Scrum підхід. Scrum - це гнучка технологія управління проєктами, яка дозволяє командам структурувати та ефективно керувати роботою на основі цінностей, принципів та практик.

Низка ключових елементів Scrum включає щоденні зустрічі, спринти, планування спринту, беклог, огляд спринту та ретроспективу. Щоденні зустрічі спрямовані на оновлення інформації про прогрес та вирішення проблем, спринти визначають період часу для виконання конкретної роботи, а планування спринту визначає завдання та оцінки для цього періоду [2].

Беклог являє собою список всіх завдань, розподілених за пріоритетом, щоб команда фокусувалася на найважливіших. Огляд спринту дозволяє представити виконану роботу, а ретроспектива спрямована на аналіз минулих спринтів та вдосконалення робочих процесів.

Хоча методика Scrum часто використовується в розробці програмного забезпечення, принципи її можна успішно застосувати в будь-якому типі командної роботи. Отже, технологія використовується для всього циклу розробки продукту, починаючи від валідації ідей та прототипування майбутнього дизайну, закінчуючи розробкою, тестуванням програмного забезпечення перед імплементацією на робочий сервер.

1.4 Оцінка інвестиційної привабливості проєкту

Інвестиційна привабливість проєкту визначається на основі розрахунків, прогнозування прибутків та витрат проєкту. На основі попередніх розрахунків проєкт

вартує приблизно 282 394 доларів США, в які входить 269 600\$ витрат на заробітну плату співробітників та 12 794\$ на загальні фіксовані витрати у які входить оренда хостингу, резервація домену, ліцензійні підписки на ПЗ для розробки й дизайну.

В умовах конвертації USD до UAH за курсом 1 USD = 39,3 UAH на гривні ми отримуємо приблизну вартість 11 110 594 UAH. Розбивши суму на 2 роки розробки ми отримуємо наступні показники. На перший рік припадає 55% загальних витрат проекту 6 110 826 грн, на другий рік 45% аналогічно, що еквівалентна 4 999 767 грн.

Дисконтна ставка - це процентна ставка, за допомогою якої обчислюється сучасна вартість майбутніх грошових потоків або сума, за яку можна дисконтувати майбутні гроші, щоб визначити їхню сучасну вартість. Дисконтна ставка використовується в фінансовому аналізі та прогнозуванні інвестицій. При поточній дисконтній ставці у 20% за даними Національного Банку України, вираховані коефіцієнти дисконтування на 7 років виглядають наступним чином (рис. 1.3). Розрахунок коефіцієнтів дисконтування продемонстрований на у *формулі (1.1)*.

$$PV = \frac{FV}{(1+i)^n}$$

(1.1)

Де:

FV — майбутня вартість;

PV — теперішня вартість;

i — ставка дисконтування (середня річна облікова ставка рефінансування НБУ);

n — строк (число періодів) — часовий період існування заборгованості та відповідних їй грошових потоків.

Дисконтна ставка		20%
1	[Коефіцієнт дисконтування]	83,33%
2	[Коефіцієнт дисконтування]	69,44%
3	[Коефіцієнт дисконтування]	57,87%
4	[Коефіцієнт дисконтування]	48,23%
5	[Коефіцієнт дисконтування]	40,19%
6	[Коефіцієнт дисконтування]	33,49%
7	[Коефіцієнт дисконтування]	27,91%

Рис. 1.3. Коефіцієнти дисконтування на 7 років

Сукупна ймовірність розраховується як добуток попередньо пройдених ймовірностей.

Фінансовий результат розраховується, як сума всіх витрат та доходів при певному сценарії (позитивному, негативному чи нейтральному).

Ймовірність фінансового результату - це добуток сукупної ймовірності та фінансового результату.

Чиста теперішня вартість проєкту (net present value) інвестиційного проєкту - це різниця між сумою теперішніх вартостей грошових потоків від інвестицій в кожен період часу і теперішньою вартістю самих інвестицій [3, ст. 10]. NPV дозволяє визначити, яка чиста прибутковість або втрата буде з інвестиції, беручи до уваги часову вартість грошей *формула (1.2)*.

$$NPV = \sum_{t=1}^n \frac{R_t}{(1+i)^t} - C_0 \quad (1.2)$$

Де:

NPV - чиста поточна вартість;

R_t - чистий дохід у період (t);

i - ставка дисконтування;

t - номер періоду (рік),

n - загальна кількість періодів,

C_0 - початкова інвестиція.

Ймовірність NPV визначається добутком сукупної ймовірності та NPV. На основі поданої інформації та формул розраховано матрицю яка відображає інвестиційну привабливість проекту (рис. 1.4).

e-commerce проект з аналітики продажів												
1 рік проекту	2 рік проекту	1 рік продукту	2 рік продукту	3 рік продукту	4 рік продукту	5 рік продукту	Сукупна ймовірність	Фінансовий результат	Ймовірність фінансового результату	NPV	Ймовірність NPV	
		0,6	5 901	8 262	9 442	11 803	10 662	45,00%	34960,00	15732,00	9635,40	4335,93
	-5 555	0,25	3 540	3 540	2 950	1 770	2 360	18,75%	3050,00	571,88	-2294,08	-430,14
	0,75	0,15	1 180	1 180	885	590	590	11,25%	-6685,00	-752,06	-6516,97	-733,16
	-5 555											
	0,25											
							25,00%	-5555,00	-1388,75		-4629,17	-1157,29
							100,00%	25770,00	14163,06		-3804,81	2015,34

Рис. 1.4. Розрахунки інвестиційної привабливості проекту

Отже, враховуючи сферу діяльності проекту, майбутній функціонал програмного продукту, перспективність технологій та попиту серед користувачів в умовах виходу на ринок конкурентів інвестиційні розрахунки проекту мають відносно достойні показники повернення інвестицій. NPV показник в 9 559 010 грн у випадку успішного завершення проекту та формування доходів за наступні 5 років після завершення проекту демонструє привабливий рівень доходу для інвесторів проекту.

1.5 Концептуальна та математична модель проекту

Концептуальна модель проекту чудовий метод демонстрації загальної картини впливів та процесів. За допомогою каналів входу і виходу, зв'язків елементів системи, впливу внутрішнього середовища та надсистеми можна відтворити візуалізацію математичної моделі проекту (рис. 1.5).

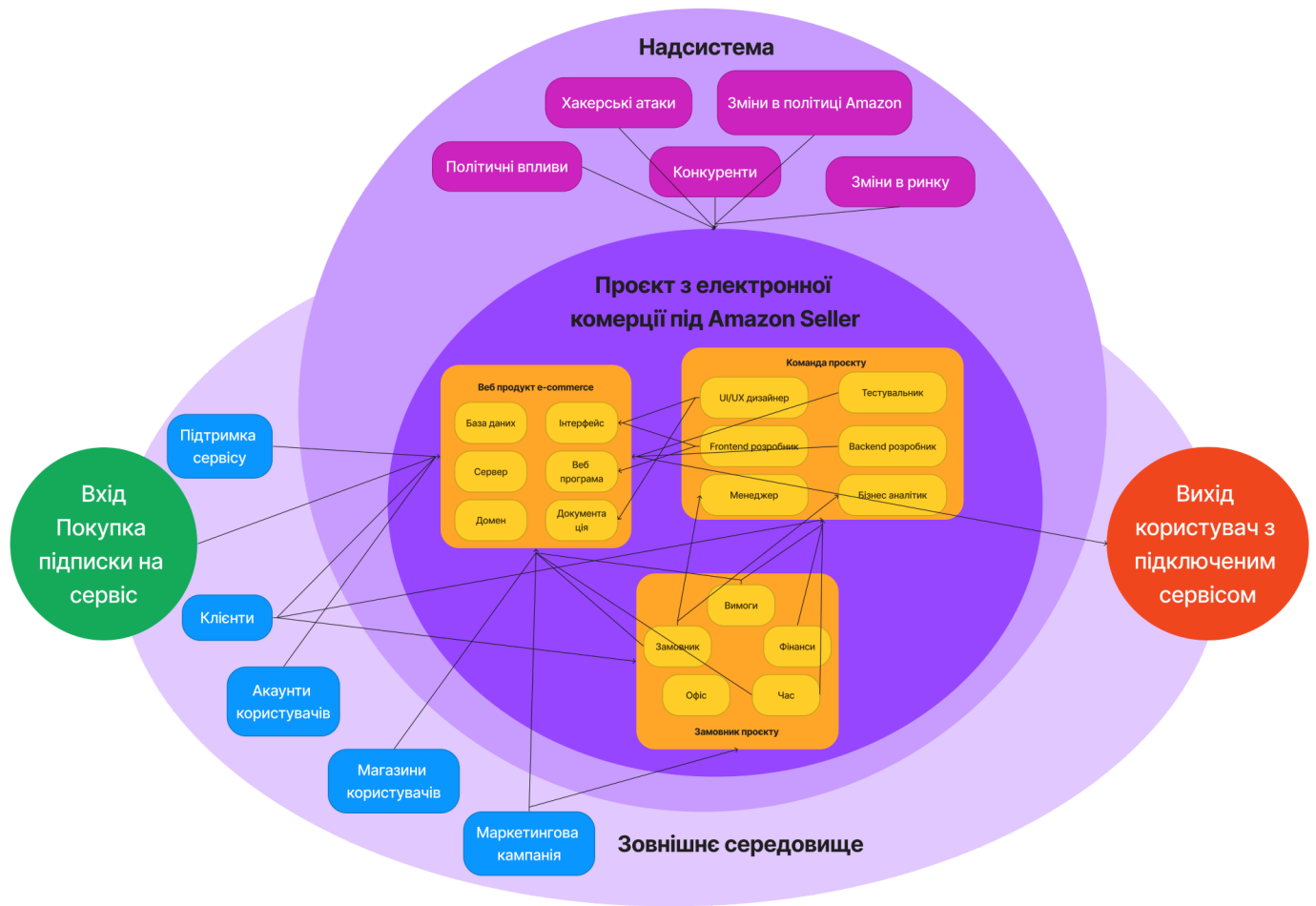


Рис. 1.5. Концептуальна модель e-commerce проекту

Як можна побачити на моделі, то ми маємо початкову точку входу (покупку підписки на сервіс) з якої починаються всі первинні процеси на проекті. Клієнти на основі маркетингової кампанії знаходять наш сервіс і створюють акаунти на яких купують підписку на сервіс та підключають свої регіональні магазини Amazon. Протягом життєвого циклу продукту клієнти підтримують фінансами сервіс, також вони комунікують з командою підтримки програмного забезпечення, щоб вирішувати персональні проблеми та випадки які трапляються у них на шляху використання. Так працює зовнішнє середовище проекту. На основі інформації від користувачів робляться функціональні зміни в проекті, щоб задовольняти потреби та максимально довго утримувати клієнтів. Всередині проект умовно поділяється на три ресурси які в свою чергу мають ще менші елементи впливаючі на розвиток проекту.

В перший елемент входять ресурси замовника такі як: офіс замовника, фінанси, час, сформовані вимоги до вебдодатка та його безпосередній вклад в проєкт. На основі поданих ресурсів збирається команда проєкту яка розробляє та обслуговує ПЗ, що і являє собою другий елемент проєкту. Третій елемент проєкту уособлює в собі програмний продукт в який входить: БД, користувацький інтерфейс, сервер, домен, документація, код вебдодатка який є проміжною ланкою у зв'язку між зовнішнім середовищем та внутрішнім середовищем проєкту.

Варто усвідомлювати, що у концептуальній моделі завжди є надсистема яка майже не підвладна впливу ні зовнішнього, ні внутрішнього середовища проєкту, але з нею можна взаємодіяти аби регулювати політику та процеси на проєкті аби уникати негативного впливу ззовні. Тобто команда проєкту має враховувати ризики змін у законодавстві країн, реагувати на оновлення в політиці користування платформою Amazon, моніторити зміни на ринку рішень аналогів та конкурентів загалом і не менш важливо виконувати заходи з кібербезпеки ПЗ.

На виході процесів взаємодії вхідних даних зовнішнього середовища та вузлів внутрішнього середовища і впливу надсистеми ми маємо задоволеного користувача який продовжуватиме користуватись послугами нашого вебдодатка, що економить йому велику кількість часу та автоматизує ведення його бізнесу.

Аби запевнитись, що концептуальна модель проєкту функціонуватиме вчасно та без проблем приносячи очікувані результати було розроблено математичну формулу, яка дозволяє визначити ефективність та успішність проєкту формула (1.3).

$$PE = PQ * CS * \left(\frac{AC}{PV}\right) * \left(\frac{PT}{ET}\right)$$

(1.3)

Де:

- PE (Project Efficiency) – ефективність проєкту значення в ідеальному випадку якого має PE = 1, що демонструє про успішність аспектів проєкту, якщо PE > 1,

то проєкт виконується нижче запланованого бюджету та часу, а якщо нижче $P < 1$, то у проєкті є наявні проблеми з ефективністю часу та бюджету, а також якості продукту і задоволеності клієнтів;

- PQ (Product Quality) – якість продукту, якого дорівнює 0 або 1;
- CS (Customer Satisfaction) – показник задоволеності клієнтів який встановлюється 1 або 0;
- AC (Actual Cost) – показник фактичної вартості виконаних робіт;
- EV (Earned Value) – вартість фактично виконаної роботи;
- PT (Project Time) – показник вираження використаної кількості місяців проєкту;
- ET (Estimated Time) – відображає орієнтовний час виконання проєкту в місяцях.

В нашому випадку розрахунок $PE = 1$ становить при використаних показниках: $PQ = 1$, $CS = 1$, $AC = 269\ 600\$$, $EV = 269\ 600\$$, $PT = 19$ та $ET = 19$. Отже, це демонструє, високий рівень ефективності та успішності проєкту, що задовільняє користувачів функціоналом та якістю продукту, а також звітує про вчасне виконання проєкту й в оцінену орієнтовну вартість робіт.

На основі ретельного аналізу показників ефективності та досліджень, проведених у цьому розділі, можна усвідомити наскільки математичні закономірності можуть допомогти визначити успішність проєкту за багатокритеріальними аспектами: якості продукту, задоволеності клієнтів, затраченої вартості та часу.

У дослідженні ми ретельно розглянули та проаналізували математичну модель успіху проєкту. А також концептуальну модель які використовуються для опису процесу. В ході роботи було виявлено ключові параметри, які впливають на результати проєкту та розроблено математичне рівняння для розрахунків та оцінювання ефективності. Крім того, ми розробили концептуальну модель, яка відображає загальну ідею проєкту та бізнес модель. Ця концептуальна модель

служить важливим керівним документом для всіх учасників процесу та дозволяє краще зрозуміти сутність та напрямок розвитку проекту.

РОЗДІЛ 2. ПЛАНУВАННЯ ПРОЄКТУ

2.1 Створення життєвого циклу проєкту

Життєвий цикл проєкту ключовий елемент при створенні проєкту, він дозволяє розбивати та структурувати та координувати загальні роботи для команди на всіх етапах. З урахуванням практик інституту PMI та сучасних життєвий цикл ІТ-проєкту SDLC включає 4 етапи: ініціацію, планування, виконання, закриття [5]. У випадку поточного проєкту, додатково виділено п'ятий етап моніторингу, який дуже тісно пов'язаний з етапом виконання на основі ітерацій scrum фреймворку.

Ініціювання – на цьому етапі визначаються концепція проєкту, проводиться аналіз ринку і пошук конкурентів, починається формування кандидатів для команди проєкту, оцінюється можливість реалізації проєкту і підготовлюються вимоги до проєкту.

Планування – на наступному етапі детально розробляється план робіт, визначаються терміни розробка календарного плану та основних віх проєкту, визначаються робочі ресурси проєкту, їх вартість та структура декомпозиції робіт. Не менш важливим постає урахування ризиків проєкту.

Виконання – етап проєкту коли команда реалізує план, займається пошуком компетентних кандидатів, виконує розробку програмного продукту за ітераціями спринтів. Проводить аналіз продуктивності виконання робіт та корегуються процеси роботи.

Контроль – на цьому етапі проходить контроль за виконанням проєкту, оцінюються результати роботи команди проєкту, відбувається перевірка якості програмного продукту на наявність функціональних проблем та помилок.

Завершення – останній крок при якому передбачається підготовка проєкту до релізу на зовнішній ринок, відбувається передача розробленої документації замовнику, а також дані доступу до продукту, наприклад паролі від акаунтів. Попередньо проводиться оцінка результатів проєкту, аналіз помилок і

відпрацьовуються рекомендації для майбутніх проєктів. Загальний вигляд фаз життєвого циклу цілей, задач та основних робіт можна побачити в таблиці 2.1.

Таблиця 2.1

Опис життєвого циклу проєкту

Фаза життєвого циклу	Ціль і задачі	Основні роботи
1	2	3
Ініціація	<ul style="list-style-type: none"> ● Вивчення оточення проєкту (ринку конкурентів, ринку споживачів); ● складання статуту проєкту; ● створення паспорта проєкту; ● формування дерева цілей; ● формування особливостей проєкту; ● створення бізнес-кейсу; ● формування користувацьких сценаріїв використання ПЗ; ● підготовка до формування команди розробки проєкту; ● аналіз початкових вимог до фінансування; 	<ul style="list-style-type: none"> ● Виконати зустріч з замовником і стейкхолдерами для аналізу оточення проєкту; ● задокументувати та підписати статут проєкту всіма сторонами які затвердять існування початку проєкту; ● сформувати детальне дерево цілей поточного проєкту з електронної комерції, проаналізувавши його сильні та слабкі сторони; ● розробити на основі знайдених даних та виконати аналіз таблиці потенційних конкурентів проєкту;

Продовження табл. 2.1

1	2	3
	<ul style="list-style-type: none"> ● створення графіку віх проекту та дорожньої карти; ● складання реєстру та матриці зацікавлених сторін проекту; ● пошук потенційних інвесторів проекту. 	<ul style="list-style-type: none"> ● спланувати сценарії використання які допоможуть розв'язати головні проблеми користувачів, обговорити їх з замовником проекту; ● створити та затвердити графік віх відштовхуючись від ключових дат та орієнтовного періоду виконання робіт та MVP; ● на основі вивчення оточення проекту сформувати реєстр зацікавлених сторін проекту включаючи: інвесторів, партнерів, клієнтів, конкурентів, команди розробки та проекту, додатково внести в список їх персональні дані; ● розпочати пошук запитів у інтернеті, створити сторінку стартапу, пошук ангелів стартапу.

Продовження табл. 2.1

1	2	3
Планування	<ul style="list-style-type: none"> • Визначення обсягу робіт, розробка та затвердження ТЗ із командою замовника; • визначення цілей та результатів проекту; • розробка календарного плану проекту; • створення структури розбивки робіт (WBS); • розробка плану управління ризиками; • складання бюджету проекту; • визначення ролей та обов'язків у проекті. 	<ul style="list-style-type: none"> • Створити та затвердити документ з ТЗ продукту, визначити цілі продукту, а також, порядок контролю виконання робіт, їх вартість, терміни тестування та здачі проекту. • сформулювати чітко визначені та досяжні цілі проекту на основі технології SMART; • розробити детальний календарний план і технічне завдання; • описати склад робіт для членів команди проекту розробити WBS та запланувати спринти на основі MVP; • зробити оцінку вартості кожного завдання; • визначити контрольні точки для оцінки реалізації проекту; • розробити документацію з ідентифікування, оцінювання ризиків, а також стратегією управління ними; • розрахувати орієнтовну вартість проекту.

Продовження табл. 2.1

1	2	3
Виконання	<ul style="list-style-type: none"> • Визначення ролей та обов'язків; • підготувати робоче ПЗ для роботи над проектом; • планування ітераційних спринтів для команди проекту; • щоденні зустрічі з командою для визначення обсягу робіт на робочий день; • ретроспективи ітерацій спринтів; • управління ресурсами проекту; • створення продукту та корегування пріоритетів робіт. 	<ul style="list-style-type: none"> • Визначити командного лідера, який буде відповідальний за створення ТЗ та розробку ПЗ; • розподілити обов'язки між членами команди та закріпити за кожним членом наявні частини обладнання; • встановити необхідне ПЗ для створення та тестування додатка; • назначити менеджера проекту відповідальним за бюджет, планування ітерацій робіт, звітування стейкхолдерам, ретроспективи та щоденні зустрічі; • створити дизайн інтерфейсу застосунку на основі досліджених даних; • зверстати FrontEnd застосунку за поданим ТЗ; • створити необхідний Backend функціонал включно з БД; • провести QA тестування під час закінчення певного етапу створення ПЗ.

Продовження табл. 2.1

1	2	3
Контроль	<ul style="list-style-type: none"> • Створення документа з ітераціями робіт та дедлайнами; • відстеження ресурсів та коштів; • забезпечення дотримання календарного плану; • імплементація протиризикових заходів; • провести QA тестування ПЗ та дизайну інтерфейсів на основі ітерацій. 	<ul style="list-style-type: none"> • Створити графік виконання робіт з поставленими задачами та дедлайнами; • відстежити дедлайни в кінці кожного спринту виконуючи відмітки в календарному плані, визначити проблемні роботи та відзвітувати про витрачений час та кошти; • виконати перерозподіл пріоритету задач для наступного спринту запобігти простою проекту; • виконати QA тестування ПЗ, відзвітувати команді про кількість програмних і візуальних багів виписати задачі та виставити пріоритети їх вирішення.
Завершення	<ul style="list-style-type: none"> • Перегляд результатів проекту; • передача результатів проекту; • отримання затвердження результатів проекту; • документування результатів проекту. 	<ul style="list-style-type: none"> • Зібрати всю команду проекту виконати передрелізне тестування продукту; • продемонструвати команді замовника та інвесторам готовий продукт;

Продовження табл. 2.1

1	2	3
		<ul style="list-style-type: none"> • виконати очищення зайвих даних, тестових директорій, відключити щомісячні підписки на ліцензійне ПО, перенести програмний продукт на фіктивний домен. • підготувати список документів до результату затвердження обох сторін; • внести останні корективи в технічну документацію продукту та передати технічну документацію з описом проекту та принципами роботи ПЗ.

2.2 Побудова структури проекту та проєктної команди

Процес побудови структури проекту починається з пошуку та відбору кандидатів для формування проєктної команди в умовах потреб та вимог проекту. Зазвичай в аутсорсингових компаніях є достатня кількість спеціалістів, які можуть покрити весь необхідний об'єм технологічного стеку визначений технічним лідером та проєктним менеджером. Але дуже часто доводиться займатись пошуком відповідних кандидатів на ринку спеціалістів для того, щоб бути повною мірою впевненим в тому, що у майбутнього члена команди не буде пробілів в знаннях технологій. Які в свою чергу можуть призвести до ризиків втрати часу на опанування нових технологій.

Отже, таким чином варто відбирати нових членів команди на основі, технічних навичок (hard skills) та комунікаційних і психологічних навичок (soft skills), а також

враховувати обов'язки та компетенції кожного зі спеціалістів. На основі перерахованої інформації сформована таблиця 2.2, яка демонструє вимоги та компетенції до співробітників проєктів за їх відповідними ролями.

Таблиця 2.2

Вимоги та компетенції проєктної команди

Спеціаліст та роль	Обов'язки та компетенції	Технічні та комунікаційні навички
1	2	3
Backend розробник (техлід)	<ul style="list-style-type: none"> ● Обслуговування серверної мережевої частини; ● проектування та обслуговування БД; ● розробка архітектури та ПЗ для клієнт-серверної частини додатка; ● постійна комунікація з командою проєкту, та навички технічного лідера; ● комунікація з сапортами програмних рішень та інтеграцій; ● аналіз системи та перевірка вразливостей системи до зовнішніх атак. 	<ul style="list-style-type: none"> ● Володіння високорівневими мовами програмування PHP та Python, опціонально Node.js, Java, C# буде плюсом; ● знання фреймворків Laravel, Symphony, Django, Flask та API HTTP, Rest, JSON; ● навички роботи з реляційними БД та СУБД MySQL, PostgreSQL і запитами; ● знання мережевих протоколів OSI моделі; ● досвід з системами контролю версій ПЗ (GIT); ● досвід роботи 5 років у сфері та знання англійської від B2.

Продовження табл. 2.2

1	2	3
Frontend розробник	<ul style="list-style-type: none"> • Верстання дизайну вебсайту; • розробка та підтримка клієнтського інтерфейсу вебзастосунка; • комунікація в команді та з представниками інших компаній; • оптимізація структури коду та даних; • базова робота з SEO оптимізацією вебдодатка. 	<ul style="list-style-type: none"> • Знання технологічного стеку та мов: JavaScript, HTML, CSS, JSON; • досвід розробки на фреймворках Angular, React.js, Vue.js; • взаємодія з API: DOM, Fetch, Websocket, Web Storage, GraphQL; • знання системи контролю версій (GIT) та Devtools; • мінімальний комерційний досвід 3 роки у сфері та знання англійської від B2.
Бізнес-аналітик	<ul style="list-style-type: none"> • Постановка вимог до проектного продукту, валідація робіт; • комунікація зі стейкхолдерами та командою проекту; • створення документації з бізнес-аналітики; • ілюстрація бізнес діаграм; • генерація ідей та технічних рішень для ітерацій продукту; • аналіз конкурентів; • робота з автоматизацією даних на проєкті. 	<ul style="list-style-type: none"> • Технічний досвід 4 роки на основі проєктів в сфері електронної комерції та фінансів; • навички роботи з вебаналітикою Google tag manager, Google Analytics 4 та Universal Analytics; • базові знання Python та R для структурування та проведення кореляцій даних; • робота з системами візуалізації даних Power BI та Tableau;

Продовження табл. 2.2

1	2	3
		<ul style="list-style-type: none"> ● навички прототипування інтерфейсів та вміння побудов бізнес діаграм BPMN, UML в Figma та Miro; ● розмовна та письмова англійська C1.
UI/UX дизайнер	<ul style="list-style-type: none"> ● Створення дизайн системи; ● комунікація зі стейкхолдерами та командою; ● обробка растрових та векторних графічних об'єктів; ● генерація ідей та створення прототипів інтерфейсів; ● робота з анімацією інтерфейсів; ● створення інформаційної архітектури та навігації продукту; ● розробка інтелектуальних карт; ● створення екранів дизайну інтерфейсу додатка. 	<ul style="list-style-type: none"> ● Знання сучасних дизайн систем Material Design, Apple HIG; ● базові знання Frontend мов JS, CSS, HTML, а також БД структури та запитів; ● знання інструментів розробника; ● робота з програмами для створення інтерфейсів та дизайну: Figma, InVision, Adobe XD, Sketch, Miro; ● професійний рівень володіння та роботи з графічними редакторами: Adobe Photoshop та Illustrator; ● комерційний досвід роботи 3 роки у сфері дизайну інтерфейсів та знанням англійської на рівні B2.

Продовження табл. 2.2

1	2	3
QA-тестувальник	<ul style="list-style-type: none"> ● Проведення планових релізів ПЗ; ● розробка стратегії управління якістю продукту; ● створення тест кейсів; ● аналіз результатів тестування; ● написання автоматизованих скриптів тестування; ● документування та звітування робіт. 	<ul style="list-style-type: none"> ● Знання сучасних методів тестування та використання протягом 3 років; ● аналітичні здібності; ● досвід написання технічної документації; ● знання високорівневих веб мов програмування JS, PHP та теорії алгоритмів; ● робота з інструментами: GIT, Selenium Web Driver, Protractor, Cypress, Jest, Postman; ● Знання англійської B2.

2.2.1 Візуалізація структури проєкту

Управління проєктом припускає його розбивку на окремі блоки, які є самостійними об'єктами планування, обліку, організації й координування, тобто побудову структури проєкту.

Структура проєкту — це сукупність взаємопов'язаних елементів і процесів проєкту, які представлені з різним ступенем деталізації. В термінах управління проєктами структура проєкту являє собою «дерево» орієнтованих на продукт компонентів, представлених обладнанням, роботами, послугами й інформацією, отриманими в результаті реалізації проєкту [6, ст. 1].

Структура проєкту покладає у свою основу комунікаційну ієрархію команди проєкту організаційну та функціональну структуру водночас. Організаційна структура проєкту встановлює ієрархію та ланцюг управління всередині проєктної команди. Загальною практикою є використання однієї з таких організаційних структур в проєктах:

- Функціональна – цей вид структури базується на функціях (наприклад, маркетинг, розробка, тестування тощо). Проектна команда формується з представників різних функціональних підрозділів, які мають відповідні навички та експертизу. В цьому випадку учасникам проектної діяльності призначаються їх ролі та які несуть за собою певну низку обов'язків.
- Проектна – в свою чергу в такій структурі керівництво та ресурси цілком призначені для конкретного проекту, без прив'язки до функціональних підрозділів.
- Матрична – у цій структурі керівництво може бути подвійним функціональним та проектним. Учасники проекту відзначаються як члени функціональних підрозділів та призначені для виконання конкретних завдань в рамках проекту.
- Лінійна – типова структура яка містить в собі ієрархію, де кожен учасник проекту звітує лише одному керівникові. Це дозволяє забезпечити чіткість та однозначність командних відносин. В лінійній структурі проекту керівництво зазвичай виконується однією особою або групою передвизначених осіб, які здійснюють контроль над усіма аспектами проекту. В такому варіанті існує як позитивний ефект збігання плутанини, так і негативний ефект обмеженості комунікації.
- Дивізіональна - це організаційна модель, в основі якої компанія поділяється на різні дивізії або сегменти за видами продукції, ринками або географічними областями. Кожна дивізія має своє власне керівництво та внутрішню структуру, яка дозволяє їй функціонувати як самостійній одиниці.

Отже, в нашому випадку максимально підходить обране компанією рішення матричної структури (рис. 2.1) яка несе в собі позитивні риси функціональної та проектної структур.

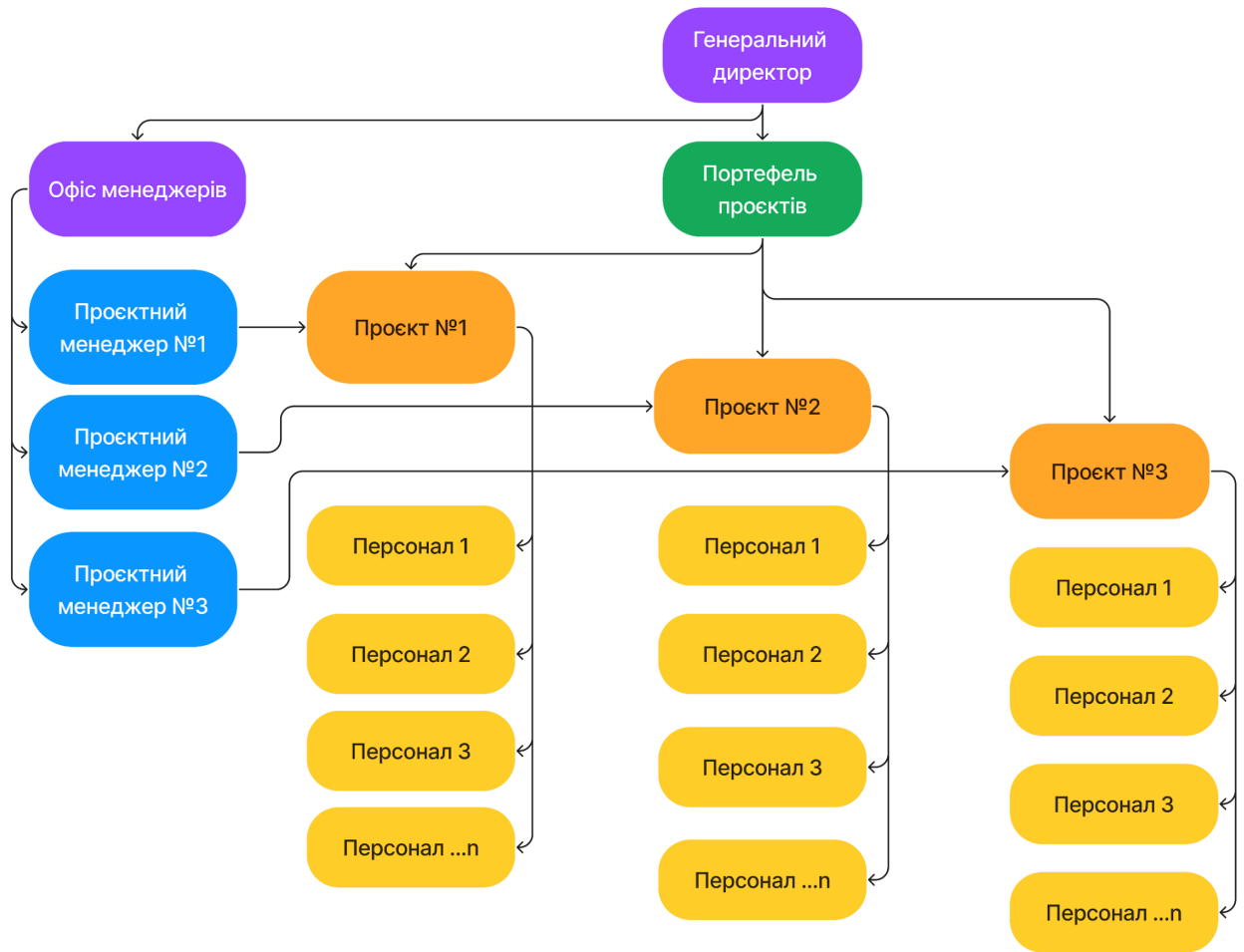


Рис. 2.1. Матрична структура підприємства

З попередньої матричної структури підприємства можна побачити детальний вигляд поточної структури проєкту включаючи дієвих представників команди проєкту. Візуалізація структури проєкту виконана на (рис. 2.2) і має наступну передвизначену структуру комунікації між членами команди, проєктним менеджером та замовником.

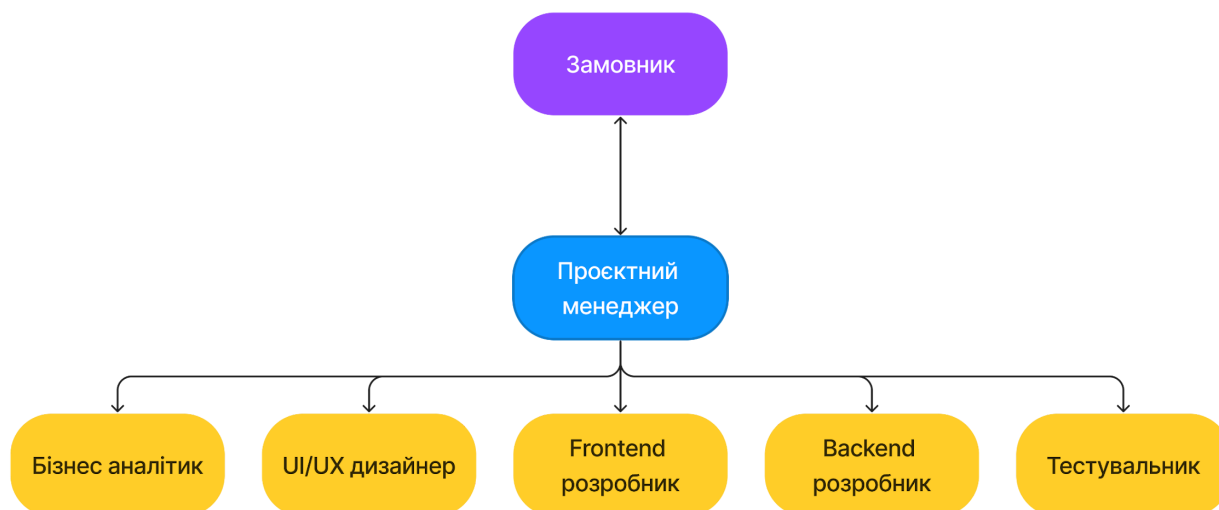


Рис. 2.2. Структура проєкту, команди та комунікація між ланками

Отже, ключовим елементом в проєкті, який визначає організацію та функціонування проєктної команди є організаційна структура. Її чітка визначеність сприяє ефективності та успішному виконанню поставлених завдань для учасників залучених до розробки проєктної платформи з електронної комерції.

2.2.2 Розподіл компетентностей, ролей та зон відповідальності

Відповідно до процесів виконання сучасних проєктів, які стають все більш складними виявляється потреба у розподілі ролей та компетентностей в корпоративному середовищі. Такий розподіл надає можливість чітко окреслити зони компетентностей співробітників та об'єми робіт з функціональними обов'язками кожного з членів команди. Відштовхуючись від потреб в команді проєкту розробки платформи з електронної комерції є наступні важливі ролі:

Backend розробник (Tech Lead та Software Architect) – особа яка одноосібновідповідає за написання клієнт-серверного коду, розробку архітектури програмного забезпечення налаштування БД. Також має компетенції перевірки якості

написання коду іншим розробником, для процесів покращення ефективності роботи інформаційної системи.

Frontend розробник – член команди який одноосібно займається написанням коду для взаємодії візуальної частини додатку з користувачем. А також версткою підготовлених дизайнером макетів інтерфейсів.

Проектний менеджер (Scrum master) – ключова особа яка займається управлінням часовими рамками, трудовими та матеріальними ресурсами проекту та одноосібно несе відповідальність за успішність виконання проекту. Додатково роль вимагає постійного керування ризиками та ключовими показниками ефективності.

Бізнес-аналітик – ключова особа відповідальна за апробування, формування системних вимог у проекті задля досягнення задоволення клієнта та майбутніх користувачів продукту діяльності. Роль передбачує контроль фінансових ресурсів та КРІ, вебаналітики, збору документації та аналіз ринку конкурентів. Займається пошуком представлення даних у продукті й формує документацію.

UI/UX дизайнер – роль яка передбачає створення креативних матеріалів для проекту в цифровому та друкованому форматі. Основна задача створення дизайну для інтерфейсу ПЗ за сучасними вимогами та найкращими практиками індустрії.

QA-тестувальник – особа яка відтворює послідовність дій з тестування додатку. Створює стратегії управління якістю ПЗ, а також документує віднайдені проблеми в системі.

Власник продукту (замовник) – це ключова роль у методології розробки програмного забезпечення Scrum. Головна відповідальність власника продукту полягає у максимізації вартості продукту, розробленого командою розробників. Основні функції продукт овнера включають: визначення вимог, пріорітезація завдань, комунікація з розробниками через проектного менеджера, контролем беклогу задач. Ключовою особливістю ролі продукт овнера є відповідальність за максимізацію цінності продукту та забезпечення ефективного використання часу та ресурсів розробки.

Чудовим інструментом для демонстрації відповідальностей для зменшення конфліктних ситуацій, невизначеностей на проєкті та підвищення ефективності та навчальної дисципліни є RACI матриця. При заповненні матриці RACI важливо переконатися, що всі зацікавлені сторони або групи зацікавлених сторін були визначені. Потім проводиться подальший аналіз для присвоєння RACI з метою визначення рівня відповідальності, що очікується від кожної зацікавленої сторони та/або групи [7]. В матрицю входять наступні маркери: Responsible (відповідальний), Accountable (зобов'язаний), Consulted (консультований) та Informed (інформований). Відповідно до вище перерахованих позитивних характеристик, RACI матриця проілюстрована на (рис. 2.3).

Ролі / Завдання	Проектний менеджер	Frontend Розробник	Backend Розробник	Бізнес Аналітик	QA-тестувальник	Продакт Овнер
Планування проєкту	A	I	I	C	I	R
Аналіз вимог	C	I	C	A	C	R
Розробка фронтенду	I	R	C	I	C	C
Розробка бекенду та БД	I	C	R	I	C	C
Створення дизайну інтерфейсу	I	C	C	C	C	C
Управління продуктом	A	I	I	A	C	R
Управління командою та ресурсами	R	I	C	C	I	R
Управління ризиками	A	I	C	A	A	R
Тестування якості продукту	C	A	A	I	R	C

R Responsible
 A Accountable
 C Consulted
 I Informed

Рис. 2.3. демонстрація RACI матриці

2.3 Формування бюджету та оцінювання ресурсів

На сьогоднішній день вартість проекту стоїть достатньо важливим питанням для усіх замовників. Побаження замовника постійно зводиться до виконання якісного програмного продукту за низьку ціну відносно ринкової вартості. Саме тому український ІТ-ринок привертає увагу більшість замовників.

При оцінці вартості проекту слід враховувати вартість робочих ресурсів, апаратного та програмного забезпечення протягом процесу розробки. Оцінка вартості проекту охоплює часовий розріз витрат на проєкті відносно погодинної ставки робітників. Базуючись на такому підході, що максимально адаптований як для аутсорсингової ІТ-компанії, так і для проекту погляд впав на контракт оплати «Time and materials». Детальні дані оцінки вартості робіт проєкту можна переглянути в (рис. 2.4).

Pages	Business Analysis	UI/UX Designer	Frontend Dev	Backend Dev	QA Testing	PM	Business Analysis	UI/UX Design	Frontend Dev	Backend Dev	QA Testing	PM
Landing	80	64	60	24	56	142	\$1 200,00	\$640,00	\$1 200,00	\$600,00	\$840,00	\$2 840,00
- Login	48	16	30	74	86	142	\$720,00	\$160,00	\$600,00	\$1 850,00	\$1 290,00	\$2 840,00
- Create account	48	16	30	82	32	142	\$720,00	\$160,00	\$600,00	\$2 050,00	\$480,00	\$2 840,00
Account settings	158	80	70	90	180	142	\$2 370,00	\$800,00	\$1 400,00	\$2 250,00	\$2 700,00	\$2 840,00
Home page	98	54	120	110	120	142	\$1 470,00	\$540,00	\$2 400,00	\$2 750,00	\$1 800,00	\$2 840,00
Finances:												
- Dashboards	150	90	140	180	160	142	\$2 250,00	\$900,00	\$2 800,00	\$4 500,00	\$2 400,00	\$2 840,00
- Expences	120	90	120	142	135	142	\$1 800,00	\$900,00	\$2 400,00	\$3 550,00	\$2 025,00	\$2 840,00
- Products	132	80	120	148	147	142	\$1 980,00	\$800,00	\$2 400,00	\$3 700,00	\$2 205,00	\$2 840,00
- Customer retur	150	90	120	187	134	142	\$2 250,00	\$900,00	\$2 400,00	\$4 675,00	\$2 010,00	\$2 840,00
Inventory:												
- Stock	170	72	110	120	120	142	\$2 550,00	\$720,00	\$2 200,00	\$3 000,00	\$1 800,00	\$2 840,00
- Contacts	107	60	70	122	80	142	\$1 605,00	\$600,00	\$1 400,00	\$3 050,00	\$1 200,00	\$2 840,00
- Purchase order	115	72	84	126	100	142	\$1 725,00	\$720,00	\$1 680,00	\$3 150,00	\$1 500,00	\$2 840,00
- Shipments	130	72	146	134	100	142	\$1 950,00	\$720,00	\$2 920,00	\$3 350,00	\$1 500,00	\$2 840,00
- Removal order	140	33	140	125	127	142	\$2 100,00	\$330,00	\$2 800,00	\$3 125,00	\$1 905,00	\$2 840,00
Buyers & Orders:												
- Orders	150	50	140	120	135	142	\$2 250,00	\$500,00	\$2 800,00	\$3 000,00	\$2 025,00	\$2 840,00
- Review reques	120	20	140	120	80	142	\$1 800,00	\$200,00	\$2 800,00	\$3 000,00	\$1 200,00	\$2 840,00
- Feedback	120	20	120	120	64	142	\$1 800,00	\$200,00	\$2 400,00	\$3 000,00	\$960,00	\$2 840,00
Web bot assistar	140	45	160	152	64	146	\$2 100,00	\$450,00	\$3 200,00	\$3 800,00	\$960,00	\$2 920,00
Time reserves f	544	256	480	544	480	640	BA Rate , \$	Design Rate , \$	Frontend Rate , \$	Backend Rate , \$	QA Rate , \$	PM Rate , \$
							\$15,00	\$10,00	\$20,00	\$25,00	\$15,00	\$20,00
Total time & Res	2720	1280	2400	2720	2400	3200	\$8 160,00	\$2 560,00	\$9 600,00	\$13 600,00	\$7 200,00	\$12 800,00
Total:	\$269 600,00						\$40 800,00	\$12 800,00	\$48 000,00	\$68 000,00	\$36 000,00	\$64 000,00

Рис. 2.4. Розрахункова вартість робіт проєкту

Переглядаючи попередній рисунок можна побачити, що розробка продукту та модулів проекту розбита на складові сторінок та загальної інформаційної архітектури проекту. Також на ньому можна побачити ставку кожного співробітника проекту, загальні витрати часу та резерв часу відведений на непередбачувані обставини. Загальна вартість робочих ресурсів дорівнює 269 600\$ що за обмінним курсом 1 USD до 39,3 UAH еквівалентно 10 607 223 UAH.

Наступний (рис. 2.5) демонструє фіксовані витрати на матеріали і ресурси для відтворення продукту включаючи покупку домену, оренду сервера та підписки на ліцензійне програмне забезпечення для співробітників.

Fixed expenses	Per one	Quantity	Cost
Domain	\$5,00	60	\$300,00
Web server rent	\$150,00	60	\$9 000,00
Figma	\$45,00	20	\$900,00
Adobe cloud	\$36,00	20	\$720,00
GITLAB	\$29,00	20	\$580,00
Confluence	\$12,00	20	\$240,00
Jira	\$16,00	20	\$320,00
PHP Storm	\$448,00	1	\$448,00
Web Storm	\$286,00	1	\$286,00
Total:	\$293,00		\$12 794,00

Рис. 2.5. Фіксовані витрати проекту

Підсумовуючи вартість фіксованих витрат за аналогічним обмінним курсом валют 12 794 USD що дорівнює 503 370 UAH. Отже, загальний відведений бюджет становить 11 110 593 UAH чого достатньо для роботи над проектом співробітникам усіх ролей у режимі часткової зайнятості протягом життєвого циклу проекту. У випадку змін до вимог продукту вартість розробки може змінитись в залежності від бажаного кінцевого результату, що буде обговорюватись з замовником та інвесторами окремо.

2.4 Планування віх та WBS проєкту

WBS - це ієрархічна та інкрементна декомпозиція проєкту на фази, результати та пакети робіт. Це деревоподібна структура, яка показує розподіл зусиль, необхідних для досягнення мети, наприклад, програми, проєкту або контракту [9]. WBS має три основні цілі:

- опис розбивки або складу роботи у завданнях;
- планування роботи за проєктом;
- оцінка вартості кожного завдання.

Ступінь деталізації у WBS залежить від рівня точності, необхідного в оцінках, а також рівня відстеження, необхідного для цих оцінок. Проєктам, які мають дуже низький допуск відставання за термінами в розкладі або вартості, зазвичай потрібен більш деталізований WBS, а також ретельне відстеження перебігу робіт і витрат у WBS. Цей вид проєктів поширений в будівництві та інженерній галузі.

Натомість проєкти в таких галузях, як засоби масової інформації та реклама, програмне забезпечення та інфраструктура ІТ, як правило, є єдиними у своєму роді, а продуктивність праці залежить від досвіду та компетенції людини, яка виконує завдання. Тому ці галузі використовують WBS для отримання приблизного розміру проєкту, а не для відстеження перебігу цього проєкту в деталях [8].

Планування робіт проєкту відбувається у форматі підготовки WBS (Work Breakdown Structure), який сформований в свою чергу проєктним менеджером та членами команди. Цей потужний інструмент, дозволяє систематизувати та ієрархізувати всі елементи, необхідні для виконання конкретного проєкту. Цей розділ описує значення WBS та кроки її розробки.

Як можна побачити на (рис. 2.6.) проілюстрована розбивка робіт проекту у форматі WBS яка розділена на 6 гілок. Перша гілка відповідає за роботи з формування команди проекту. Друга гілка виконує розбиття ієрархії робіт відносно задач з бізнес аналітики продукту. Третя гілка створена з орієнтацією на розробку сучасного дизайну інтерфейсу додатка. Четверта орієнтована на розробку програмного забезпечення. П'ята відповідає за тестування програмного продукту. А шоста за управління проектом та його ресурсами.

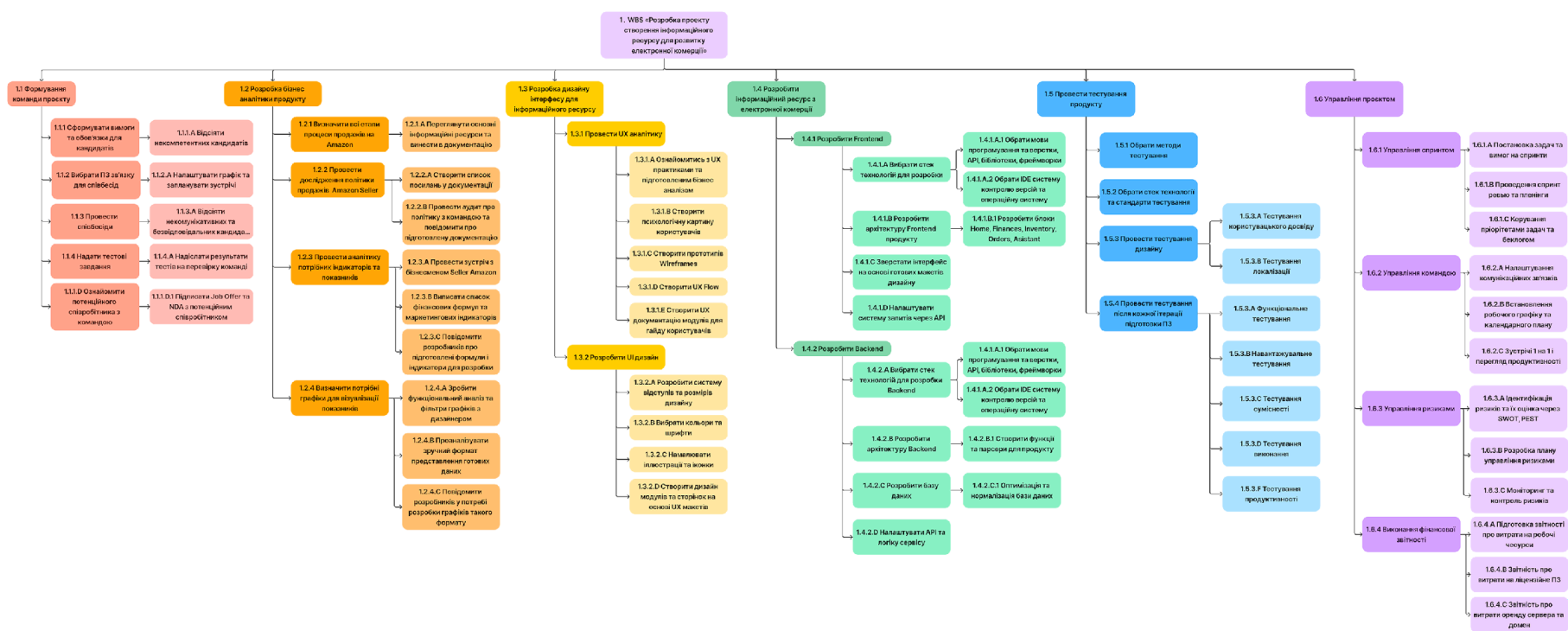


Рис. 2.6. Загальна структура робіт проекту WBS

WBS допомагає усвідомити всі аспекти проекту та організувати їх в логічну структуру. Кожен робоча гілка та завдання може бути привласнена та керована окремо, сприяючи кращому контролю над процесом виконання проекту. WBS дозволяє краще розуміти залежності між завданнями та етапами проекту, що сприяє ефективному плануванню ресурсів та часу. Як можна побачити роботи закодовані в чисельно-буквену форматі.

Розробка проекту не обходиться без створення віх. Віхи проекту визначають ключові події або досягнення, які служать орієнтирами у виконанні проекту. Віхи - це конкретні, вимірювані критичні події або досягнення, які вказують на важливі етапи у виконанні проекту і надають можливість відстежувати прогрес проекту за графіком. Вони допомагають забезпечити фокус команди на ключових моментах та орієнтують на досягнення мети [10, 11]. На основі таких потреб створена таблиця 2.3 яка відображає віхи проекту.

Таблиця 2.3

Критичні віхи проекту

№	Дата віхи	Назва	Відповідальний
1	2	3	4
1	27.03.2023	Прийняття рішення про розробку проекту.	Керівник проекту
2	03.04.2023	Розроблений паспорт проекту.	Керівник проекту
3	05.04.2023	Створений бізнес-кейс.	Бізнес-аналітик
4	06.04.2023	Сформовано дерево цілей.	Керівник проекту
5	07.04.2023	Сформовано вимоги до проекту.	Керівник проекту
6	10.04.2023	Затвердження та завершення етапу ініціювання.	Замовник проекту
7	11.04.2023	Розроблено віхи проекту.	Керівник проекту
8	12.04.2023	Розроблено (WBS).	Бізнес-аналітик

1	2	3	4
9	13.04.2023	Розробка плану управління ризиками.	Керівник проєкту
10	14.04.2023	Формування кошторису проєкту.	Керівник проєкту
11	17.04.2023	Затвердження та завершення етапу планування проєкту.	Замовник проєкту
12	16.05.2023	Проведено бізнес-аналітику формул, індикаторів та політики Amazon.	Бізнес-аналітик
13	29.05.2023	Завершення етапу найму членів проєктної команди.	HR-менеджер
14	21.06.2023	Проведено UX аналітику та спроектовано прототипи дизайну.	UI/UX дизайнер
15	28.07.2023	Створено всі UI екрани дизайну інтерфейсів.	UI/UX дизайнер
16	15.11.2023	Завершення розробки Frontend.	Frontend розробник
17	26.12.2023	Завершення розробки баз даних.	Backend розробник
18	10.05.2024	Завершення Backend розробки логіки та архітектури.	Backend розробник
19	16.07.2024	Завершення написання документації продукту.	Команда проєкту
20	17.07.2024	Завершення етапу виконання.	Замовник проєкту
21	18.07.2024	Контрольна перевірка якості програмного продукту та технічної документації.	QA тестувальник
22	02.08.2024	Проведено перевірку бізнес документації та UX документації.	UI/UX дизайнер
23	05.08.2024	Оцінка фінансування проєкту.	Керівник проєкту
24	06.08.2024	Завершення етапу контролю.	Замовник проєкту
25	14.08.2024	Передача персональних корпоративних даних, паролів, облікових записів замовнику проєкту.	Керівник проєкту
26	22.08.2024	Завершення етапу здачі.	Замовник проєкту

1	2	3	4
27	20.09.2024	Оцінка результатів проєкту.	Керівник проєкту, бізнес-аналітик та замовник проєкту.
28	27.09.2024	Церемонія закінчення проєкту.	Керівник проєкту.

Таким чином, процес деструктуризації робіт та розробка віх проєкту є важливим етапом у плануванні та контролі проєкту. Віхи проєкту забезпечують чітке структурування та відстеження виконання робіт, що дозволяє забезпечити успішну реалізацію проєкту в рамках встановлених термінів та бюджету.

2.5 Календарний план та таймлайни проєкту

Календарне планування включає в себе процес збору інформації про часові рамки виконання проєкту в залежності від годинного розподілу робочого графіку проєкту, завантаженості персоналу, а також вихідних календарних днів та свят. Календарний план надає можливість чітко визначити дні комунікації з клієнтом, а також розподілити завантаженість команди [12, 13].

Попередньо узгодивши умови робіт та обговоривши всі аспекти з замовником, було обговорено та затверджено робочий графік, який дозволить працювати команді проєкту п'ять днів в тиждень з навантаженням по вісім годин. Що сумарно дає завантаженість до 40 годин в тиждень. В обговоренні також згадувались святкові офіційні календарні дні під час яких співробітники також будуть відпочивати. Опираючись на подану інформацію та оцінки кількості робіт по модулях на проєкті було проаналізовано та створено календар за 2023-2024 роки в якому було чітко означено планові робочі та вихідні (субота, неділя) або святкові дні (відмічено жовтим кольором на календарі). Візуально календар робочих днів за часовими рамками проєкту має наступний вигляд (рис. 2.7).

Березень 2023							Квітень 2023							Травень 2023							Червень 2023							Липень 2023							Серпень 2023						
ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД
		1	2	3	4	5					1	2	1	2	3	4	5	6	7				1	2	3	4						1	2			1	2	3	4	5	6
6	7	8	9	10	11	12	3	4	5	6	7	8	9	8	9	10	11	12	13	14	5	6	7	8	9	10	11	3	4	5	6	7	8	9	7	8	9	10	11	12	13
13	14	15	16	17	18	19	10	11	12	13	14	15	16	15	16	17	18	19	20	21	12	13	14	15	16	17	18	10	11	12	13	14	15	16	14	15	16	17	18	19	20
20	21	22	23	24	25	26	17	18	19	20	21	22	23	22	23	24	25	26	27	28	19	20	21	22	23	24	25	17	18	19	20	21	22	23	21	22	23	24	25	26	27
27	28	29	30	31			24	25	26	27	28	29	30	29	30	31				26	27	28	29	30			24	25	26	27	28	29	30	28	29	30	31				

Вересень 2023							Жовтень 2023							Листопад 2023							Грудень 2023							Січень 2024							Лютий 2024													
ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД							
				1	2	3						1			1	2	3	4	5				1	2	3	1	2	3	4	5	6	7			1	2	3	4					1	2	3	4		
4	5	6	7	8	9	10	2	3	4	5	6	7	8	6	7	8	9	10	11	12	4	5	6	7	8	9	10	8	9	10	11	12	13	14	5	6	7	8	9	10	11	5	6	7	8	9	10	11
11	12	13	14	15	16	17	9	10	11	12	13	14	15	13	14	15	16	17	18	19	11	12	13	14	15	16	17	15	16	17	18	19	20	21	12	13	14	15	16	17	18	12	13	14	15	16	17	18
18	19	20	21	22	23	24	16	17	18	19	20	21	22	20	21	22	23	24	25	26	18	19	20	21	22	23	24	22	23	24	25	26	27	28	19	20	21	22	23	24	25	19	20	21	22	23	24	25
25	26	27	28	29	30		23	24	25	26	27	28	29	27	28	29	30			25	26	27	28	29	30	31	29	30	31				26	27	28	29				26	27	28	29					

Березень 2024							Квітень 2024							Травень 2024							Червень 2024							Липень 2024							Серпень 2024													
ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД	ПН	ВТ	СР	ЧТ	ПТ	СБ	НД							
				1	2	3	1	2	3	4	5	6	7			1	2	3	4	5				1	2	3	1	2	3	4	5	6	7			1	2	3	4					1	2	3	4	
4	5	6	7	8	9	10	8	9	10	11	12	13	14	6	7	8	9	10	11	12	3	4	5	6	7	8	9	8	9	10	11	12	13	14	5	6	7	8	9	10	11	5	6	7	8	9	10	11
11	12	13	14	15	16	17	15	16	17	18	19	20	21	13	14	15	16	17	18	19	10	11	12	13	14	15	16	15	16	17	18	19	20	21	12	13	14	15	16	17	18	12	13	14	15	16	17	18
18	19	20	21	22	23	24	22	23	24	25	26	27	28	20	21	22	23	24	25	26	17	18	19	20	21	22	23	22	23	24	25	26	27	28	19	20	21	22	23	24	25	19	20	21	22	23	24	25
25	26	27	28	29	30	31	29	30					27	28	29	30	31			24	25	26	27	28	29	30	29	30	31				26	27	28	29	30	31		26	27	28	29	30	31			

Вересень 2023						
ПН	ВТ	СР	ЧТ	ПТ	СБ	НД
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						

Рис. 2.7. Вигляд робочого календарю проекту за 2023-2024 роки

Для зручності було створено файл в інструменті MS Project, що дозволяє більш детально оцінити календарний план ітерацій модулів на проекті (рис. 2.8).

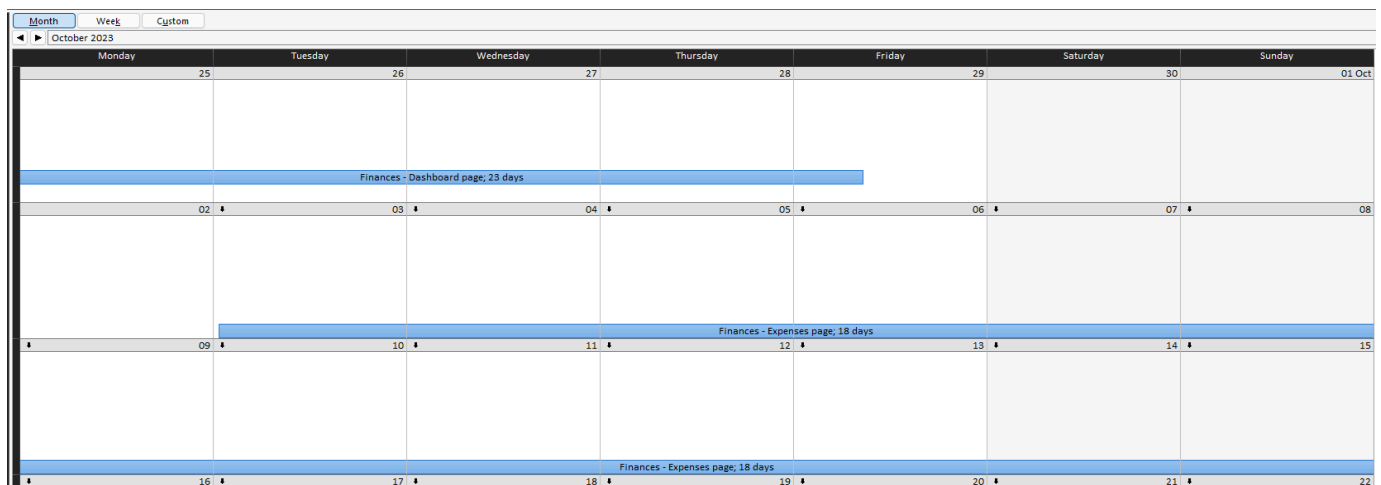


Рис. 2.8. Фрагмент календарного плану ітерацій в MS Project

Діаграма Ганта - це вид діаграми, який містить календарний план робіт, на якому операції відображені вертикально, дати вказуються горизонтально, а тривалість операцій відображають у вигляді горизонтальних смуг, які розміщують відповідно до дат початку та закінчення [14, ст. 189].

Окрім календарного плану MS Project має зручну діаграму Ганта яка демонструє в деталях завершеність ітерацій проекту та їх поточний стан відносно запланованих встановлених базових ліній (таймлайнів) проекту на (рис. 2.9).

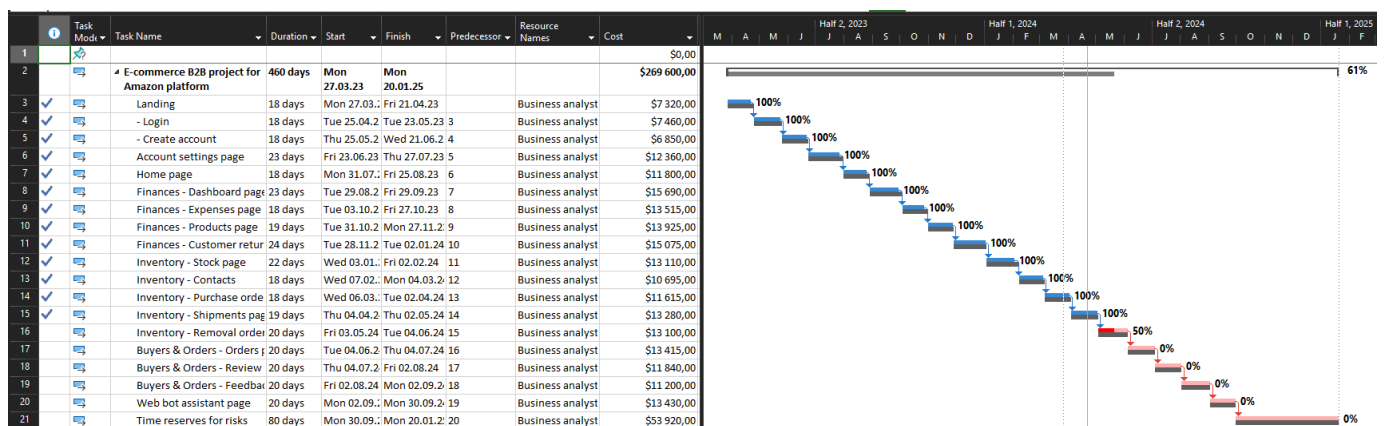


Рис. 2.9. Демонстрація ітерацій на діаграмі Ганта

Не варто забувати про те, що ми маємо аутсорсингову компанію яка має низку проєктів, а отже вона має мати можливість реагувати на паралельні проєкти. Тому варто забезпечити команду можливістю працювати на декількох проєктах в розрізі дня по 4 години. З чим нам допомагає діаграма робочих ресурсів (рис 2.10) де можна налаштувати завантаженість кожного співробітника на ітерації.

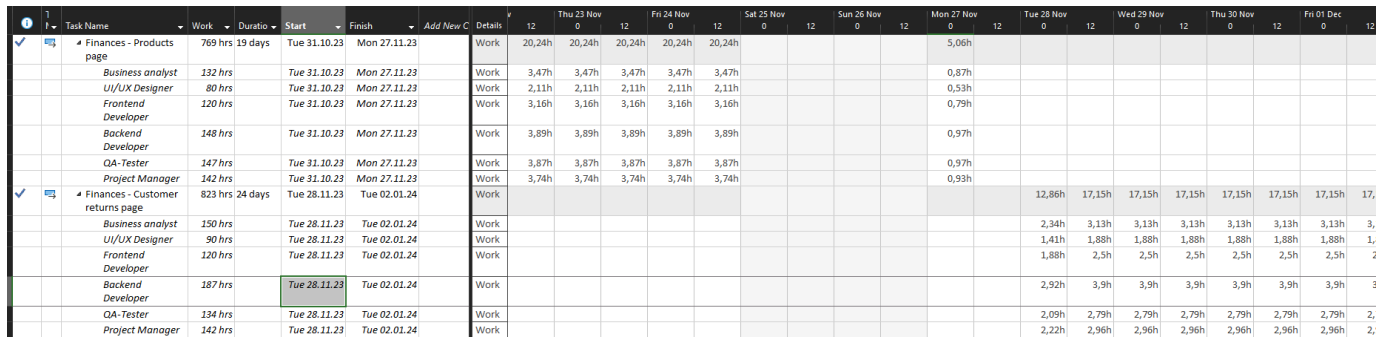


Рис. 2.10. Діаграма завантаженості робочих ресурсів

Управління ресурсами є ключовим елементом успішного виконання проєктів. В Microsoft Project існують зручні інструменти для цього. Визначення завантаженості ресурсів - перший крок до успішного управління. MS Project надає можливість аналізувати використання ресурсів та розподіляти їх між завданнями. Збалансоване навантаження - це запорука ефективного проєкту оптимізація робочих ресурсів за робочими годинами може відбуватись як вручну, так і автоматично. Максимізувати продуктивність команди, можна використовуючи підказки та рекомендації для ефективного управління ресурсами, а також патерни завантаженості ресурсів [15].

У поточному розділі було проведено детальний аналіз календарного плану та управління ресурсами проєкту. Календарне планування ключовий етап, який дозволяє чітко визначити часові рамки виконання проєкту, розподілити завантаженість персоналу та визначити дні комунікації з клієнтом. Інструмент MS Project, став достатньо корисним для створення та візуалізації календарного плану, що дозволяє зручно відстежувати хід виконання проєкту.

Діаграма Ганта виявилася ефективним інструментом для візуалізації завершеності ітерацій проєкту та їх відповідності запланованим таймлайнам. Крім того, враховано можливість паралельної роботи команди над декількома проєктами, що підтримується за допомогою належного розподілу робочого часу між проєктами.

Управління ресурсами також було ретельно виконано. Визначення завантаженості ресурсів та збалансоване їх розподілення між завданнями є максимально важливим для успішного виконання проєкту. Інструменти MS Project надають зручні можливості аналізу та оптимізації використання ресурсів, що сприяє максимізації продуктивності команди.

У цілому, поточний розділ дозволив детально вивчити та впровадити ефективні методи календарного планування та управління ресурсами, що стане запорукою успішного виконання проєкту.

РОЗДІЛ 3. АНАЛІЗ ІНСТРУМЕНТІВ ТА ПРОЦЕСІВ ДІЯЛЬНОСТІ ЧЛЕНІВ КОМАНДИ

3.1 Вибір інструментів комунікації, дизайну, розробки та документації

Сучасні проєкти не обходяться без інструментів менеджменту та комунікації. За допомогою них відбувається спрощення процесу проведення зустрічей, відстежування проблем з завданнями закріпленими до членів команди. Однією з компаній які поставляють послуги та продукти для менеджменту проєктів є Atlassian. Головними програмами які були обрані для проєкту є Confluence та Jira які розповсюджуються як по платній підписці, так і безкоштовній для команд не більше ніж з 10 осіб.

Confluence – це вебплатформа для спільної роботи та управління знаннями в компаніях та організаціях в особливості для документації. Вона дозволяє користувачам створювати, редагувати та спільно працювати над сторінками з різним контентом, таким як текст, зображення, таблиці, відео та іншими типами даних. Confluence охоплює функції, які дозволяють організаціям зберігати, організувати та керувати знаннями, документами та інформацією про проєкти. Завдяки зручному інтерфейсу та широкому спектру можливостей Confluence забезпечує зручний спосіб роботи між колегами які працюють одному вебпросторі [16, 17].

Atlassian Jira – це ПЗ для управління проектами та задачами, розроблене компанією Atlassian. Jira дозволяє командам стежити за проектами, планувати роботу, відстежувати прогрес, співпрацювати та звітувати про результати. Jira має ряд функцій, що дозволяють керувати проектами, включаючи створення та призначення завдань, установлення термінів та пріоритетів, відстеження часу та вартості робіт, створення звітів та багато іншого. Jira також підтримує різні технології управління проектами, такі як Scrum та Kanban.

Завдяки широкому спектру функцій та інтеграції з іншими інструментами Atlassian, такими як Confluence та Bitbucket, Jira є популярним інструментом для управління проектами в компаніях та організаціях різних розмірів і галузей [16, 18].

Як основною платформою для корпоративного обміну повідомленнями та дзвінками була обрано Slack. Slack – це комунікаційна платформа, яка дозволяє користувачам обмінюватись повідомленнями та файлами в режимі реального часу. Slack було розроблено для полегшення комунікації та співпраці між робочими групами та командами.

За допомогою Slack користувачі можуть створювати різні канали для обговорення різних тем, ділитись файлами, спілкуватись у приватних повідомленнях, проводити відеоконференції та багато іншого. Крім того, Slack може бути інтегрований з іншими сервісами, такими як Google Drive, Dropbox, Trello, Jira, GitHub на основі API з'єднань, що дозволяє користувачам працювати з різними інструментами у єдиному середовищі.

Slack став популярним інструментом для комунікації та співпраці в командах різних розмірів і галузей. Він дозволяє користувачам легко спілкуватись та співпрацювати, зменшуючи потребу у використанні електронної пошти та інших форм комунікації.

Для спільної та зручної роботи програмістів на проєкті має використовуватись система контролю версій. Сам для цього був обраний GitLab – це вебплатформа для управління репозиторіями Git, яка містить в собі відкритий вихідний код. Вона надає

інструменти для керування проєктами, співпраці над кодом та автоматизації процесу розробки програмного забезпечення. GitLab дозволяє створювати приватні репозиторії, ділитися кодом з іншими користувачами та командами, використовувати систему контролю версій Git для керування змінами в коді, створювати гілки та злигтя в коді, виконувати тестування та інтеграцію з іншими сервісами [19].

Як система управління контейнерами для розгортання HTTP серверу, PostgreSQL БД, вебдодатку та WebSocket було обрано Docker. Контейнери - це пакети програмного забезпечення, які включають в себе всі необхідні компоненти для запуску програми, такі як код, залежності, бібліотеки та конфігурації. Docker дозволяє упакувати додаток та всі його залежності у стандартизованому середовищі, яке може бути легко перенесене між різними комп'ютерами.

Docker використовує технологію контейнеризації, яка дозволяє ізолювати програми одну від одної та відділяти їхні середовища від операційної системи. Кожен контейнер містить своє власне середовище виконання, включаючи файли, бібліотеки, залежності та конфігураційні файли. Docker використовує ядро операційної системи хоста для виконання контейнерів, що дозволяє їм бути легкими та швидкими [20].

Щоб виконувати автоматизовані тести та збирати й аналізувати обсяги даних журналів на проєкті використано Graylog. Він забезпечує потужні інструменти для пошуку, фільтрації та візуалізації даних журналу, що дозволяє оперативно виявляти проблеми, відстежувати стан системи та реагувати на події в реальному часі.

Як середовища для розробки ПЗ розробниками було обрано програмні продукти Jet Brains які в свою чергу надають весь необхідний спектр інструментів для інтерпретації та компіляцію коду. Серед середовищ обрано PHP Storm для бекенду та Web Storm для фронт енд розробки.

Як основну систему управління реляційними базами даних обрано PostgreSQL.

Вона дозволяє безперешкодно створювати зберігати та обробляти дані різних типів та структур. Також ця СУБД як і інші платформи мають підтримку різних операційних систем і надають можливість редагувати відкритий код.

Не менш важливим постають інструменти дизайну інтерфейсів. У нашому випадку обрано програму Figma, що дозволяє проектувати інтелектуальні карти, організовувати та візуалізувати структури даних. На основі цього у програмі далі розроблюються UX макети та дизайн матеріали інтерфейсів які передають в розробку після валідації тестувальником.

Іншими інструментами візуалізації складних растрових та векторних об'єктів використано програмне забезпечення від компанії Adobe, а саме: Photoshop для обробки фотографій та складних ефектів та Illustrator, щоб працювати зі згрупованою масштабованою векторною графікою.

Отже, за допомогою поточного набору перерахованих інструментів виконання проекту у всіх аспектах стає більш зручним. Додатково ці інструменти дозволяють контролювати процеси виконання задач у команді та підтримувати постійний зв'язок і виправляти помилки, або вносити нові зміни у разі потреби.

3.2 Вибір технологічного стеку

Підбір технологічного стеку для проекту обумовлюється фактором вимог до кінцевого продукту проєктної діяльності. В головні умови вибору технологічного стеку входять: орієнтовний бюджет, складність архітектури та швидкодія ПЗ, безпека даних, розмір команди розробників і їх рівень знань технологій, можливість масштабованості. Не менш важливим є погляд в бік новизни технологій та їх підтримки після випуску веб-додатку на ринок користувачів. Це має продовжити загальний життєвий цикл програмного продукту і зробити підтримку програмного забезпечення більш простим оскільки більшість бібліотек, високорівневих мов програмування будуть підтримуватись як стандарт індустрії.

На основі досліджених даних взято за основу наступний технологічний стек програмного забезпечення. Для розробки Frontend частини взято три базові мови та фреймворк.

HTML (HyperText Markup Language) – мова яка використовується для створення структури вебсторінок, його наповнення контентом та зв'язування зовнішніх та внутрішніх скриптів.

CSS (Cascading Style Sheets) – мова каскадних таблиць використовується для стилізації і оформлення елементів вебсторінок прописуючи набір параметрів для кожного окремого набору елементів.

JS (JavaScript) – є високорівневою мовою програмування, яка використовується для створення динамічного змісту на вебсторінках. Вона широко використовується для додавання інтерактивності до вебсторінок, включаючи анімацію, обробку подій, валідацію форм, маніпуляцію DOM (Document Object Model) і багато іншого.

Angular Framework – це фреймворк JavaScript для розробки вебдодатків. Він надає структурні компоненти та інструменти для створення потужних інтерфейсів користувача. Набір Angular має в собі CLI (Command Line Interface) інструмент командного рядка, який допомагає швидко створювати, розгортати та управляти проектами під Angular. Також додатково цей фреймворк має бібліотеку компонентів під назвою Angular Material, яка надає готові компоненти із матеріальним дизайном для швидкої інтеграції вебдодатків.

У випадку написання Backend частини додатку було обрано наступні технології.

PHP – основна високорівнева мова програмування для клієнт-серверної логіки вебдодатків та сайтів.

Laravel – це фреймворк PHP, який надає структуру та інструменти для розробки бекенду, включаючи маршрутизацію, роботу з базою даних, аутентифікацію, валідацію форм.

Composer – інструмент що дозволяє управляти залежностями в PHP та встановлювати бібліотеки включаючи Laravel.

REST API (Application Programming Interface) - це інтерфейс програмування додатків, який дозволяє взаємодіяти з вебдодатком чи сервісом за допомогою HTTP запитів. Він базується на архітектурному стилі REST і використовує HTTP методи (GET, POST, PUT, DELETE) для виконання операцій з ресурсами.

SQL (Structured Query Language) – як основна мова програмування для створення та управління базами даних на основі СУБД PostgreSQL.

Поточний стек технологій дозволяє створювати та розгортати програмне забезпечення, контролювати його версії, обмінюватись інформацією між вузлами, писати кастомні функції для модулів веб-додатку. Такий оптимальний набір технологій забезпечує швидкість впровадження та масштабованість проекту.

3.3 Дослідження процесів виконання дизайну

В проєкті розробка дизайну інтерфейсів відбувається поетапно на основі традиційного «дизайну мислення». Який передбачає за собою цілий стратегічний підхід до розробки продуктів, який ставить користувача в центр уваги і використовує ітераційний процес для забезпечення оптимальної взаємодії з користувачем та розв'язання його проблем. У цьому розділі можна побачити наступні ключові етапи цього підходу та його вплив на створення високоякісних інтерфейсів для майбутніх користувачів.

Етапи Design Thinking:

1. Емпатія (Empathize) - на початковому етапі розробки інтерфейсу дизайнери активно спілкуються та співпереживають потребам користувачів. Вони проводять спостереження, інтерв'юють користувачів і збирають дані для глибокого розуміння проблем та потреб аудиторії [21, ст. 2].
2. Визначення проблеми (Define) - на цьому етапі дизайнери аналізують отримані дані та визначають ключові проблеми, з якими стикаються користувачі. Вони

формулюють чітке завдання для подальшої роботи та визначають основні цілі проекту наряду з менеджером та бізнес-аналітиком [21, ст. 3].

3. **Ідеація (Ideate)** - на етапі ідеації команда збирається разом для генерації креативних ідей для вирішення визначених проблем. Важливо створювати безліч варіантів і не обмежувати себе стандартними рішеннями. Тут доцільно використовувати брейншторми за участі команди [21, ст. 4].
4. **Прототипування (Prototype)** - на цьому етапі дизайнери перетворюють обрані ідеї в конкретні прототипи. Це можуть бути макети інтерфейсу, друківані матеріали або інші форми, що допомагають візуалізувати концепції [21, ст. 5].
5. **Тестування (Test)** - прототипи піддаються тестуванню з користувачами для отримання зворотного зв'язку. Дизайнери аналізують реакції та використовують отримані дані для вдосконалення інтерфейсу [21, ст. 6].

Design Thinking дозволяє ставити потреби та вимоги користувачів в центр уваги, забезпечуючи створення інтерфейсів, які відповідають реальним потребам аудиторії. Етап ідеації дозволяє команді дизайнерів виходити за межі стандартних рішень та знаходити креативні підходи до вирішення проблем. Підхід Design Thinking передбачає ітерації та постійне вдосконалення продукту на основі отриманого зворотного зв'язку від користувачів, що допомагає забезпечити високу якість інтерфейсу. Design Thinking підтримує спільну роботу різних фахівців, включаючи дизайнерів, розробників, маркетологів та інші групи, що сприяє комплексному розгляду задач та забезпеченню успішного вирішення проблем.

Використання Design Thinking у розробці дизайну інтерфейсів сприяє створенню продуктів, які не лише естетично привабливі, але й ефективно вирішують завдання та задовольняють потреби користувачів.

3.3.1 Дослідження процесів прототипування користувацького досвіду

Після попередніх етапів які були описані в минулому розділі, можна перейти до опису діяльності та загального дослідження користувацького досвіду при розробці

інтерфейсів. Зазвичай для дослідження таких цілей першопочатково визначаються користувачькі персони у форматі карточок з описом даних майбутніх користувачів. Після чого вони валідуються менеджером проєктів або продуктів знаходячи реальне підтвердження, тому що такі особи дійсно існують в повсякденному житті та ми дійсно розуміємо всі необхідні потреби.

Після того як дизайнер проєкту розуміє для чого дійсно потрібен певний функціонал він вилаштовує логічну структуру та ієрархію майбутнього цифрового продукту використовуючи інструменти прототипування. Далі на основі цих даних та побажань замовника виконується розробка прототипів інтерфейсів, з яких вилаштовується користувачький потік, на основі сценарії використання продукту та випадків в яких може потрапити користувач в процесі роботи. Один з таких прототипів високої якості продемонстровано на (рис. 3.1).

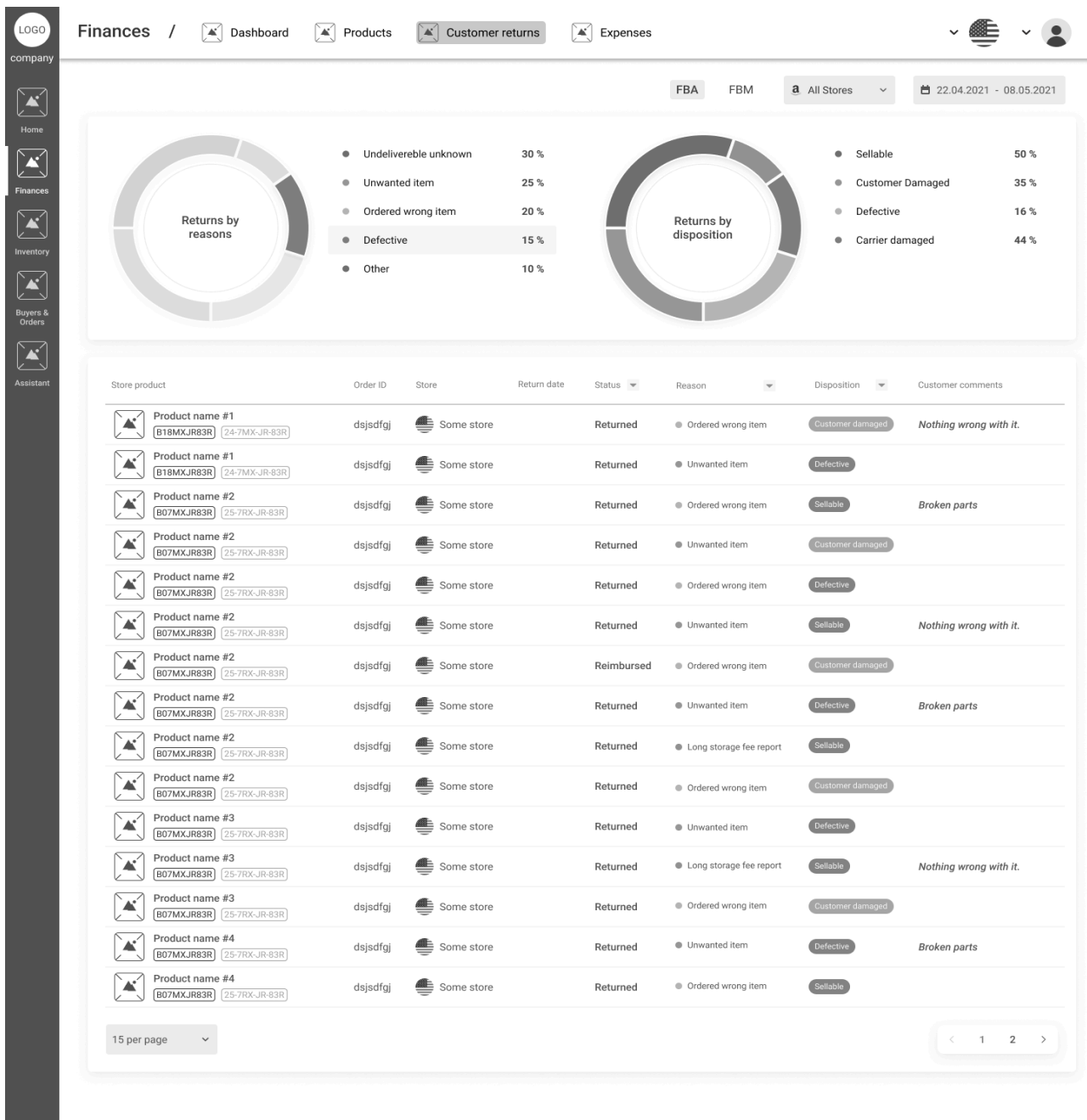


Рис. 3.1. Високоякісний прототип інтерфейсу користувача для сторінки «повернень товарів»

Користувачський потік - це сценарії за якими зазвичай ланцюговою реакцією будуть поводитись користувачі для того, щоб виконати свою ціль перебування на сайті. Таким чином протопити екранів збираються в один потік і на його основі вилаштовуються анімовані клікабельні протопити для демонстрації замовнику.

Не рідко в ході роботи дизайнери проекту використовують найкращі практики які можна знайти в інтернеті якими діляться інші спеціалісти на форумах та інформаційних ресурсах. Найкращий приклад пошук інформації щодо реструктурування великих даних у форматі таблиць для девайсів з невеликими екранами.

Додатково існують випадки коли дизайнери сумніваються у виборі одного з декількох варіантів прототипів для майбутнього використання на ресурсі. На основі цього дизайнери виконують A&B тести які надають можливість протестувати на користувачах який варіант інтерфейсу їм буде більш до вподоби на основі регулярного використання.

3.3.2 Елементи візуалізації дизайну

Результати дослідження інтерфейсів та прототипів після підтвердження замовником передають на етап розробки дизайну. В свою чергу дизайнер обирає дизайн систему, як основну мову комунікації з користувачем в залежності від технологій розробки та функціональних обмежень платформи.

В дизайн систему (рис. 3.2) входить інформація, щодо використаних варіантів шрифтів їх розмірів та накреслення, кольорова схема з нейтральними та брендинговими кольорами, яка надає змогу звернути увагу користувача на елементи інтерфейсів, що надають можливість взаємодіяти з сервісом та маніпулювати з даними в додатку інформуючи користувача про те які дії можна виконати на основі візуальних ефектів та станів елементів, а також система відступів для стандартизованої розмітки по всьому вебдодатку з електронної комерції.

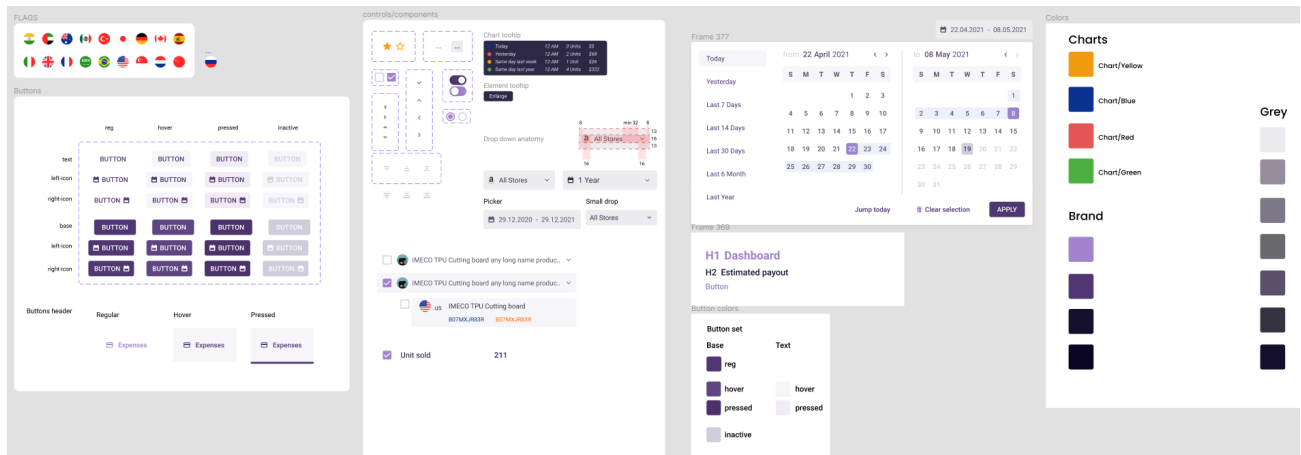


Рис. 3.2. Приклад дизайн системи проекту

Далі дизайнер обирає референси та починає відмальовувати макети інтерфейсів на основі яких відбувається подальша розробка програмного забезпечення та візуалізація. Після цього етапу важливо щоб весь інтерфейс був затверджений замовником та протестований QA-тестувальником. Ну а також у випадку крайньої потреби дизайнер може продемонструвати складні анімації, щоб презентувати замовнику, або донести до розробників які саме ефекти анімації і таймінги мають бути використані в продукті і при яких умовах взаємодії користувачів.

3.4 Дослідження процесів розробки

Розробка програмного забезпечення це ключовий процес який об'єднує всі етапи проекту та попередніх досліджень оточення в єдиний програмний продукт, який направлений на задоволення користувачів.

Починаючи з Frontend розробки після збору вимог технічного завдання проекту відбувається відбір необхідного стеку технологій, який буде використовуватись для розробки всього необхідного функціоналу. Далі на основі досліджень найкращих практик відбувається прототипування інтерфейсів на якому Frontend розробник займається валідацією ідей які пропонує дизайнер для покращення користувацького

досвіду. Після чого дизайнер на основі дизайн системи готує макети інтерфейсу та створює анімації, які будуть відображатись під час взаємодії користувача з інтерфейсом. В цей час розробник займається пошуками відповідних бібліотек для анімації інтерфейсу.

Після завершення підготовки макетів дизайну вони передаються розробнику на верстання на основі мов HTML, CSS створюючи каркас додатку. Після завершення верстання відбувається налаштування поведінки елементів інтерфейсу розробником на основі фреймворку Angular та мови JavaScript. На цьому етапі розробляються різні компоненти та функціональність додатка, такі як форми, кнопки, навігаційне меню. Після готовності компонентів на основі станів елементів відбувається створення функцій та запитів для комунікації з серверним кодом за допомогою API.

Паралельно з Frontend розробником працює Backend розробник, який розроблює архітектуру та базу даних додатку. Для цього Backend розробник використовує інструмент візуалізації Data Flow Diagram. Діаграма потоків надає можливість відтворити процеси комунікації між персональним комп'ютером користувача з поданими мережевими вузлами для використання веб-додатку. Частина процесів ведеться на основі взаємодії користувача з третіми особами й компаніями які поставляють послуги з перевезення та закупівлі товарів за певними запитам користувача і таким чином вони логуються в нашій системі. Інша частина процесів виконується окремо автоматично на основі клієнт-серверних запитів які повертають результат, на основі персональних даних користувача та даних продажів на основі інформації з серверів компанії Amazon. Таким чином дані аналізуються, трансформуються у форматі табличних даних та графіків якими зручно користуватись з однієї платформи для вирішення більшості бізнес задач. Візуалізація клієнт-серверної архітектури в форматі діаграми потоків даних продемонстровано на (рис. 3.3).

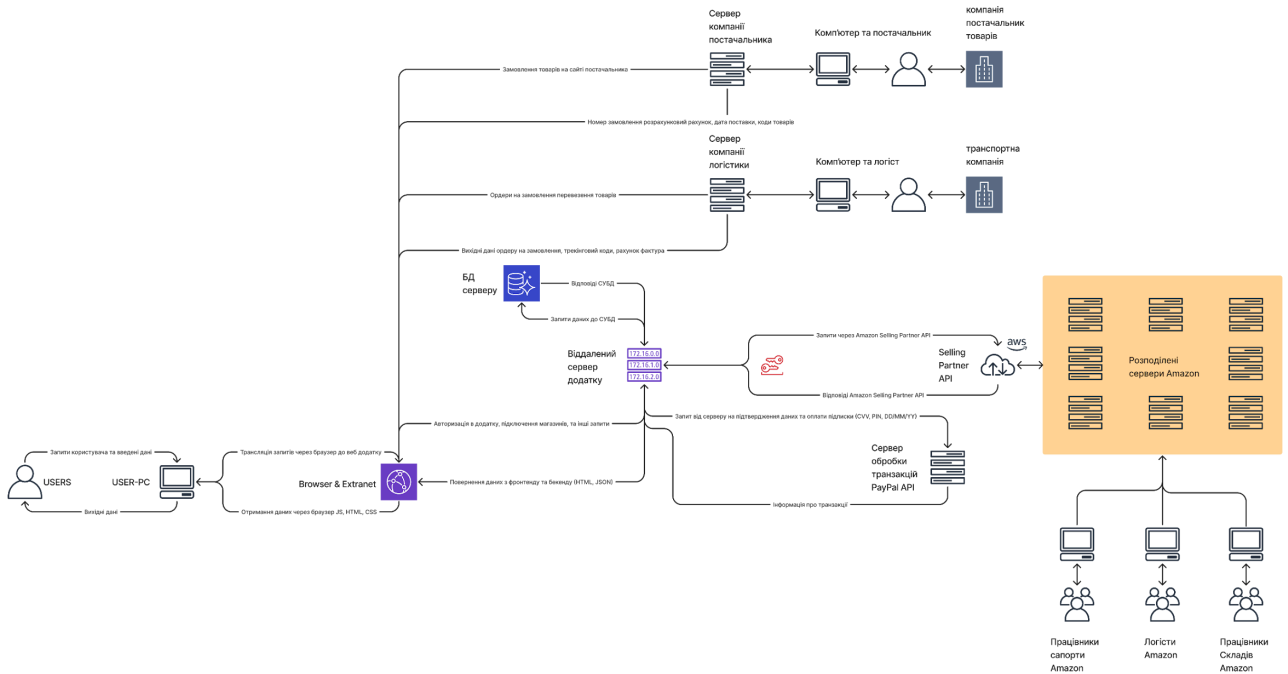


Рис. 3.3. Клієнт-серверна архітектура додатку на основі діаграми потоків даних

Важливість клієнт-серверної архітектури полягає в забезпеченні ефективного та безпечного обміну даними між клієнтськими та серверними компонентами проекту. Ця архітектура дозволяє розділити функціональні обов'язки між клієнтами, які надають інтерфейс для користувачів, та серверами, які забезпечують обробку даних та логіку додатку. Вона сприяє зменшенню навантаження на клієнтські пристрої, полегшує синхронізацію даних та забезпечує централізоване керування та безпеку інформації.

Відштовхуючись від напрацювань архітектури та вимог до продукту. Відбувається створення структури бази даних проекту. При повній готовності БД, бекенд розробник займається написанням логіки на мові PHP та фреймворку Laravel і налаштовує API інтеграції для передачі даних в мережі, що дозволяють клієнтській частині взаємодіяти з сервером проводити маніпуляції з даними й транзакції. Приклад інтеграції Amazon Selling Partner API на (рис. 3.4).

```
16 class SellingPartnerApi
17 {
18     use Traits\Feed,
19         Traits\Finance,
20         Traits\InboundShipment,
21         Traits\Inventory,
22         Traits\Marketplace,
23         Traits\Order,
24         Traits\Product,
25         Traits\Report,
26         Traits\FulfillmentOrder,
27         Traits\Store,
28         Traits\Messaging;
29
30     private Credentials $credentials;
31     private Client $client;
32
33     private Store $store;
34
35     private string $sellerId;
36     private string $marketplaceId;
37     private string $awsService;
38     private string $awsRegion;
39
40     public function __construct($store)
41     {
42         $timeouts = !isTest() ? ['timeout' => 10, 'connect_timeout' => 3] : ['timeout' => 999, 'connect_timeout' => 999]; //when debugging tests we
         need big timeout
43
44         $this->client = new Client($timeouts);
45         $this->credentials = new Credentials(config('app.aws_selling_partner_access_key_id'), config('app.aws_selling_partner_secret_access_key'));
46
47         $this->store = $store;
48
49         $this->sellerId = $store->seller_id;
50         $this->awsService = $this->store->marketplace('awsService');
51         $this->awsRegion = $this->store->marketplace('awsRegion');
52         $this->marketplaceId = $this->store->marketplace('marketplaceId');
53     }
54 }
```

Рис. 3.4. Приклад підключення Amazon Seller Partner API за допомогою PHP

Таким чином REST-орієнтований API Amazon Seller Partner дозволяє комунікувати з вузлами мережі Amazon і робити зміни в даних: завантажувати їх, обробляти, а також відсилати назад дані для внесень змін на регіональних маркетплейсах. Інші приклади коду застосунку проілюстровані на (рис. А.1 - А.4) в додатку А.

Після ітераційних поставок функціональності ПЗ відбувається тестування коду бекенду включаючи тести безпеки, юніт-тестування, навантаження системи. Виконавши всі тести відбувається розгортання додатку на тестовому середовищі та перенесення коду на головний репозиторій системи GIT.

3.5 Проктування бази даних

Реляційна БД це один з основних компонентів e-commerce сервісу який надасть можливість користувачам взаємодіяти з товарами займатись їх менеджментом на основі реляційних зв'язків табличних даних, мови запитів. Користувачі додатку

мають бачити на платформі ключові показники, які уособлюють в собі ефективність процесу ведення свого бізнесу з продажів.

Реляційна База даних є центральним елементом, де зберігаються всі важливі дані про товари, клієнтів, транзакції, замовлення та іншу інформацію, необхідну для функціонування й аналізу проєкту. Вона є надійним інструментом для зберігання, управління та доступу до даних у зручному та ефективному форматі. Крім доступу на читання користувачі мають можливість персоналізувати певну інформацію в базі даних, в особливості вводити нову інформацію, корегувати наявні та видаляти застарілі або неліквідні дані [22, ст. 1 - 2].

З детально проведеного бізнес-аналізу вимог до програмного забезпечення було змодельовано концептуальну архітектуру модулів проєкту на основі якої буде відбуватись проєктування бази даних для e-commerce вебдодатку під платформу Amazon. Взаємодію розділів модулів візуалізовано у вигляді ромбоподібних геометричних фігур, а сторінки модулів продукту продемонстровані у вигляді закруглених прямокутних форм. Загальну архітектуру модулів можна переглянути на (рис. 3.5).

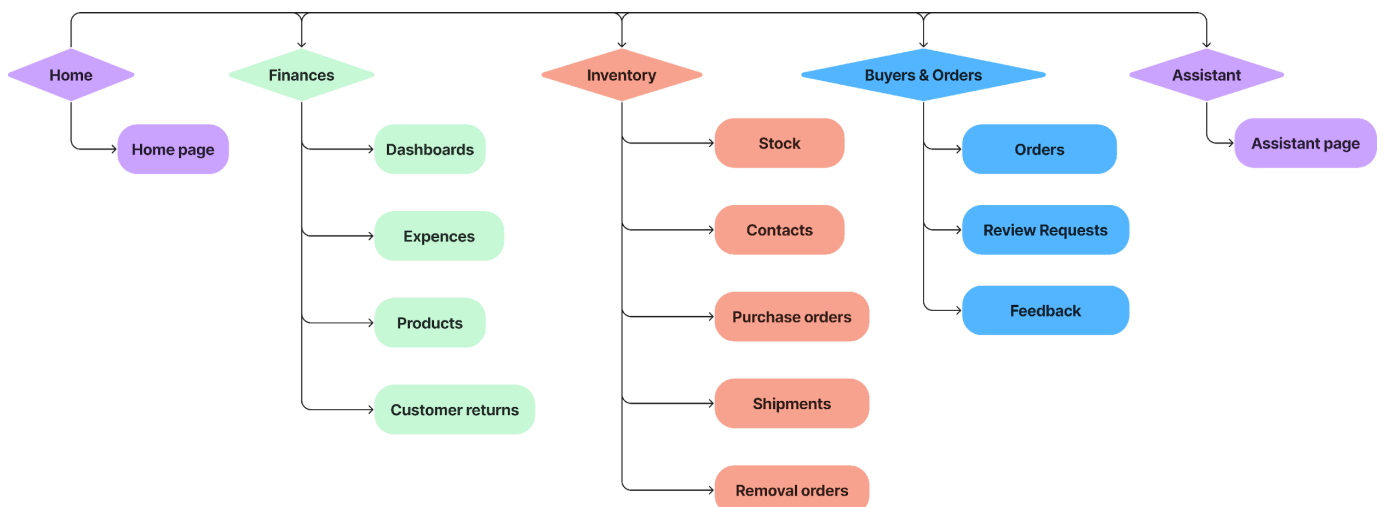


Рис. 3.5. Концептуальна архітектура модулів проєкту

Отже, можна помітити, що продукт проекту поділений на 5 модулів та підсторінки. Перший модуль «Home» який має одну сторінку, на ній розташовані віджети та таблиці швидкого доступу моніторингу останніх активностей в магазині – це виконано задля швидкого моніторингу найпріоритетніших показників користувачами платформи, таких як останні замовлення, перевезення та дії з доходами та товарами. Фізична модель БД модуля «Home» продемонстрована на (рис. 3.6) [23, ст. 5].

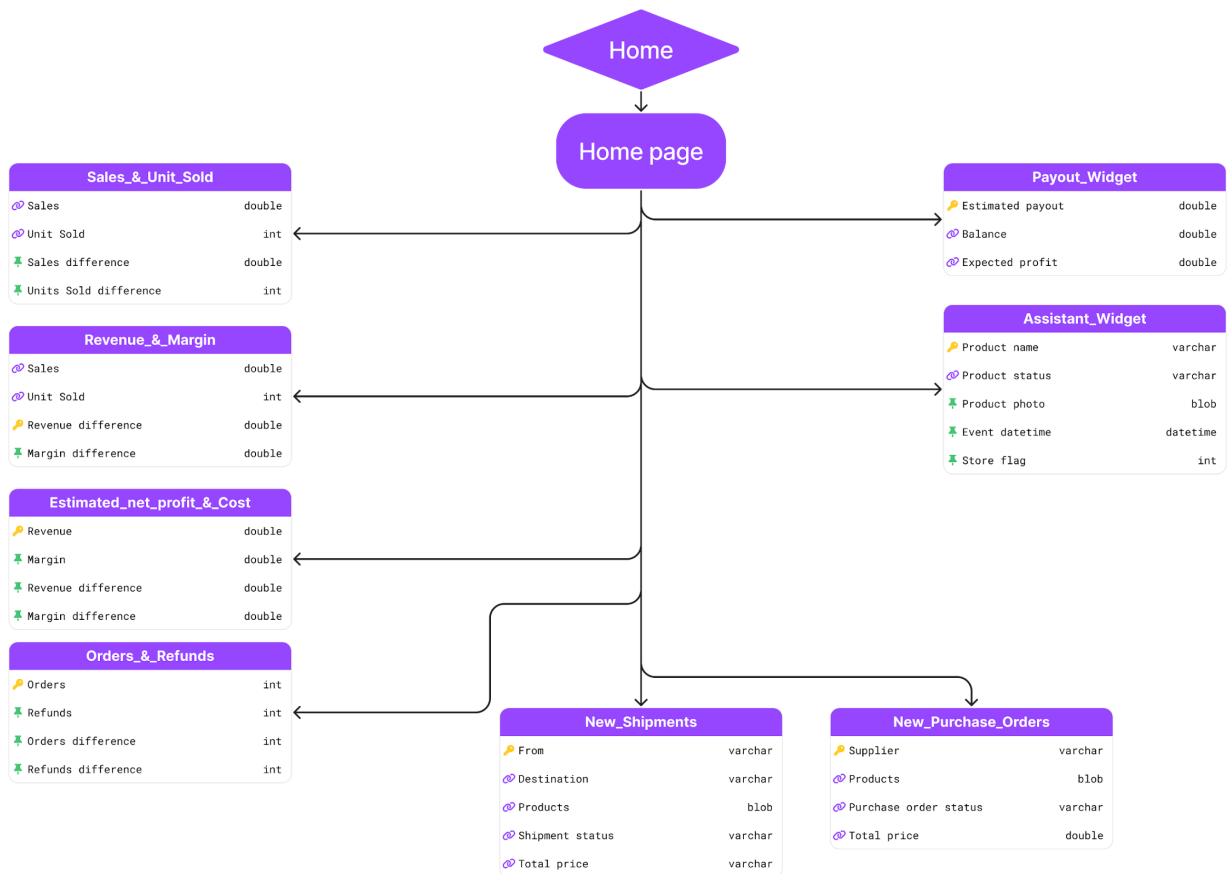


Рис. 3.6. Фізична модель бази даних модулю «Home»

Наступний модуль, який відповідає за підготовку фінансової та рекламної аналітики продажів. В собі уособлює та формує велику кількість графічних стовпцевих графіків та табличних даних зібраних в одному місці задля зручності користувачів. Фізична модель модуля «Finances» виглядає наступним чином (рис. 3.7).

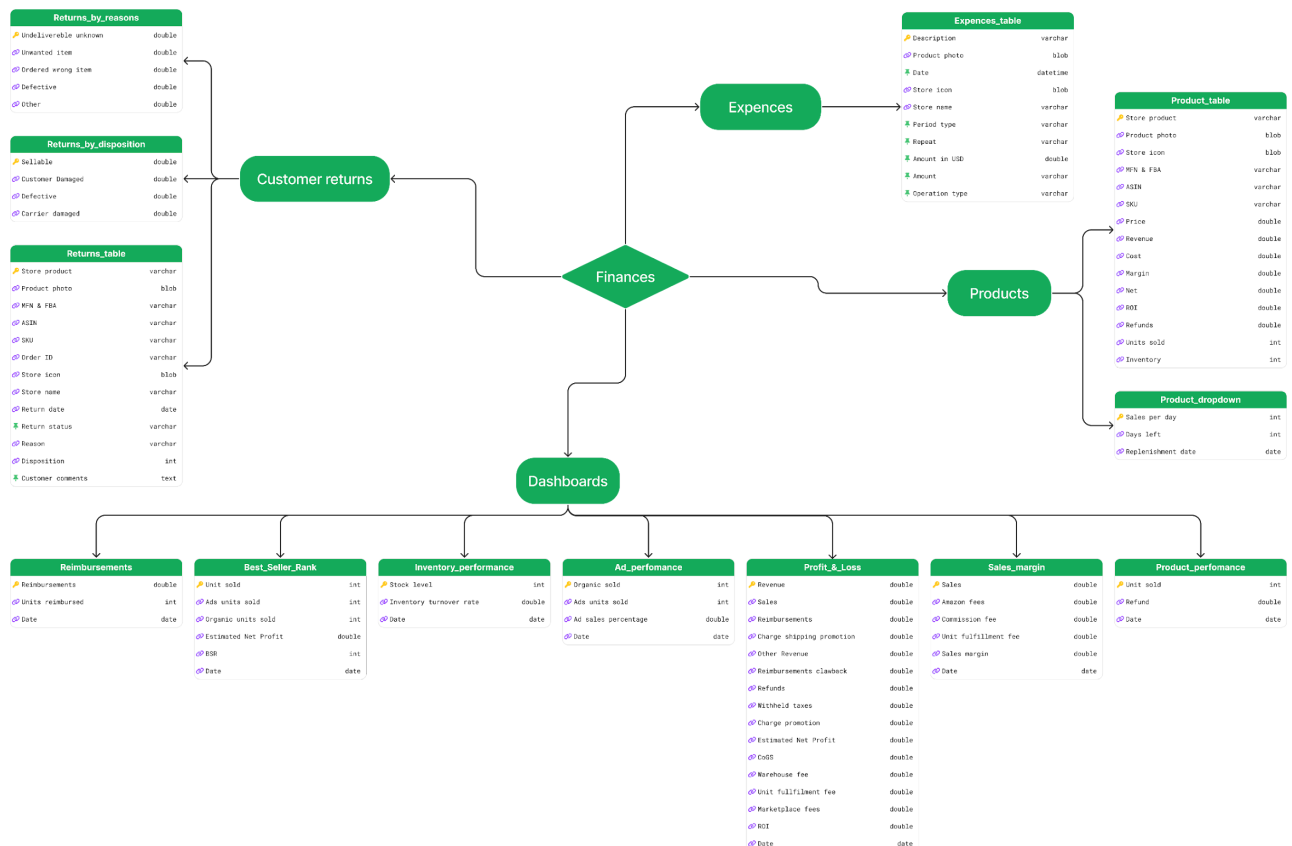


Рис. 3.7. Фізична модель бази даних модулю «Finances»

Модуль «Inventory» створений для збору інформації про інвентаризацію та обробку. Він поділений на 5 сторінок. «Stock» сторінка в яку входить інвентаризація всіх магазинів. «Shipments» розділ для опрацювання транспортування товарів, «Contacts» для створення нових постачальників товарів. «Purchase orders» для закупок нових товарів на склади користувачів. «Removal orders» розділ для

інформування користувачів про дії з товарами після довготривалого зберігання. Фізичну модель бази даних модулю «Inventory» можна переглянути на (рис. 3.8).

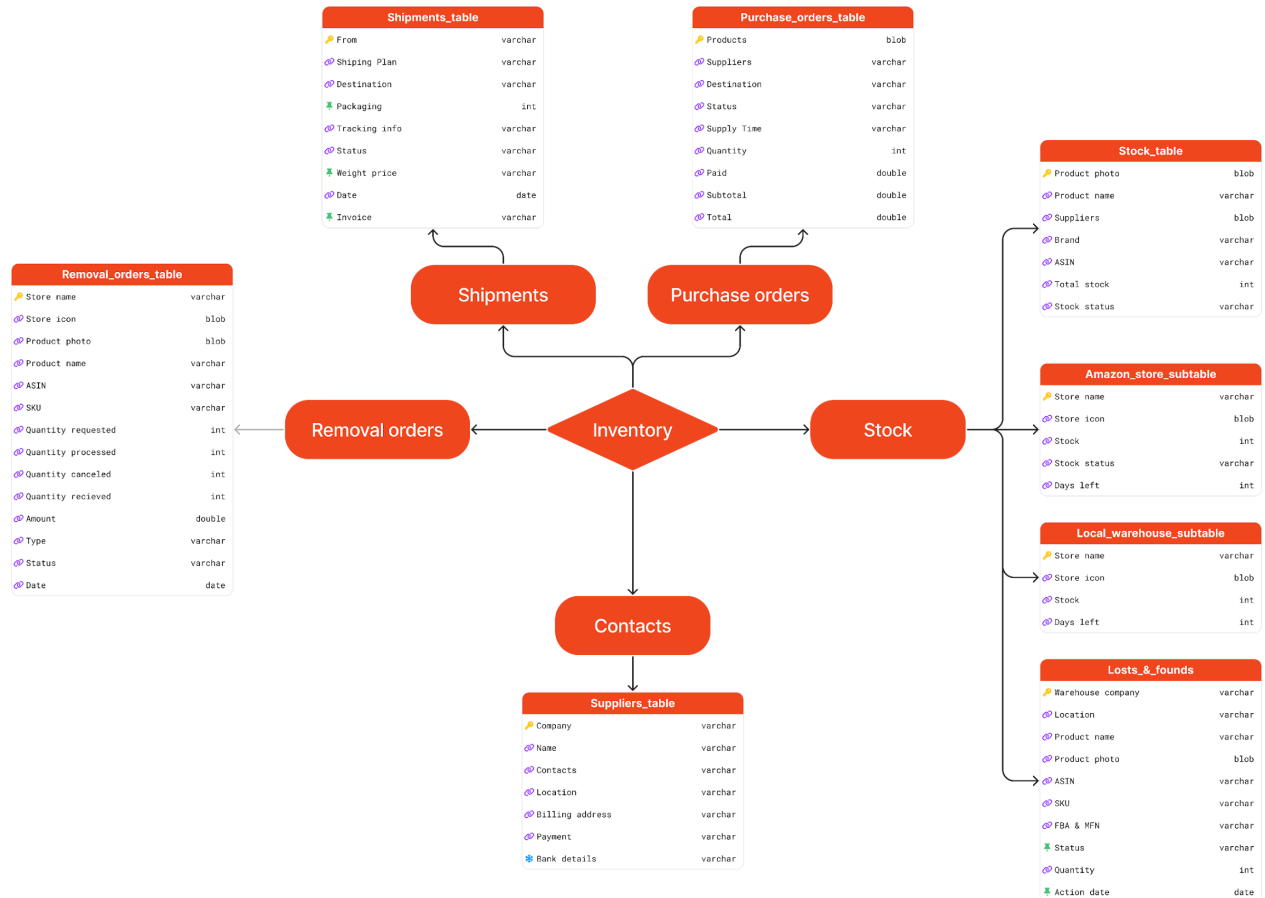


Рис. 3.8. Фізична модель бази даних модулю «Inventory»

Модуль «Buyers & Orders» слугує для забезпечення підприємців Amazon швидким зворотнім зв'язком від покупців. Таким чином бізнесмени мають змогу відреагувати на замовлення покупців їх товарів, промоніторити відгуки про якість товарів і додатково зробити запити на відгук про товар. Фізичну модель бази даних модулю «Buyers & Orders» можна переглянути на (рис. 3.9).

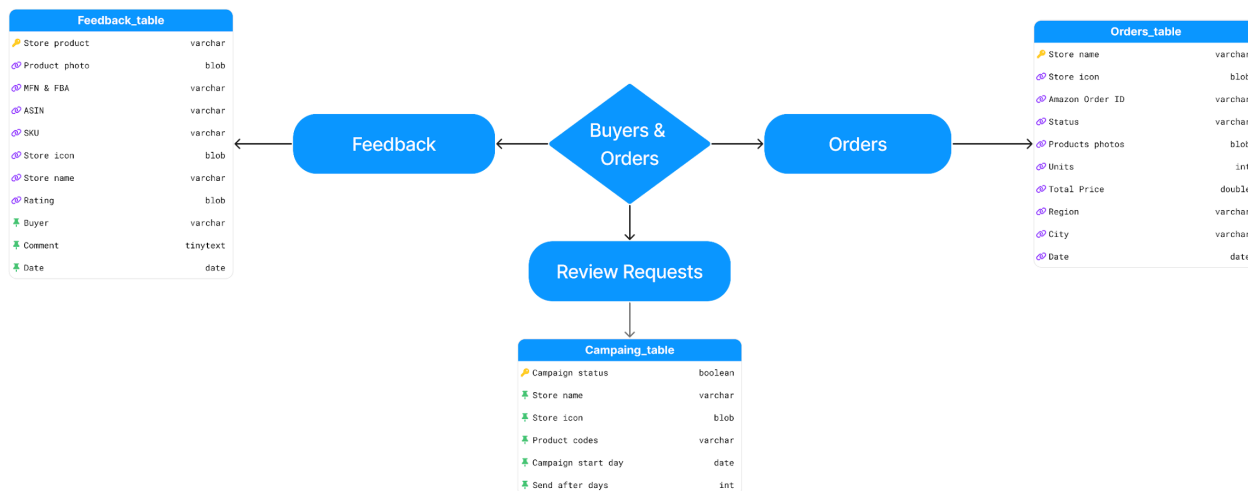


Рис. 3.9. Фізична модель бази даних модулю «Buyers & Orders»

Останній модуль формує дані про процеси на майданчику продажів за допомогою бази даних та бота. Ілюстрація фізичної моделі бази даних модулю «Assistant» (рис 3.10).

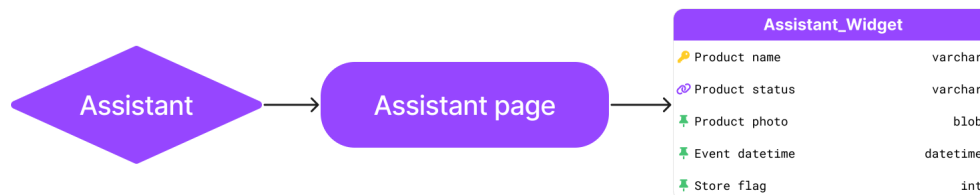


Рис. 3.10. Фізична модель бази даних модулю «Assistant»

Враховуючи подані концепції можна зробити висновки, що на цей момент часу будь-яка модель бази даних для e-commerce уособлює в собі структуровані та нормалізовані набори різних типів даних. На яких можна проглянути параметри відношення, індексацію між різними таблицями при створенні та первинні ключі.

3.6 Тестування програмного продукту

Тестування програмного забезпечення, ключовий і кінцевий етап в розробці нового функціоналу який використовується для перевірки якості. Під час роботи з програмним забезпеченням тестувальник виконав наступні види робіт:

Ознайомлення із вимогами - тестувальник ретельно вивчає вимоги до вебдодатка, щоб зрозуміти очікувані функціональність та нефункціональні властивості.

Планування тестування – процес включає в себе визначення обсягу тестування, вибір типів тестів, призначення ресурсів та планування термінів.

Функціональне тестування – перевірка основної функціональності розроблюваного ПЗ. Включає тестування введення та виведення даних, обробку форм, роботу логіки додатка та інші функціональні аспекти.

Навантажувальне тестування – вид тесту для визначення максимальної робочої спроможності вебдодатка, перевірка стійкості та продуктивності при великому обсязі одночасних користувачів або запитів.

Тестування безпеки – виявлення та вирішення потенційних уразливостей вебдодатка, перевірка на вразливості до атак, таких як SQL-ін'єкції, перехоплення сесій.

Тестування сумісності – визначення та перевірка сумісності розроблюваного функціоналу вебдодатка з різними браузерами, операційними системами та пристроями.

Тестування інтеграції – перевірка взаємодії між різними компонентами системи для забезпечення їх правильної роботи як єдиного цілого, зазвичай виконується на тестовому репозиторії додатку.

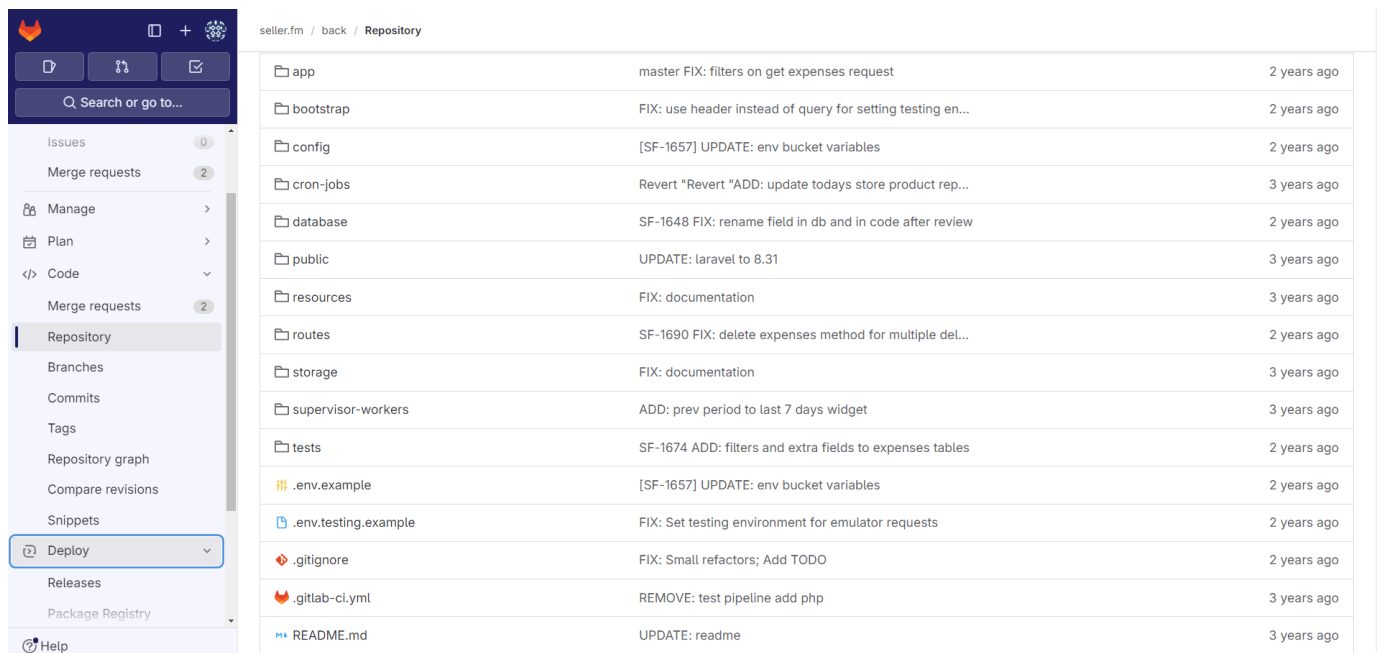
Регресійне тестування – виконання цього типу тесту означає що нові зміни в коді не впливатимуть на поточну функціональність. Використовуючи автоматизовані тести, перевірка на відсутність регресійних помилок.

Автоматизоване тестування – використання інструментів та скриптів для автоматизованого тестування та прискорення процесу тестування, щоб забезпечити більше покриття коду, який потребує тестування.

Тестування відмовостійкості – проводиться для того, щоб простежити здатність вебдодатка працювати під відмовами компонентів та вирішення проблем без втрати функціональності.

Звітність та документація – останній етап перед закриттям задачі це звітування про результати тестування, документування виявлених помилок та розробка інструкцій для користувачів або інших членів команди.

Ну і кінцевим етапом після тестування програмного забезпечення проєкту відбувається перенесення даних з Stage на Master гілку (рис. 3.11) в системі GitLab.



The screenshot shows the GitLab web interface for a repository named 'seller.fm'. The left sidebar contains navigation options: Issues (0), Merge requests (2), Manage, Plan, Code, Merge requests (2), Repository (selected), Branches, Commits, Tags, Repository graph, Compare revisions, Snippets, Deploy, Releases, Package Registry, and Help. The main area displays a file tree with folders like 'app', 'bootstrap', 'config', 'cron-jobs', 'database', 'public', 'resources', 'routes', 'storage', 'supervisor-workers', 'tests', and files like '.env.example', '.env.testing.example', '.gitignore', 'gitlab-ci.yml', and 'README.md'. Each file entry includes a commit message and a timestamp (e.g., '2 years ago').

File/Folder	Commit Message	Time
app	master FIX: filters on get expenses request	2 years ago
bootstrap	FIX: use header instead of query for setting testing en...	2 years ago
config	[SF-1657] UPDATE: env bucket variables	2 years ago
cron-jobs	Revert "Revert "ADD: update todays store product rep..."	3 years ago
database	SF-1648 FIX: rename field in db and in code after review	2 years ago
public	UPDATE: laravel to 8.31	3 years ago
resources	FIX: documentation	3 years ago
routes	SF-1690 FIX: delete expenses method for multiple del...	2 years ago
storage	FIX: documentation	3 years ago
supervisor-workers	ADD: prev period to last 7 days widget	3 years ago
tests	SF-1674 ADD: filters and extra fields to expenses tables	2 years ago
.env.example	[SF-1657] UPDATE: env bucket variables	2 years ago
.env.testing.example	FIX: Set testing environment for emulator requests	2 years ago
.gitignore	FIX: Small refactors; Add TODO	2 years ago
gitlab-ci.yml	REMOVE: test pipeline add php	3 years ago
README.md	UPDATE: readme	3 years ago

Рис. 3.11 – Загальна ієрархія GIT репозиторію проєкту

РОЗДІЛ 4. МОНІТОРИНГ ВИКОНАННЯ ПРОЄКТУ ТА ЗВІТУВАННЯ

4.1 Постановка задач, робіт та дедлайнів

Створення задач починається з постановки вимог. Їх створення виконується бізнес-аналітиком та апробація виконується з власником продукту, який допоможе визначити кінцеві характеристики та функціонал потрібні вебдодатку з електронної комерції. Далі на основі вимог, цілей проєкту, розробленої WBS та інформаційної структури проєкту відбувається формулювання й постановка задач на основі безлічі допустимих альтернатив та критеріїв вибору [24, ст. 6]. У програмному забезпеченні Jira є наступні цільові сутності для постановки задач:

- epic – це сутність яка репрезентує досягнення певної довгострокової цілі і закріплюється задачами з поточного спринту;
- task – тобто задача проєкту, яка має свій опис відношення до певного спринту;
- sub-task – іноді задачі мають комплексну та послідовну кількість дій яку можна розбити на невеличкі підзадачі для цього слугує sub-task;
- story – умовна користувачька історія, яка описує потреби в певному функціоналі та може бути пов'язана з іншими задачами спринту;
- bug – сутність яка створюється для простежування та описання проблем програмного забезпечення.

Отже, беклог таск менеджера наповнюється задачами пов'язаними з epic та story після чого відбувається оцінювання спринту в story points за scrum технологією в залежності від об'єму та складності задач [25]. Оцінювання задач в scrum points йде за наступною шкалою:

- 0 – завдання не потребує часу або може бути виконано дуже швидко;
- 1 – дуже маленька задача, яка займе дуже мало часу;
- 2 – мала задача, яку можна виконати досить швидко;
- 3 – задача середньої складності, вимагає трохи більше часу та уваги;
- 5 – потребує значної уваги та може зайняти багато часу;

- 8 – велика задача, яка потребує великої кількості часу та уваги;
- 13 – дуже велика задача, яка потребує значних зусиль та часу.
- 20 – величезна задача, яка потребує розбиття на менші частини;
- 40 – задача настільки велика або незрозуміла, що її оцінка непрактична.
- 100 – задача розміром з проекту також потребує перерозподілу та розбивки.

Приклад оформлення користувацької історії зі створення лендінгу веб-додатку в одному зі спринтів проекту наведений на (рис. 4.1).

The screenshot displays a Jira issue page. At the top, there are two issue keys: SF-5 and SF-9. The main title is "Створити лендінг веб-додатку". Below the title, there are action buttons: "Attach", "Create subtask", "Link issue", and a dropdown menu. The "Description" section contains the text: "Як замовник проекту, я хочу, щоб моя команда розробила лендінг-сторінку для мого e-commerce веб-додатку на Amazon, щоб привернути більше уваги для потенційних користувачів додатку." Below the description, the "Критерії прийняття:" (Acceptance Criteria) section lists five numbered items: 1. Лендінг-сторінка повинна бути простою у використанні і навігації. 2. Вона повинна включати інформацію про сервіси які надає веб-додаток. 3. Вона повинна мати інформацію про плани підписок на сервіс. 4. Вона повинна мати контактну інформацію та форму зворотного зв'язку для покупців. 5. Вона повинна бути оптимізована для пошукових систем, щоб покупці могли легко знайти мій веб-сайт. On the right side, there is a "Details" sidebar with the following information: Reporter: Vlad Havryliuk; Releases: (empty); Labels: None; Story Points: 20; Sprint: Доска Спринт 1; Priority: Highest; Parent: NEW SF-5 Аналіз ринку ко...

Рис. 4.1. Користувацька історія зі створення лендінгу веб-додатку

На наступному (рис. 4.2) продемонстровано оформлення задачі, яка включає в себе опис, критерії прийняття, статус, відповідальну особу та пріоритет виконання.

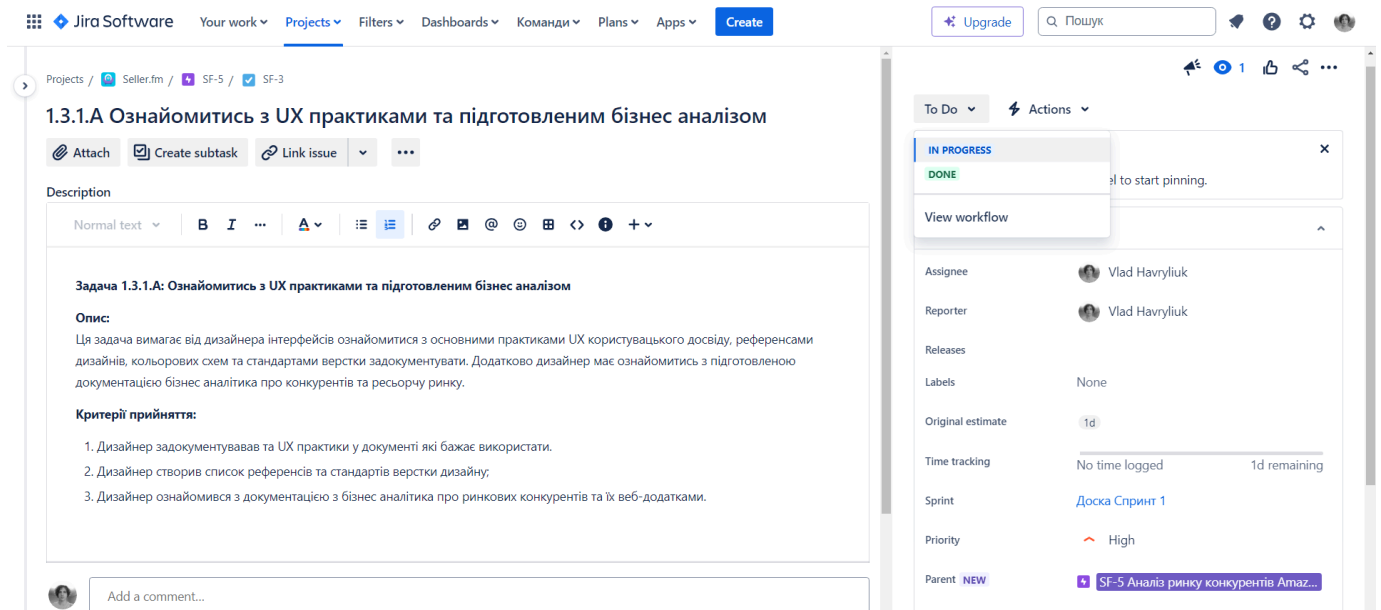


Рис. 4.2. Приклад задачі для дизайнера інтерфейсів в менеджері задач Jira

Після детального планування задач, спринт має наступний вигляд (рис. 4.3).

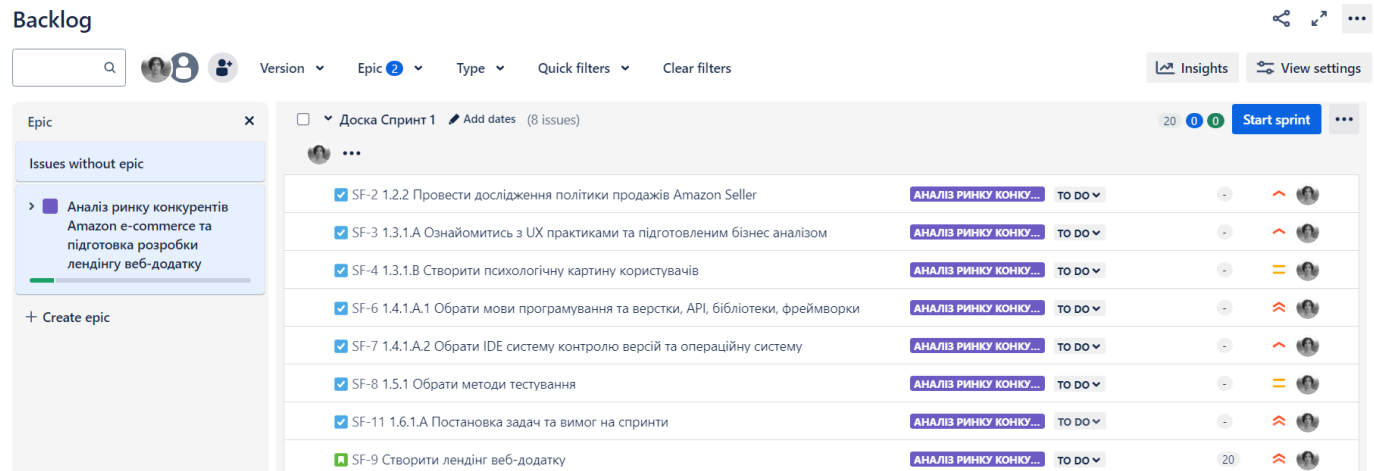


Рис. 4.3. Демонстрація запланованої ітерації спринту в беклозі інструменту Jira

4.2 Контроль виконання робіт

Невід'ємною частиною виконання робіт є процес їх контролю, відслідковування прогресу виконання та наступної звітності. Виконання робіт починається після чіткої постановки обсягів робіт на спринт плануванні та їх

оцінювання, але окрім цього перед початком спринту команда також виконує ще одну не менш важливу церемонію технології scrum. Таку церемонію найменують daily scrum meeting, тобто це формат попередньо спланованих зустрічей команди на початку дня. Така церемонія існує для того, щоб команда проєкту була усвідомлена про заплановані роботи один одного та синхронізувала свої плани на робочий день заздалегідь. Зазвичай такі щоденні зустрічі займають від 30 до 60 хвилин часу, тобто кожному члену команди відводиться від 5-10 хвилин на пояснення своїх планів робіт та можливість пояснити які процеси, чи явища на проєкті блокують продовження виконання задач. Під час зустрічі не менш важливим постає можливість для проєктного менеджера зібрати та агрегувати задачі та цілі команди в один список робіт над якими має працювати команда і додати певні коментарі до них.

Під час виконання робіт також ведеться комунікація між членами команди, яка допомагає у вирішенні складних та комплексних задач. По завершенню задач співробітники проєкту заходять в програму для менеджменту задач та змінюють статус задачі у стан готовності до перевірки. На основі такого запиту доєднується QA-тестувальник й проєктний менеджер і починається перевірка задачі за критеріями прийняття виконаних робіт. Після таких маніпуляцій задача, або підтверджується та переходить у статус виконаних робіт, або повертається до списку робіт які потрібно переробити з попереднім коментарем від відповідальної особи за оцінювання якості виконаної роботи.

Ближче до кінця робочого дня проєктний менеджер займається загальним моніторингом прогресу виконаних робіт та помічає які серед запланованих робіт були виконані, а які були переслані на повторне перевиконання. При цьому робота береться виконавцем на перевиконання за умови інформування та попередньої згоди проєктного менеджера, оскільки достатньо часто трапляються перевитрати часових ресурсів на задачі які не були розраховані на такі об'єми часу та фінансів.

Після виконаної ітерації спринту відбувається процес демонстрації функціоналу та виконаних об'ємів робіт клієнту. У демонстраційну звітність

замовнику входять приклади функціоналу, поширення доступів до файлів та даних і знімки екрану або відеозаписи, що слугують звітністю про виконані роботи. Витрачений час на роботи оформлюється у вигляді розрахункового рахунку де можна прозоро побачити погодинну ставку кожного члена команди у вигляді розрахунків добутку погодинної ставки працівника і фактично затрачених годин. Розрахунковий рахунок у такому випадку має бути сплачений за домовленістю сторін у визначені терміни. Достатньо часто для стейкхолдерів проекту застосовується демонстрація дорожньої карти, задля відображення загального прогресу досягнень проекту поділеної на річні квартали, де виділяються довгострокові цілі проекту та бажані досягнення.

4.3 Управління ризиками проекту

4.3.1 Загальні положення про ризики та фреймворк MSF

Ризики та загрози постійно переслідують будь-який проект, в особливості якщо річ йде про ІТ-сферу де всі процеси, вимоги та технології достатньо динамічно змінюються. Управління ризиками в проекті відбувається на основі аналізу та вивчення джерел виникнення загроз та прийнятті рішень на основі експертних оцінок. Зазвичай вони впливають на декілька параметрів або збільшення часу виконання робіт та фінансових витрат на проект, або загалом призводять до критичних ситуацій, наприклад ліквідації проекту.

Microsoft Solutions Framework – це набір принципів, моделей, дисциплін, концепцій і вказівок для надання послуг інформаційних технологій від Microsoft. MSF не обмежується лише розробкою програм; він також застосовний до інших ІТ-проектів, таких як розгортання, створення мереж або інфраструктурних проектів. MSF не змушує розробника використовувати певну методологію (наприклад, водоспадну модель або гнучку розробку програмного забезпечення).

MSF – включає в себе модель проектної групи (модель команди, MSF Team Model) регламентує ролі учасників проекту, ділянки їх компетенції та

відповідальності, а також містить рекомендації щодо виконання відповідних завдань. Передбачено шість ролей: менеджер з розроблення (program manager), розробник (developer), тестувальник (tester), менеджер релізів (release manager), фахівець з підтримки користувачів (user experience specialist).

Модель процесів (MSF Process Model) – загальна методологія розроблення і впровадження ІТ рішень, що характеризується гнучкістю, рекомендаційним характером і може застосовуватися для різноманітних проєктів. Вона охоплює весь життєвий цикл і поєднує каскадну та спіральну моделі життєвого циклу.

Передбачено такі основні фази процесу розробки:

Розроблення концепції (Envisioning) – визначення складу команди, структури проєкту та бізнес цілей, оцінювання поточної ситуації, створення загальної картини і області дії проєкту, визначення вимог і профілів користувачів, розроблення концепції рішення, оцінювання ризику.

Планування (Planning) – розроблення проєкту та архітектури рішення, створення функціональної специфікації, розроблення планів і календарного графіка, створення середовища розроблення, тестування і пілотної експлуатації.

Розробка (Developing) – створення прототипу додатку, програмних компонентів додатку, рішення (послідовність щоденних або частіших складань), закриття розробки (реалізація всіх функцій, постачання коду і документації).

Стабілізація (Stabilizing) – тестування (тестування компонентів, баз даних, інфраструктури, захисту, інтеграції, аналіз зручності роботи з продуктом, навантажувальне тестування, регресивне тестування, ведення звітності), пілотна експлуатація в тестовому середовищі за участі користувачів і застосування реальних сценаріїв роботи.

Впровадження (Deploying) – розгортання основних компонентів, рішення в цілому, стабілізація, розгортання і передавання в експлуатацію замовнику [26].

Кожна фаза процесу розроблення завершується головною опорною точкою (major milestone) – моментом, коли всі члени команди синхронізують отримані

результати. Призначення таких точок у тому, що вони дозволяють оцінити життєздатність проєкту. Їхні результати видимі не тільки команді розробки, але й замовникові. Після аналізу результатів команда розробників і замовник спільно вирішують, чи можна переходити на наступну фазу. Таким чином, головні опорні точки - це критерії переходу з однієї фази проєкту на іншу.

Ітеративність процесу MSF полягає в його багаторазовому повторенні упродовж усього циклу розробки та існування продукту. На кожній успішній ітерації у продукт включаються тільки ті нові інструменти та функції, які задовольняють постійно змінювані вимоги бізнесу.

Таким чином за загальною орієнтованістю на розробку проєкту під гнучкою методологією Scrum та ітеративними підходами до впровадження ризик менеджменту та адаптивністю до змін саме MSF є одним з найкращих виборів для впровадження.

4.3.2 Дослідження та ідентифікація ризиків

Зазвичай за характером ризику проєкту розподіляють на: технічні, планування та оцінювання, зовнішні, людські, фінансові та економічні, природні ризики. Далі виконується процес ідентифікації ризиків на основі інструментів SWOT, PEST аналізу.

Після чого виконується оцінка ризиків на основі матриці оцінювання ризиків проєкту. І наступним чином відбувається створення плану протиризових заходів та поступове впровадження та профілактики ризиків та оцінювання ризиків експертами.

Першочергово для аналізу зовнішніх впливів в проєкті доцільно використано PEST-аналіз для отримання показників з прийняття рішень. Розбиття на політичні, економічні, соціальні та технічні проблеми допоможе зрозуміти позитивні та негативні впливи, які можуть відбутись з проєктом у посеред процесу виконання таблиця 4.1 [27].

Таблиця 4.1

PEST-аналіз оточення проєкту

Political (Політичні)	Economical (Економічні)
1	2
Проблеми виходу продукту на український ринок через політичну нестабільність під час війни.	Вигідний застарілий курс при конвертації на сервісах та покупці підписок.
Законодавство регулює електронну комерцію, з питань захисту персональних даних покупців та користувачів.	Інфляція та девальвація курсу гривні.
Антимонопольне регулювання проєкту законодавством у випадку монополії на ринку.	Зниження кількості закупок товарів бізнесменами через високу конкуренцію на ринку продажів.
Зміни в правилах митного тарифу та обмеження логістики.	Не бажання покупців чекати доставки товарів з певної країни. Через, що менше бізнесменів займаються продажами.
Ембарго на продаж товарів в певних країнах.	Велика кількість ІТ-спеціалістів на робочому ринку під час економічної нестабільності.
Social (Соціальні)	Technological (Технологічні)
Висока зацікавленість у віддаленому бізнесі з продажів.	Унікальність та універсальність проєктного продукту через технологічний стек.
Пошук підприємцями зручних інструментів для аналізу продажів.	Складність створення серверної архітектури проєкту.
Низька вартість входу в бізнес з продажів Amazon.	Нормалізація, відмовостійкість та оптимізація архітектури бази даних.
Не прив'язаність до демографічного розташування підприємців і можливість торгівлі та доступу до платформи з будь-якої країни.	Складність пошуку кандидатів з потрібним рівнем знань на посади розробників в команді.

Висока складність програмного продукту для певних соціальних груп.	Загрози взлому програмного забезпечення та грошового скаму з транзакцій при налаштуванні платіжних систем.
--	--

За даним поділом аналізу на політичні, економічні, соціальні та технологічні аспекти надалі організовано та проведено експертне оцінювання залученими експертами.

В ролі експертів залучені:

- Власник продукту – експерт №1;
- UI/UX дизайнер – експерт №2;
- Бізнес-аналітик – експерт №3;
- Backend розробник – експерт №4;
- Frontend розробник – експерт №5;
- Проєктний менеджер – експерт №6.

В результаті розроблено таблиці Б.1 – Б.6 з оцінюванням кожного аспекту які винесені окремо в Додаток А.

SWOT-аналіз проєкту також є невід’ємною частиною аналізу проєкту, який було виконано в процесі залучення наукової практики. Де було досліджено головні чинники та фактори проєкту під час SWOT-аналізу які описані в таблиці 4.2.

Таблиця 4.2

SWOT-аналіз проєкту з розвитку електронної комерції

Сильні сторони (Strengths) [28]	Слабкі сторони (Weaknesses) [28]
1	2
Досвід та знання команди в електронній комерції та автоматизації бізнес-процесів.	Відсутність досвіду в роботі команди з Amazon Seller API.

Наявність ресурсів та фінансової підтримки для розробки продукту.	Потенційна залежність від сторонніх сервісів та зміни їх API.
Високий попит на інструменти з автоматизації бізнес-процесів Amazon Seller.	Потреба в ознайомленні всіх членів команди з політикою продажів Amazon.

Продовження табл. 4.2

1	2
Широкий спектр функціонала який може надати готовий продукт.	Великі затрати часу на розробку ідеальної Backend архітектури продукту.
Велика кількість джерел та ідей для розробки документації продукту.	Великі затрати та складність розробки графіків та застосування фільтрів.
Можливості (Opportunities) [28]	Загрози (Threats) [28]
Ріст ринку електронної комерції та популярність Amazon Seller серед продавців, які бажають займатись підприємницькою діяльністю по всьому світу.	Можливі зміни в політиці продажів, зберігання товарів та використання Amazon Seller.
Можливість розширення Amazon Seller після війни на ринку українських підприємців.	Поява нових SaaS платформ конкурентів у сфері аналітики продажів Amazon.
Зростання кількості продавців, що шукають інструменти для оптимізації продажів та підвищення ефективності бізнесу.	Ризик нестабільності економічної та політичної ситуації в країнах, де знаходяться цільові аудиторії.
Зростання кількості продавців, що шукають інструменти для оптимізації продажів та підвищення ефективності бізнесу.	Ризики грошового скаму сервісу та хакінгу продукту.
Можливість розширення нового функціоналу програмного продукту	Активні зміни та потреби користувачів платформи у нових модулях та інструментах аналізу.

після його випуску на ринок користувачів.	
---	--

Частково відштовхуючись від SWOT-аналізу було використано матрицю ризиків проекту. В свою чергу матриця поділяється на дві таблиці. В першій таблиці 4.3, сформовано матрицю оцінювання ризиків проекту з описом імовірності ризиків позначені літерами в діапазоні А-Е та серйозність небезпеки виражена у цифровому діапазоні від 1 - 4. А в другій безпосередня оцінка ризиків та стратегія керування таблиця 4.4 [29, ст. 33].

Таблиця 4.3

Матриця оцінювання ризиків

	Катастрофічна (1)	Критична (2)	Гранична (3)	Незначна (4)
1	2	3	4	5
Часта (А)	1А	2А	3А	4А
Можлива (В)	1В	2В	3В	4В
Випадкова (С)	1С	2С	3С	4С
Віддалена (D)	1D	2D	3D	4D
Неймовірна (E)	1E	2E	3E	4E

Таблиця 4.4

Результати оцінювання та керування ризиками

№	Ризик	Оцінка	Стратегії керування ризиками
1	2	3	4
1	Можливі зміни в політиці продажів, зберігання товарів та використання Amazon Seller.	3В	Моніторинг змін у політиці ведення бізнесу та доповідання змін від керівника команди.

2	Поява нових SaaS платформ конкурентів у сфері аналітики продажів Amazon.	2B	Перегляд змін на ринку електронної комерції під Amazon. Впровадження розробки конкурентоспроможного функціонала.
3	Ризик нестабільності економічної та політичної ситуації в країнах, де знаходяться цільові аудиторії.	2A	Впровадження маркетингових рекомендацій, маніпуляції зі зменшення вартості підписки на сервіс.

Продовження табл. 4.4

1	2	3	4
4	Ризики грошового скаму сервісу та хакінгу продукту.	2C	Перегляд Backend розробником вразливостей сервісу.
5	Активні зміни та потреби користувачів платформи у нових модулях та інструментах аналізу.	3D	Проведення аудитів та опитування потреб в аудиторії. Запровадження розробки нового функціоналу.
6	Проблеми з енергопостачанням та інтернетом під час розробки ПЗ.	1B	Закупка керівництвом генераторів, акумуляторів, ноутбуків, безперебійників живлення. Можливість використовувати асинхронний 8-годинний робочий графік.
7	Проблеми зі змінами в Amazon Seller API для розробників.	1D	Перегляд технічних змін компанією Amazon та документування для ознайомлення розробників.

Отже, за підготовлені матеріали свідчать про важливість систематичного та глибокого оцінювання потенційних небезпек та загроз, які можуть виникнути протягом виконання проєкту. Проведений аналіз дозволяє ідентифікувати ключові

ризиків, класифікувати їх за ступенем впливу та ймовірністю виникнення, а також розробляти стратегії мінімізації або управління ними.

Надзвичайно важливим є ретельне планування та реалізація заходів з ризик-менеджменту, щоб забезпечити стабільний хід проєкту та досягнення його цілей в умовах невизначеності та змін. Ретельний аналіз ризиків є необхідною складовою успішної реалізації будь-якого проєкту, допомагаючи забезпечити його ефективність, стабільність та відповідність поставленим завданням і цілям.

4.3.3 Розробка стратегії управління ризиками

Етапи та загальна стратегія управління ризиками в проєкті виглядає наступним чином. Після процесів ідентифікації ризиків, детального аналізу причин їх можливої появи було впроваджено класифікацію ризиків та їх наступний поділ на ризики постійного характеру та тимчасового характеру.

На основі поданої інформації команда проєкту на чолі з проєктним менеджером та стейкхолдерами, почала реагувати і планувати стратегії запобігання появи ризиків. Додатково плануючи час співробітників для вивчення нових освітніх матеріалів пов'язаних з технологічним стеком проєкту і заглиблення та пізнання всієї необхідної інформації для безперешкодної розробки проєкту, включаючи політику використання і процеси роботи з платформою Amazon. Також було додатково впроваджено аудити з кібербезпеки для уникнення проблем втрати інформації.

У випадку ризиків постійного характеру керівництво проєкту додатково виділяє додаткові суми коштів для їх покриття максимально мінімізуючи негативний вплив. Такі проблеми можуть виражатись, наприклад в потребі купити співробітнику технічний засіб задля продовження процесу виконання проєкту який має фіксований вид витрат. Або готовність до появи нового конкурента на ринку, яка потребує додаткових маркетингових та функціонально-технічних впроваджень, щоб бути максимально конкурентоспроможним.

Отже, такий моніторинг ризиків, протиризикові процеси та аудити, дійсно гарно проявляють себе в умовах робіт та загальному закінченню робіт в компанії.

4.4 Оцінювання якості виконаних робіт

Будь-яка робота після виконання має бути оцінена відповідно до встановлених процесів у проєкті. Оцінювання якості виконаних робіт в рамках ІТ проєкту є критичним етапом, спрямованим на визначення відповідності результатів проєкту вимогам та стандартам якості.

Перш за все, оцінка якості виконаних робіт важлива для забезпечення відповідності результатів проєкту вимогам замовника. ІТ проєкт зазвичай має чіткі технічні специфікації та функціональні вимоги, які повинні бути виконані в рамках виконаних задач. Оцінка якості допомагає перевірити, чи були ці вимоги виконані вірно задля цього.

Крім того, оцінка якості допомагає виявити та виправити будь-які дефекти та недоліки у продукті до його впровадження. Це сприяє підвищенню надійності та ефективності системи, а також забезпечує задоволення потреб та очікувань користувачів.

Оцінювання якості задач виконується двома спеціалістами перший з яких проєктний менеджер і другий QA-тестувальник. Після виконання задачі її виконавець доповідає про завершення та перенесення її на тестування. В свою чергу за наступної можливості тестувальник бере задачу на ревізію, переглядаючи її початковий опис та задум. Ознайомившись з завданням й описом, тестувальник переходить до переглядів критеріїв прийому виконаної задачі, які визначає власник продукту в нашому випадку це замовник. Таким чином виконана задача має мати максимальну відповідність до початково закладеного технічного завдання. І тестувальник проводить функціональні тести інтерфейсу, рецензування коду з пошуком наявних помилок при розробці, а також застосовує автоматизовані засоби

аналізу коду. Після підтвердження проходження тестування якості, задачу переглядає проєктний менеджер та вносить зміни в статус задачі записуючи час її виконання.

4.5 Аналіз фінансових витрат та показників ефективності

Затрати матеріальних й часових ресурсів проєкту, а також зусилля членів команди та стейкхолдерів є вирішальним аспектом загальної успішності проєкту. Аналіз фінансових витрат та показників ефективності в ІТ проєктах є важливою складовою для визначення ступеня виконання проєкту з погляду фінансової дисципліни та ефективності використання ресурсів.

Для аналізу фінансових витрат ми використовуємо інструмент MS Project та фінансові показники. Cash Flow, Cost і Cumulative Cost - це ключові фінансові показники, які використовуються для вимірювання фінансової продуктивності та витрат в рамках проєкту.

Cash Flow (потік грошей) - це показник, який відображає рух грошових коштів в рамках проєкту протягом певного періоду часу. Cash Flow включає в себе всі витрати та доходи, що здійснюються протягом цього періоду. Важливою характеристикою потоку грошей є те, що він враховує реальні грошові рухи, тобто фактичні виплати та отримання коштів.

Cost (витрати) - наступний показник, демонструє суму грошей, витрачених на виконання певного виду робіт або завершення певної фази проєкту. Витрати включають в себе зарплату працівників, послуги та інші витрати, пов'язані з виконанням проєкту.

Cumulative Cost (кумулятивні витрати) - це сума всіх витрат, які були здійснені на проєкт з моменту його початку до певного моменту часу. Cumulative Cost відображає загальні фінансові витрати на проєкт, включаючи всі попередні витрати та нові витрати, здійснені до поточного моменту.

Ці фінансові показники є важливими для керівництва проєктом та стейкхолдерів, оскільки вони допомагають відстежувати фінансовий стан проєкту,

приймати управлінські рішення та забезпечувати фінансову стабільність та успішне завершення проєкту.

Ознайомитись з результатами графіку грошового потоку проєкту можна на наступній ілюстрації (рис. 4.4).

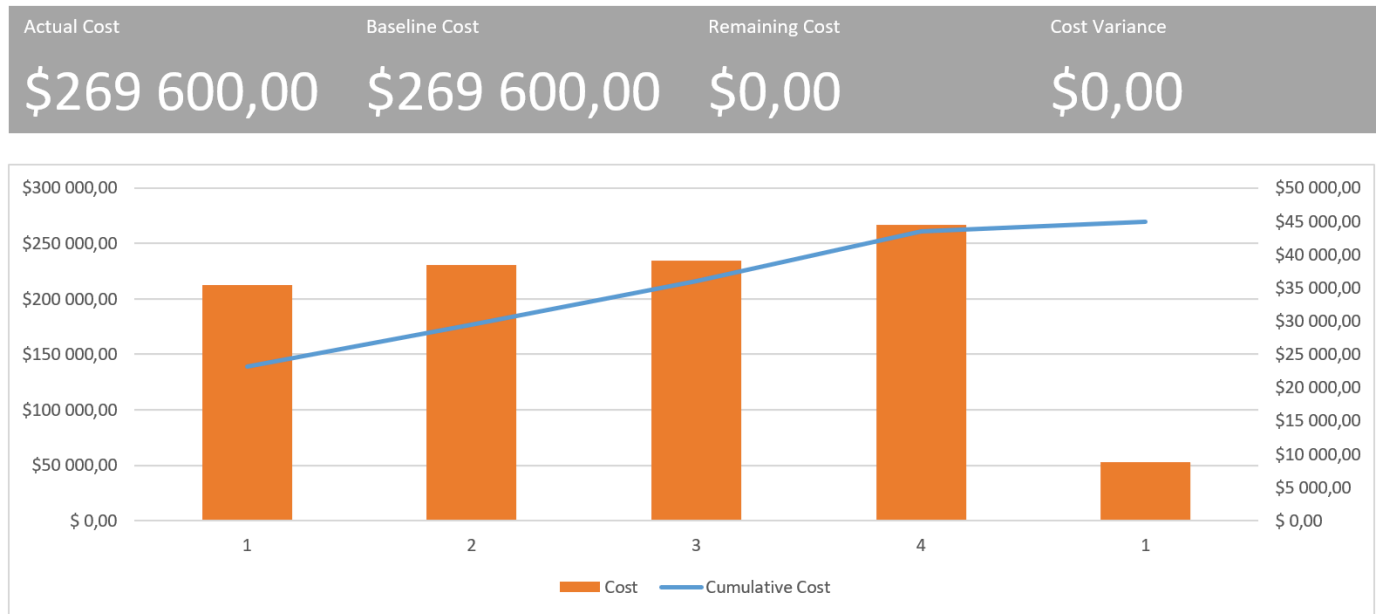


Рис. 4.4. Графік грошового потоку проєкту

Далі наступним чином ми можемо порівняти фактичні та планові показники проєкту які виражені у наступних показниках.

Actual Cost of Work Performed (ACWP) - фактична вартість виконаних робіт з наростаючим підсумком відображається в часі для кожного звітного періоду [30, ст. 147]. Він враховує тільки ті витрати, які були фактично здійснені на завершення певних етапів робіт.

Budgeted Cost of Work Scheduled (BCWS) - планова вартість запланованих робіт або плановий обсяг зображує бюджет з наростаючим підсумком, який відображено в часі, коли передбачається робити витрати згідно з планом проєкту [30, ст. 147]. Він використовується для порівняння з фактичними витратами та визначення відхилень від плану.

Budgeted Cost of Work Performed (BCWP) - планова вартість виконаних робіт або освоєний обсяг, наступний ключовий показник відображає заплановані витрати на виконання робіт, які фактично були виконані до певного моменту часу [30, ст. 148].

Побачити результати зведення графіку планових та поточних затрат на проєкт можна на наступному графіку (рис. 4.5).

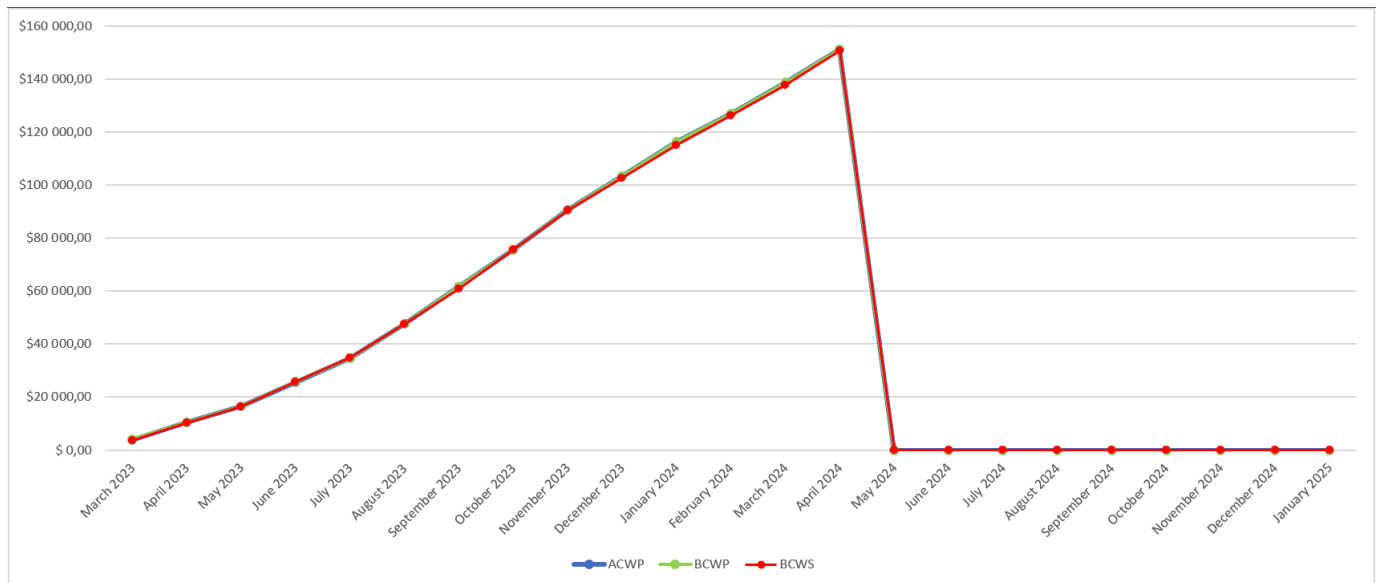


Рис. 4.5. Графік ACWP, BCWP та BCWS

Як можна поглянути, то поточні ресурси майже не відрізняються від запланованих, з цього можна зрозуміти що проєкт виконується за планом станом на квітень 2024 року. В переліку важливих графіків можна відзначити графіки робочих ресурсів, їх показники затрачених годин на проєкті, роботи яка залишилась для виконання і відсоток завершеності робіт. І також порівняти скільки годин спеціалісти у порівнянні один з одним витратили на проєкті, демонстрація графіків ресурсів проілюстровано на (рис. 4.6).

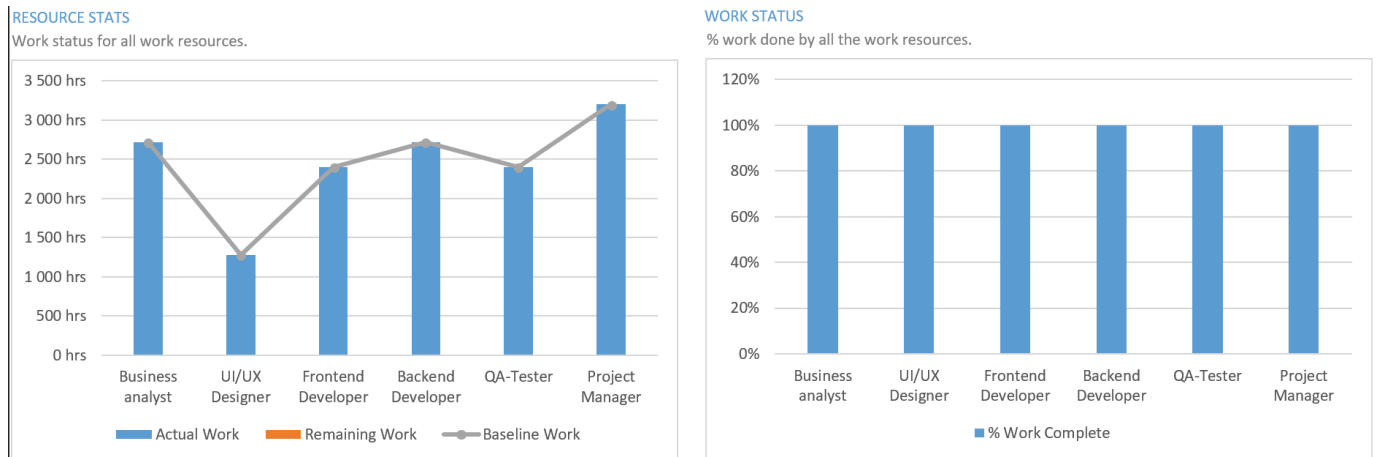


Рис. 4.6. Графіки ресурсів та статусів завершеності робіт по ролях

Доповненням до вищезгаданих графіків є зведені показники вартості ресурсів де можна побачити погодинну вартість кожного співробітника та фактичні виконані об'єми робіт в годинах (рис. 4.7).

Name	Actual Work	Actual Cost	Standard Rate
Business analyst	2 720 hrs	\$40 800,00	\$15,00/hr
UI/UX Designer	1 280 hrs	\$12 800,00	\$10,00/hr
Frontend Developer	2 400 hrs	\$48 000,00	\$20,00/hr
Backend Developer	2 720 hrs	\$68 000,00	\$25,00/hr
QA-Tester	2 400 hrs	\$36 000,00	\$15,00/hr
Project Manager	3 200 hrs	\$64 000,00	\$20,00/hr

Рис. 4.7. Деталі вартості робочих ресурсів

Використання цих показників допомагає керівництву проєкту зрозуміти, як ефективно використовуються фінансові ресурси, контролювати витрати та приймати рішення щодо подальшого управління проєктом.

Розділ аналізу фінансових витрат та показників ефективності зосереджується на важливості ефективного управління ресурсами проєкту та вимірюванні його фінансової продуктивності. Фінансові показники, такі як Cash Flow, Cost і Cumulative

Cost, відіграють ключову роль у визначенні ступеня виконання проєкту з фінансової точки зору та ефективності використання ресурсів.

Використання інструментів, таких як MS Project, для аналізу фінансових витрат дозволяє ефективно відстежувати фінансовий стан проєкту та приймати управлінські рішення. Порівняння фактичних та запланованих витрат, а також використання ресурсів, допомагає керівництву зрозуміти, наскільки ефективно використовуються ресурси та вчасно реагувати на можливі зміни.

ВИСНОВКИ

В ході виконання наукової магістерської роботи було проведено комплексне дослідження, що охоплює аспекти розробки ІТ-проєкту з електронної комерції для потреб підприємців які займаються продажами на платформі Amazon та потребують надійного інструменту для автоматизації та масштабування свого бізнесу.

Дослідження проєкту демонструє високий рівень зацікавленості потенційних користувачів, а також наукову інноваційність яка полягає в рівні автоматизації бізнес-процесів, що забезпечують повний цикл ведення бізнесу починаючи від закупок товарів та інвентаризації закінчуючи детальною фінансовою аналітикою на основі фінансових і маркетингових показників. Не менш важливим внеском в наукову діяльність є розроблення модуля який відповідає за аналіз та генерацію рекомендацій на основі підключених регіональних магазинів користувачів та штучного інтелекту. Така комбінація дає можливість відкрити успішний бізнес навіть особам які займаються цим вперше.

Дипломна робота поділяється на чотири детально описані розділи, перший присвячений дослідженню та аналізу проєкту. При розробці проєкту було проаналізовано сучасні проєкти з електронної комерції які мають попит. Обрано сферу та сформовано концептуальну тематику проєкту. В результаті проведеного дослідження та аналізу проєкту, було отримано глибоке розуміння ключових аспектів його розвитку та реалізації. Бізнес-аналіз проєкту дозволив чітко визначити сферу діяльності продукту, виявити конкурентні переваги та встановити цілі, спрямовані на досягнення успішного впровадження. Дослідження стейкхолдерів підкреслило важливість взаємодії з різними зацікавленими сторонами для забезпечення успішного розвитку проєкту.

Дослідження документів та артефактів проєкту дало можливість зрозуміти основні етапи реалізації та визначити потрібні ресурси для його успішної реалізації. Вибір стандартів та технологій менеджменту був здійснений з урахуванням вимог

проекту та специфіки його реалізації, що дозволило ефективно організувати та контролювати процеси в рамках проекту. Оцінка інвестиційної привабливості проекту підтвердила його потенціал для отримання прибутковості та визначила стратегічні кроки для максимізації вигоди від інвестицій.

У другому розділі було проведено детальне планування ключових аспектів реалізації проекту. Створення життєвого циклу проекту дало можливість чітко визначити етапи його виконання та встановити необхідні критерії успіху на кожному етапі. Побудова структури проекту та проектної команди була здійснена з урахуванням специфіки завдань та вимог проекту, що дозволило ефективно розподілити компетентності, ролі та зони відповідальності серед учасників команди. Формування бюджету та оцінювання ресурсів було здійснене з метою забезпечення ефективного використання фінансових та людських ресурсів проекту. Планування віх та WBS проекту дозволило чітко визначити завдання та визначити послідовність їх виконання, що сприяло забезпеченню структурованості та організованості в ході виконання проекту. Календарний план та таймлайни проекту були розроблені для управління часовими ресурсами та встановлення реалістичних термінів виконання завдань.

В третьому розділі описані процеси виконання проекту в кожній зі сфер діяльностей членів проекту. Також проаналізовано сучасні інструменти проєктування, дизайну й розробки які надають можливість команді виконати всі необхідні роботи вчасно та з достойним рівнем якості.

Четвертий розділ був направлений менеджмент й контроль виконання проекту за вказаним бюджетом. Менеджмент описаний повною мірою з врахуванням ідентифікації ризиків і застосування різних інструментів кількісного та якісного дослідження.

Загалом, результати проведеного аналізу створюють надійну базу для подальшого розвитку та вдосконалення проекту, а також можуть послужити основою для подальших досліджень у сфері управління проєктами та бізнес-аналізу.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Адміністрація порталу "Букліб", "Дерево цілей" підприємства, 2024 р. [Електронний ресурс]: <https://buklib.net/books/25617/>
2. Schwaber K., & Sutherland J., "Посібник Scrum. Scrum.org", 2020р. [Електронний ресурс]: <https://scrumguides.org/scrum-guide.html>
3. Редактори НТУ «Дніпровська політехніка», "Чиста теперішня вартість (ЧТВ) інвестиційного проєкту", 2024р. [Електронний ресурс]: https://eaf.nmu.org.ua/ua/Students/%D0%A0%D0%95%D0%9A%D0%9E%D0%9C%D0%95%D0%9D%D0%94%D0%90%D0%A6%D0%A1%D0%90%D0%9C%D0%9E%D0%A1%D0%A2%D0%99%D0%9D_I%D0%BD%D0%B2%D0%B5%D1%81%D1%82%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F.pdf
4. Шевченко Т., "Стейкхолдери проєкту: хто такі та чому важливо налагодити з ними комунікацію", 2023р. [Електронний ресурс]: <https://wizeclub.education/blog/stejkholderi-projektu-hto-taki-ta-chomu-vazhливо-nalagoditi-z-nimi-komunikatsiyu/>
5. Blake J., "Project managing the SDLC", 2024р. URL: <https://www.pmi.org/learning/library/project-managing-sdlc-8232>
6. Адміністрація сайту національного університету «Острозька Академія», "Структуризація проєкту", 2024р. [Електронний ресурс]: https://www.oa.edu.ua/download/Lektsija_5.PDF
7. Зосим М., "Матриця відповідальності (RACI Matrix)", 2022р. [Електронний ресурс]: <https://www.maxzosim.com/raci-matrix/>
8. Редакція порталу Microsoft Learn, "Огляд робочих структур проєкту", 2023р. [Електронний ресурс]: <https://learn.microsoft.com/uk-ua/dynamics365/project-operations/prod-pma/work-breakdown-structures>

9. Зосим М., “Структура розбиття робіт (Work Breakdown Structure - WBS)”, 2023р. [Електронний ресурс]: <https://www.maxzosim.com/struktura-rozbittia-robot/>
10. Редактори освітнього порталу «JA Europe», “Розділ 3: Планування проєкту”, 2024р. [Електронний ресурс]: <https://elearn.jaeurope.org/pm/ua/>
11. Освітній портал державного університету «Житомирська політехніка», “Основи планування проєкту. Структуризація проєкту”, 2020р. [Електронний ресурс]: https://learn.ztu.edu.ua/pluginfile.php/255463/mod_folder/content/0/Agile%20Project%20Management/%D0%9B%D0%B5%D0%BA%D1%86%D1%96%D1%8F%204.PPTX?forcedownload=1
12. Чумаченко М. Г., Шведа Н., “Особливості календарного планування проєкту”, 2017р. [Електронний ресурс]: https://elartu.tntu.edu.ua/bitstream/lib/21362/2/IRSP_2017_Shveda_N-Features_of_the_project_scheduling_69-70.pdf
13. Адміністрація освітнього порталу «Pidru4niki», “Календарне планування проєкту”, 2024 р. [Електронний ресурс]: https://pidru4niki.com/74426/ekonomika/kalendarne_planuvannya_proektu
14. Інститут Управління Проєктами «PMI», “НАСТАНОВА РМВОК сьоме видання та стандарт з управління проєктами”, 2024 р. [Електронний ресурс]: <https://pmiukraine.org/pmbok7>
15. Команда підтримки корпорації «Microsoft», “Перегляд завантаженості та доступності ресурсів у класичній програмі Project”, 2021р. [Електронний ресурс]: <https://support.microsoft.com/en-us/office/view-resource-workloads-and-availability-in-project-desktop-3ee16869-68ad-4e63-bcb3-278ae34f7459>

16. Келлі Д., «Початок роботи з Jira, Confluence та скрам фреймворком», 2024 р.
[Електронний ресурс]:
<https://www.atlassian.com/agile/scrum/jira-confluence-scrum>
17. Адміністрація порталу «Atlassian», «Основи роботи з Confluence», 2024р.
[Електронний ресурс]:
<https://www.atlassian.com/software/confluence/resources/guides/get-started/overview#about-confluence>
18. Адміністрація порталу «Atlassian», «Jira для команд», 2024р. [Електронний ресурс]:
<https://www.atlassian.com/software/jira/guides/getting-started/who-uses-jira#for-project-management-teams>
19. Побережняк А., «Що таке Gitlab», 2023р. [Електронний ресурс]:
<https://cloudfresh.com/ua/cloud-blog/shho-take-gitlab/>
20. Юшко О., «Що таке Docker: простими словами про контейнеризацію», 2020 р.
[Електронний ресурс]:
<https://blog.ithillel.ua/articles/shcho-take-docker-prostimi-slovami-pro-konteynerizatsiyu>
21. Шенкс М., «Вступ до дизайну мислення посібник з процесу», 2024р.
[Електронний ресурс]:
<https://web.stanford.edu/~mshanks/MichaelShanks/files/509554.pdf>
22. Адміністрація сайту «Харківського національного економічного університету імені Семена Кузнеця», «Лекція 3 Реляційні бази даних» 2024 р. [Електронний ресурс]:
https://pns.hneu.edu.ua/pluginfile.php/169738/mod_folder/content/0/%D0%9B%D0%B5%D0%BA%D1%86%D1%96%D1%8F%203%20%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D1%96%D0%B9%D0%BD%D1%96%20%D0%B1%D0%B0%D0%B7%D0%B8%20%D0%B4%D0%B0%D0%BD%D0%B8%D1%85.docx?forcedownload=1

23. Адміністрація віртуального університету «ЛДУ БЖД», “Лекція. Середовище баз даних”, 2024р. [Електронний ресурс]:
https://virt.ldubgd.edu.ua/pluginfile.php/38818/mod_resource/content/1/2.%20%D0%A1%D0%B5%D1%80%D0%B5%D0%B4%D0%BE%D0%B2%D0%B8%D1%89%D0%B5%20%D0%B1%D0%B0%D0%B7%20%D0%B4%D0%B0%D0%BD%D0%B8%D1%85.pdf
24. Морозов В.В., “Постановка задачі прийняття управлінських рішень. Технології прийняття рішень”, 2024р. [Електронний ресурс]:
https://drive.google.com/file/d/1BOPTpPckYBFo8o4npUyB66TLj7G_jggG/view
25. Девідсон Д., “Чому ми використовуємо сторі поінти для оцінювання?”, 2014р. [Електронний ресурс]:
<https://www.scrum.org/resources/blog/why-do-we-use-story-points-estimating>
26. Редакція порталу «Посібники статті та книги для студентів», “Автоматизована підтримка методології Microsoft Solution Framework”, 2022р. [Електронний ресурс]:
<https://posibniki.com.ua/post-avtomatizovana-pidtrimka-metodologiiio-microsoft-solutions-framework>
27. Адміністрація студентської бібліотеки «Букліб», “PEST-аналіз мікросередовища підприємства”, 2024 р. [Електронний ресурс]:
<https://buklib.net/books/36596/>
28. Оголь А., “SWOT-аналіз із прикладами”, 2021р. [Електронний ресурс]:
<https://esputnik.com/uk/blog/swot-analiz-iz-prikladami>
29. Касьяненко В., Дубініна О., “Управління та аналіз ризиків у відповідності до вимог ДСТУ EN ISO 13485 та ДСТУ EN ISO 14971”, 2021 р. [Електронний ресурс]:
<https://www.slideshare.net/ImproveMedical/en-iso-13485-en-iso-14971-14052021>
30. Добровська Л. М., Аверьянова О. В. комп'ютерний практикум «КПІ», “Управління ІТ-проєктами в Microsoft Project”, 2020 р. [Електронний ресурс]:

<https://ela.kpi.ua/server/api/core/bitstreams/1546697f-b66b-41e3-bfb1-7ceebd1fd7d7/content>

ДОДАТОК А

```
1 <?php
2
3 namespace App\Transformers;
4
5 use App\Models\StoreProductDailyReport;
6
7 class DashboardStoresTransformer extends BaseDashboardTransformer
8 {
9     public function transform(StoreProductDailyReport $response) {
10         $stores = $response->stores;
11
12         return [
13             'currency' => $response->currency,
14             'stores' => [
15                 'items' => empty($stores) ? null :
16                     self::returnStores($stores, $response->time),
17             ],
18         ];
19     }
20
21     private function returnStores($data, $time)
22     {
23     |
24     {
25         $stores = [];
26         foreach ($data as $id => $store) {
27             $stores[] = [
28                 'id' => $store->hash_id,
29                 'asin' => $store->asin,
30                 'marketplace_code' => $store->marketplace_code,
31                 'store_name' => $store->name,
32                 'currency_symbol' => $store->marketplace('currencySymbol'),
33                 'current' => [
34                     'from' => $time['from_date'],
35                     'to' => $time['to_date'],
36                     'revenue' => round($store->dashboard_data['revenue'], 2),
37                     'cost' => round($store->dashboard_data['cost'], 2),
38                     'net_profit' => round($store->dashboard_data['net_profit'], 2),
39                     'refunds' => round($store->dashboard_data['refunds'], 2),
40                     'units_sold' => $store->dashboard_data['units_sold'],
41                 ],
42                 'previous' => empty($previousData[$id]) ? null : [
43                     'from' => $time['previous_from_date'],
44                     'to' => $time['previous_to_date'],
45                     'revenue' => round($store->previous_dashboard_data['revenue'], 2),
46                     'cost' => round($store->previous_dashboard_data['cost'], 2),
47                     'net_profit' => round($store->previous_dashboard_data['net_profit'], 2),
48                     'refunds' => round($store->previous_dashboard_data['refunds'], 2),
49                     'units_sold' => $store->previous_dashboard_data['units_sold'],
50                 ],
51             ];
52         }
53
54         return $stores;
55     }
56 }
```

Рис. А.1. Laravel трансформація даних магазинів для панелі керування

```

1 |<?php
2
3 | namespace App\Mail;
4
5 | use Illuminate\Bus\Queueable;
6 | use Illuminate\Mail\Mailable;
7 | use Illuminate\Queue\SerializesModels;
8
9 | class TestEmail extends Mailable
10 | {
11 |     use Queueable, SerializesModels;
12
13 |     private $fromAddress;
14 |     private $fromName;
15 |     private $message;
16
17 |     /**
18 |      * Create a new message instance.
19 |      *
20 |      * @return void
21 |      */
22 |     public function __construct($fromAddress, $fromName, $subject, $message)
23 |     {
24 |         $this->fromAddress = $fromAddress;
25 |         $this->fromName = $fromName;
26 |         $this->subject = $subject;
27 |         $this->message = $message;
28 |     }
29
30 |     /**
31 |      * Build the message.
32 |      *
33 |      * @return $this
34 |      */
35 |     public function build()
36 |     {
37 |         return $this->from($address = $this->fromAddress, $name = $this->fromName)
38 |             ->subject($this->subject)
39 |             ->view('emails.test')
40 |             ->with(['email_message' => $this->message]);
41 |     }
42 | }

```

Рис. А.2. Скрипт відправлення електронних листів за допомогою PHP та Laravel

```

1 {
2   "private": true,
3   "scripts": {
4     "dev": "npm run development",
5     "development": "cross-env NODE_ENV=development node_modules/webpack/bin/webpack.js --progress --hide-modules --config",
6     "watch": "npm run development -- --watch",
7     "watch-poll": "npm run watch -- --watch-poll",
8     "hot": "cross-env NODE_ENV=development node_modules/webpack-dev-server/bin/webpack-dev-server.js --inline --hot --con",
9     "prod": "npm run production",
10    "production": "cross-env NODE_ENV=production node_modules/webpack/bin/webpack.js --no-progress --hide-modules --confi",
11  },
12  "devDependencies": {
13    "axios": "^0.19",
14    "bootstrap": "^4.0.0",
15    "cross-env": "^7.0",
16    "jquery": "^3.2",
17    "laravel-mix": "^5.0.1",
18    "lodash": "^4.17.13",
19    "popper.js": "^1.12",
20    "resolve-url-loader": "^2.3.1",
21    "sass": "^1.20.1",
22    "sass-loader": "^8.0.0",
23    "vue": "^2.5.17",
24    "vue-template-compiler": "^2.6.10"
25  }
26 }

```

Рис. А.3. JSON пакет управління залежностями та скриптами проєкту

```

1 <?php
2
3 namespace App\Console\Commands;
4
5 use Illuminate\Console\Command;
6 use Illuminate\Support\Facades\Artisan;
7
8 class Migrate extends Command
9 {
10    protected $signature = 'migrate:version';
11
12    protected $description = 'Migrate database by app version';
13
14    public function __construct()
15    {
16        parent::__construct();
17    }
18
19    public function handle()
20    {
21        $appVersion = config('app')['version'];
22
23        $command = 'migrate --path=database/migrations/' . $appVersion;
24        echo 'Command is: ' . $command . "\n";
25
26        Artisan::call($command);
27        if (isTest()) {
28            echo "\n*** Migrated ***\n";
29        } else {
30            echo Artisan::output();
31        }
32
33        return 0;
34    }
35 }

```

Рис. А.4. Міграція БД на основі версії додатку за допомогою Artisan Laravel

ДОДАТОК Б

Таблиця Б.1

Експертна оцінка впливу політичних факторів

Фактори впливу політичного середовища	Характер впливу фактору	Оцінка експертів на ступінь впливу (max=10)						Середній бал
		Екс. №1	Екс. №2	Екс. №3	Екс. №4	Екс. №5	Екс. №6	
1	2	3	4	5	6	7	8	9
Проблеми виходу продукту на український ринок	-	4	4	5	4	3	4	-4
Регулювання е-commerce персональних даних користувачів	+	2	2	3	3	3	2	+2.5
Антимонопольне регулювання	-	6	7	7	8	7	6	-6.83
Зміни на митниці та обмеження логістики	-	4	5	4	6	6	5	-5
Ембарго на продаж продуктів в певних країнах	-	7	8	6	5	6	7	-6.5

Оцінювання впливу економічних факторів

Фактори впливу економічного	Характер впливу фактору	Оцінка експертів на ступінь впливу (max=10)						Середній бал
		Екс. №1	Екс. №2	Екс. №3	Екс. №4	Екс. №5	Екс. №6	
1	2	3	4	5	6	7	8	9
Вигідний застарілий курс при конвертації на сервісах та покупці підписок	+	5	6	5	6	4	5	+5.16
Інфляція та девальвація курсу гривні	-	4	3	3	5	3	4	-3.66
Зниження закупок товарів бізнесменами	-	6	6	5	6	6	4	-5.5
Небажання очікування покупців довгої міжрегіональної доставки товарів	-	2	3	3	4	2	2	-2.66
Велика кількість ІТ-спеціалістів під час економічної нестабільності	+	7	8	8	7	8	6	+7.33

Оцінювання впливу соціальних факторів

Фактори соціального впливу	Характер впливу фактору	Оцінка експертів на ступінь впливу (max=10)						Середній бал
		Екс. №1	Екс. №2	Екс. №3	Екс. №4	Екс. №5	Екс. №6	
1	2	3	4	5	6	7	8	9
Висока зацікавленість у віддаленому бізнесі з продажів	+	7	9	7	8	8	8	+7.83
Пошук підприємцями зручних інструментів аналізу продажів	+	7	8	8	6	6	7	+7
Низька вартість входу в бізнес з продажів Amazon	+	4	6	5	6	7	4	+5.33
Можливість міжрегіональної торгівлі	+	8	8	6	7	6	6	+6.83
Висока складність програмного продукту для певних соціальних груп	-	2	3	2	2	1	1	-1.83

Таблиця Б.4

Оцінювання впливу технологічних факторів

Фактори технологічного впливу	Характер впливу фактору	Оцінка експертів на ступінь впливу (max=10)						Середній бал
		Екс. №1	Екс. №2	Екс. №3	Екс. №4	Екс. №5	Екс. №6	
1	2	3	4	5	6	7	8	9
Унікальність та універсальність проектного продукту через технічний стек	+	6	6	8	7	8	8	+7.16
Складність створення серверної архітектури	-	4	4	5	5	5	5	-4.66
Нормалізація, відмовостійкість та оптимізація БД	+	6	6	7	7	8	7	+6.83
Складність пошуку кандидатів з потрібним рівнем знань	-	4	4	5	4	3	3	-3.83
Загрози взлому ПЗ та скаму	-	6	5	5	7	7	6	-6

Найбільші впливи PEST

Політичні		Економічні	
1	2	3	4
Фактор	Вага	Фактор	Вага
Антимонопольне регулювання	-6.83	Зниження закупок товарів бізнесменами	-5.5
Регулювання e-commerce персональних даних користувачів	+2.5	Велика кількість ІТ-спеціалістів під час економічної нестабільності	+7.33
Соціальні		Технологічні	
Висока складність програмного продукту для певних соціальних груп	-1.83	Загрози взлому ПЗ та скаму	-6
Висока зацікавленість у віддаленому бізнесі з продажів	+7.83	Унікальність та універсальність проєктного продукту через технічний стек	+7.16

Впровадження управлінських рішень в проєкті

Фактори	Зміни в галузі	Зміни в організації	Дії
1	2	3	4
Політичні	1. Антимонопольне регулювання на ринку.	1. Формування антимонопольних комітетів.	1. Моніторинг стану та реформ комітетів.
	2. Регулювання e-commerce персональних даних користувачів.	2. Перевірка надійності даних клієнтів.	2. Запровадження сертифікацій програмного забезпечення.
Економічні	1. Зниження закупок товарів бізнесменами.	1. Зменшення оборту ринку продажів на Amazon.	1. Запровадження знижок та можлива маркетингова кампанія для продукту.
	2. Велика кількість IT-спеціалістів під час економічної нестабільності	2. Більший вибір якісного IT-спеціаліста.	2. Збільшення часу роботи HR-менеджера на пошук якісного співробітника.
Соціальні	1. Висока складність програмного продукту для певних соціальних груп.	1. Постійний розвиток населення за допомогою IT заходів і посібників.	1. Розробка інтуїтивно зрозумілих UX інструкцій та інтеграцій в продукті.

Продовження табл. Б.6

1	2	3	4
	2. Висока зацікавленість у віддаленому бізнесі з продажів.	2. Збільшення кількості бізнесменів Amazon.	2. Забезпечення надійної та швидкої роботи проєктного продукту.
Технологічні	1. Загрози взлому ПЗ та скаму.	1. Постійні кібератаки на ресурси.	1. Запровадження заходів з кібербезпеки.
	2. Унікальність та універсальність проєктного продукту через технологічний стек.	2. Постійні зміни в технологічному стеку	2. Відведення часу команди на вивчення змін в технологічному стеку.