

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки  
Факультет Комп'ютерних наук  
(повна назва)  
Кафедра Інформаційних управляючих систем  
(повна назва)

## КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти другий (магістерський)  
Дослідження методів пріоритезації вимог до ІТ-продуктів  
(тема)

Виконав:  
студент 2 курсу, групи УПГІТм-21-1  
Костянтин КРАВЧИК  
(власне ім'я, прізвище)


Спеціальність 122 Комп'ютерні  
науки  
(код і повна назва спеціальності)

Тип програми освітньо-наукова  
(освітньо-професійна або освітньо-наукова)

Освітня програма Управління проєктами  
в галузі інформаційних технологій  
(повна назва освітньої програми)

Керівник проф. каф. ІУС Максим ЄВЛАНОВ  
(посада, власне ім'я, прізвище)

Допускається до захисту  
Зав. кафедри

  
(підпис)

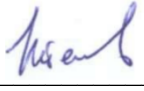
Костянтин ПЕТРОВ  
(власне ім'я, прізвище)

2023 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук  
Кафедра Інформаційних управляючих систем  
Рівень вищої освіти другий (магістерський)  
Спеціальність 122 Комп'ютерні науки  
(код і повна назва)  
Тип програми освітньо-наукова  
(освітньо-професійна або освітньо-наукова)  
Освітня програма Управління проєктами в галузі інформаційних технологій  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри 

(підпис)

« 03 » квітня 20 23 р.

**ЗАВДАННЯ**  
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Кравчику Костянтину Володимировичу  
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів пріоритезації вимог до ІТ-продуктів

затверджена наказом університету від 03 квітня 2023 р. № 319Ст

2. Термін подання студентом роботи до екзаменаційної комісії 18.05.2023

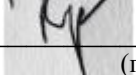
3. Вихідні дані до роботи сучасні дослідження в сфері пріоритезації вимог до ІТ-продуктів, сучасні дослідження методів пріоритезації MoSCoW, RICE, Buy-a-feature, моделі Кано, методу зваженої оцінки.


4. Перелік питань, що потрібно опрацювати в роботі дослідження існуючих методів пріоритезації вимог до ІТ-продуктів, їх переваг та недоліків; визначення характеристик, за якими можливо дослідити доречність використання метода у практичній ситуації; надання порівняльної характеристики досліджених методів; експериментальна перевірка ефективності досліджених методів.

## КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Вивчення досліджень на обрану тему	03.04.2023 – 05.04.2023	Виконано
2	Систематизація вивчених досліджень, вибір методів пріоритезації для дослідження	06.04.2023	Виконано
3	Аналіз переваг та недоліків обраних методів	07.04.2023 – 09.04.2023	Виконано
4	Визначення критеріїв, за якими можливо надати характеристику методу пріоритезації	10.04.2023 – 14.04.2023	Виконано
5	Порівняльний аналіз досліджених методів згідно визначених характеристик	14.04.2023 – 15.04.2023	Виконано
6	Визначення умов експерименту для перевірки ефективності досліджених методів	16.04.2023 – 19.04.2023	Виконано
7	Проведення експерименту	20.04.2023 – 28.04.2023	Виконано
8	Оформлення результатів експерименту	29.04.2023 – 30.04.2023	Виконано
9	Оформлення пояснювальної записки	01.05.2023 – 11.05.2023	Виконано
10	Оформлення презентаційних матеріалів захисту	12.05.2023 – 14.05.2023	Виконано
11	Представлення на рецензування	15.05.2023	Виконано
12	Представлення дипломного проекту в ЕК	19.05.2023	Виконано

Дата видачі завдання 03 квітня 2023 р.

Студент  \_\_\_\_\_  
(підпис)

Керівник роботи  \_\_\_\_\_ проф. каф. ІУС Максим ЄВЛАНОВ  
(підпис) (посада, власне ім'я, прізвище)

## РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи: 82 с., 1 рис., 10 таблиць, 12 джерел, 1 додаток.

ВИМОГИ ДО ІТ-ПРОДУКТІВ, МЕТОД ЗВАЖЕНОЇ ОЦІНКИ, ПРІОРИТЕЗАЦІЯ, МЕТОДИ ПРІОРИТЕЗАЦІЇ, МОДЕЛЬ КАНО, BUY-A-FEATURE, MOSCOW, RICE.

Об'єкт дослідження: процес пріоритезації вимог до ІТ-продуктів на всіх етапах життєвого циклу проєкта по розробці ІТ-продукту.

Предмет дослідження: найбільш розповсюджені методи пріоритезації вимог до ІТ-продуктів.

Мета роботи: порівняти методи пріоритезації вимог до ІТ-продуктів, надати характеристики, за якими можливо зробити висновок про доречність використання методу пріоритезації у практичній ситуації, дослідити методи пріоритезації на предмет ефективності.

Методи дослідження: вивчення, аналіз і систематизація існуючих дослідницьких матеріалів, абстрагування для надання характеристики кожному з методів, порівняння досліджених методів, проведення і вивчення результатів експерименту з використання методів у практичній ситуації.

В ході роботи проведено аналіз існуючих методів пріоритезації вимог до ІТ-продуктів. На підставі проведеного аналізу запропоновано перелік критеріїв для характеристики методів пріоритезації, за допомогою яких проведено порівняльний аналіз досліджених методів пріоритезації.

Проведено експеримент, за результатами якого доведено ефективність кожного з досліджених методів та доведені гіпотези, видвинуті під час проведення порівняльного аналізу досліджених методів.

Пропонується використовувати результати роботи як основу для визначення доцільного метода пріоритезації вимог для проєкта.

## ABSTRACT

Explanatory note: 82 p., 1 fig., 10 tabl., 12 sources, 1 ann.

BUY-A-FEATURE, KANO MODEL, MOSCOW, PRIORITIZATION, PRIORITIZATION METHODS, REQUIREMENTS FOR IT-PRODUCTS, RICE, WEIGHTED SCORING METHOD.

Object of research: the process of prioritizing requirements for IT products at all stages of the life cycle of an IT product development project.

Subject of research: the most common methods of prioritizing requirements for IT products.

Purpose: to compare the methods of prioritizing requirements for IT products, to provide characteristics that can be used to conclude that the prioritization method is appropriate in a practical situation, to investigate the prioritization methods for effectiveness.

Research methods: study, analysis and systematization of existing research materials, abstraction to characterize each of the methods, comparison of the studied methods, conducting and studying the results of an experiment on the use of methods in a practical situation.

In the course of the work, an analysis of existing methods for prioritizing requirements for IT products was carried out. Based on the analysis, a list of criteria for characterizing the prioritization methods is proposed, with the help of which a comparative analysis of the studied prioritization methods is carried out.

An experiment was carried out, which proved the effectiveness of each of the studied methods and proved the hypotheses put forward during the comparative analysis of the studied methods.

It is proposed to use the results of the work as a basis for determining the appropriate method of prioritizing requirements for a project.

## ЗМІСТ

Вступ.....	8
1 Процес пріоритезації вимог до ІТ-продуктів. Методи пріоритезації.....	11
1.1 Загальна характеристика процесу пріоритезації вимог до ІТ-продукту	11
1.2 Характеристика обраних для дослідження методів пріоритезації вимог до ІТ-продуктів.....	13
1.2.1 Метод MoSCoW .....	13
1.2.2 Метод RICE.....	16
1.2.3 Модель Кано .....	19
1.2.4 Метод Buy-a-function .....	24
1.2.5 Метод зваженої оцінки (Weighted Scoring) .....	26
2 Порівняльна характеристика методів пріоритезації вимог до ІТ-продуктів	30
2.1 Основні характеристики методів пріоритезації.....	30
2.2 Порівняння методів пріоритезації вимог до ІТ-продуктів за наданими характеристиками.....	33
3 Експериментальна перевірка ефективності методів пріоритезації вимог до ІТ-продуктів.....	40
3.1 Умови експерименту.....	40
3.2 Вхідні дані.....	41
3.3 Формування контрольних результатів.....	43
3.4 Застосування методів пріоритезації вимог.....	44
3.4.1 Метод MoSCoW .....	44
3.4.2 Метод Buy-a-function .....	46
3.4.4 Метод зваженої оцінки .....	50

3.4.5 Модель Кано .....	52
3.5 Висновок з проведення експерименту .....	53
4 Використання результатів дослідження .....	57
Висновки .....	58
Перелік джерел посилання .....	60
Додаток А. Графічний матеріал.....	<b>Ошибка! Закладка не определена.</b>

## ВСТУП

ІТ-продукти стають дедалі більш важливим елементом в бізнес- та технологічних процесах у сучасному світі. Швидкий темп розвитку технологій та зміна вимог користувачів диктують необхідність постійного удосконалення та підтримки ІТ-продуктів. Висока конкуренція на ринку підштовхує до все більшого ускладнення проєктів задля задоволення потреб користувачів.

Повний цикл розробки ІТ-продукту вимагає занадто великих вкладень коштів та часу. Ризик завеликої втрати часу, що призведе до неактуальності продукту і краху проєкту у найгіршому випадку, стає дедалі більшим. Для запобігання даному ризику були впроваджені гнучкі методології управління проєктами, такі як Scrum і Kanban.

Однією з найголовніших задач при застосуванні гнучких методологій управління проєктами є пріоритезація вимог до ІТ-продукту. Правильний розподіл часу та сил є запорукою успіху проєктів, тому що дозволяє сконцентруватись на найважливіших задачах в першу чергу та перевірити гіпотезу не вкладаючи завеликих коштів.

Стрімкий темп подій у сучасному світі вимагає постійної адаптації до змін, так як успіх ІТ-продукта сильно залежить від його актуальності. Швидка реакція на зміни ринку та поточні події дає велику перевагу над конкурентами. За таких умов, критичною необхідністю є постійний перегляд пріоритетів.

За рахунок такої актуальності проблеми пріоритезації вимог до ІТ-продуктів, було сформовано багато методів організації даного процесу. Кожен з методів має свої переваги та недоліки, які зумовлюють доречність використання того чи іншого методу на практиці.

Використання існуючих методів значно підвищує ефективність процесу пріоритезації вимог, однак недоречне їх використання може призвести до критичних наслідків, таких як надмірна витрата часу і коштів або неправильне

визначення пріоритетів – кожен з цих ризиків може призвести до краху усього проєкту.

Саме тому, для ефективного управління проєктом необхідно знати різні методи пріоритезації вимог до ІТ-продуктів і доречність їх використання у тій чи іншій практичній ситуації.

Отже, дослідження методів пріоритезації вимог до ІТ-продуктів є актуальною та важливою темою, яка стане корисною як для наукового світу, так і для використання на практиці.

Об'єкт дослідження: процес пріоритезації вимог до ІТ-продуктів на всіх етапах життєвого циклу проєкту по розробці ІТ-продукту.

Предмет дослідження: найбільш розповсюджені методи пріоритезації вимог до ІТ-продуктів.

Мета роботи: порівняти методи пріоритезації вимог до ІТ-продуктів, надати характеристики, за якими можливо зробити висновок про доречність використання методу пріоритезації у практичній ситуації, дослідити методи пріоритезації на предмет ефективності.

Завдання:

- дослідження існуючих методів пріоритезації вимог до ІТ-продуктів, їх переваг та недоліків;
- визначення характеристик, за якими можливо дослідити доречність використання метода у практичній ситуації;
- надання порівняльної характеристики досліджених методів;
- експериментальна перевірка ефективності досліджених методів.

Методи дослідження: для вирішення завдань МКР використано комплекс наукових методів дослідження.

Теоретичні методи дослідження: вивчення, аналіз і систематизація існуючих дослідницьких матеріалів на тему методів пріоритезації вимог до ІТ-продуктів. Абстрагування як метод для надання характеристики кожному з

методів, їх переваг та недоліків. Порівняння досліджених методів за визначеними характеристиками.

Емпіричні методи дослідження: проведення і вивчення результатів експерименту з використання методів у практичній ситуації для перевірки гіпотез, поставлених при використанні теоретичних методів.

# 1 ПРОЦЕС ПРІОРИТЕЗАЦІЇ ВИМОГ ДО ІТ-ПРОДУКТІВ.

## МЕТОДИ ПРІОРИТЕЗАЦІЇ

### 1.1 Загальна характеристика процесу пріоритезації вимог до ІТ-продукту

Процес пріоритезації вимог - це важливий етап розробки програмного забезпечення, який передбачає визначення та ранжування важливості вимог для конкретного проекту. Цей процес гарантує, що команда розробників зосередиться на найбільш важливих вимогах, аби забезпечити найбільшу цінність для зацікавлених сторін. Крім того, процес визначення пріоритетів допомагає управляти проектними ризиками, визначаючи потенційні проблеми на ранній стадії проекту, що дозволяє команді вирішити їх до того, як вони стануть значними.

Процес пріоритезації вимог починається зі збору всіх вимог до проекту. Вимоги можуть надходити з різних джерел, таких як інтерв'ю із зацікавленими сторонами, опитування або анкетування користувачів, спостереження за робочими процесами тощо. Після того, як всі вимоги будуть зібрані, команда проаналізує та оцінить кожен вимогу, щоб визначити її важливість та цінність для проекту.

Існує безліч методів визначення пріоритетності вимог, кожен з яких має свої переваги та недоліки. Метод MoSCoW є одним з найпопулярніших і найпростіших методів визначення пріоритетів. Він класифікує вимоги за чотирма рівнями пріоритетності: Must have, Should have, Could have і Won't have. Вимоги Must have - це ті, які є необхідними для успіху проекту, в той час як вимоги Should have є важливими, але не критично важливими. Вимоги Could have є бажаними, але не обов'язковими, а вимоги Won't have не включені в обсяг проекту.

Ще одним популярним методом визначення пріоритетності вимог є модель Кано. Вона класифікує вимоги за трьома категоріями: базові потреби, потреби продуктивності та потреби захоплення. Базові потреби - це вимоги,

які необхідні для належного функціонування програмного забезпечення, тоді як потреби продуктивності - це ті, що покращують продуктивність програмного забезпечення. Потреби захоплення - це функції, які захоплюють користувача і відрізняють програмне забезпечення від його конкурентів.

Процес аналітичної ієрархії (АНР) - це ще один метод визначення пріоритетів, який передбачає створення ієрархії критеріїв та оцінку кожної вимоги на основі її важливості для кожного критерію. Цей метод особливо корисний, коли є кілька зацікавлених сторін з різними пріоритетами.

Після визначення пріоритетності вимог команда розробників може почати працювати над найважливішими вимогами в першу чергу. Такий підхід гарантує, що проект принесе найбільшу користь зацікавленим сторонам на ранній стадії розробки.

Процес визначення пріоритетності вимог - це ітеративний процес, який слід переглядати протягом життєвого циклу розробки програмного забезпечення. По мірі просування проекту і надходження нової інформації, команді може знадобитися перегляд пріоритетності вимог, щоб гарантувати, що проект продовжує приносити найбільшу користь зацікавленим сторонам. Зацікавлені сторони також можуть змінювати свої вимоги в міру того, як вони дізнаються більше про проект, тому команда повинна бути готова до відповідного коригування пріоритетів.

Отже, процес визначення пріоритетності вимог є важливим етапом життєвого циклу розробки програмного забезпечення, який гарантує, що програмний проект принесе найбільшу цінність зацікавленим сторонам. Визначаючи та ранжуючи важливість вимог, команда розробників може зосередити свої зусилля на найбільш важливих вимогах та ефективно управляти проектними ризиками. Існує кілька методів визначення пріоритетності вимог, і вибір методу залежить від конкретних потреб проекту та залучених зацікавлених сторін. Процес визначення пріоритетності вимог - це ітеративний процес, який слід переглядати протягом життєвого циклу

розробки програмного забезпечення, щоб гарантувати, що проект принесе найбільшу цінність зацікавленим сторонам.

1.2 Характеристика обраних для дослідження методів пріоритезації вимог до ІТ-продуктів

### 1.2.1 Метод MoSCoW

MoSCoW - це метод визначення пріоритетів, який зазвичай використовується в управлінні проектами, щоб допомогти командам визначити та встановити пріоритетність завдань та вимог. У цьому розділі ми обговоримо метод визначення пріоритетів MoSCoW, як він працює та його переваги.

Метод MoSCoW - це простий, але ефективний спосіб визначити пріоритетність завдань або вимог, розділивши їх на чотири категорії: Must have, Should have, Could have і Won't have. Кожній з цих категорій присвоюється рейтинг пріоритетності на основі рівня важливості або терміновості для проекту або продукту, що розробляється:

- категорія Must have включає критичні завдання або вимоги, які необхідні для успіху проекту. Ці завдання або вимоги не можуть бути поставлені під загрозу, а їх невиконання призведе до провалу проекту. Їм надається найвищий пріоритет;

- до категорії Should have відносяться завдання або вимоги, які є важливими, але не критичними для успіху проекту. Їх можна відкласти, але необхідно виконати якнайшвидше. Їм присвоюється другий за пріоритетністю рейтинг;

- до категорії Could have відносяться завдання або вимоги, які є бажаними, але не обов'язковими для успіху проекту. Вони можуть бути відкладені або вилучені з проекту без суттєвого впливу на його результат. Їм надається третій за пріоритетністю рейтинг;

– до категорії Won't have відносяться завдання або вимоги, які не є необхідними для успіху проекту і можуть бути відкладені або повністю вилучені з проекту. Їм надається найнижчий пріоритет.

При використанні методу пріоритезації MoSCoW важливо залучити до процесу всі відповідні зацікавлені сторони. До них відносяться менеджери проекту, члени команди, клієнти та будь-які інші сторони, зацікавлені в результатах проекту.

Процес визначення пріоритетів зазвичай складається з кількох етапів. По-перше, команда проекту створює список усіх завдань або вимог, які необхідно виконати. Потім кожен пункт у списку оцінюється на основі його важливості та терміновості.

Команда проекту працює разом, щоб віднести кожне завдання або вимогу до однієї з чотирьох категорій: Must have, Should have, Could have і Won't have.

Після визначення пріоритетності завдань або вимог команда проекту може розпочати роботу над ними, починаючи з завдань або вимог, що мають бути виконані обов'язково. У міру просування проекту команда може змінювати пріоритети за необхідності, гарантуючи, що найбільш важливі завдання або вимоги будуть виконані в першу чергу.

Коли в процесі визначення пріоритетності категорій завдань за методом MoSCoW виникають суперечливі думки, важливо мати процедуру для їх вирішення. Ось кілька стратегій, які можна використовувати для управління конфліктами та досягнення консенсусу:

– сприяйте обговоренню: заохочуйте членів команди висловлювати свої погляди та активно вислуховувати думки один одного. Створіть середовище, яке сприяє відкритому спілкуванню та заохочує членів команди ставити запитання і звертатися за роз'ясненнями;

– зосередьтеся на цілях проекту: нагадайте членам команди про цілі та завдання проекту. Обговоріть, як кожне завдання або вимога сприяє

досягненню цих цілей, і подумайте, як кожне з них вписується в загальний графік проекту;

- обміркуйте вплив кожного завдання чи вимоги: оцініть потенційний вплив кожного завдання або вимоги на успіх проекту. Візьміть до уваги такі фактори, як необхідні ресурси, рівень необхідних зусиль і пов'язані з ними ризики;

- використовуйте об'єктивні критерії: Встановіть об'єктивні критерії для визначення пріоритетності завдань чи вимог. Це можуть бути такі фактори, як вартість, реалістичність або вплив кожного завдання чи вимоги;

- звертайтеся до зацікавлених сторін: проконсультуйтеся із зацікавленими сторонами проекту, такими як замовники або спонсори, щоб отримати їхню думку щодо визначення пріоритетності категорій завдань. Це допоможе гарантувати, що їхні перспективи та пріоритети будуть враховані в процесі прийняття рішень;

- ескалація до вищої інстанції: якщо конфлікт не вдається вирішити на рівні команди, винесіть питання на розгляд вищої інстанції, наприклад, спонсора або керівника проекту, для прийняття остаточного рішення.

Загалом, ключовим моментом у вирішенні конфліктів за методом MoSCoW є створення прозорого та спільного процесу, який заохочує відкриту комунікацію та об'єктивне прийняття рішень. Залучаючи до процесу всі зацікавлені сторони та розглядаючи різні точки зору, команди можуть гарантувати, що визначення пріоритетності завдань є справедливим, збалансованим та узгодженим з цілями проекту.

Переваги методу:

- чіткий і зрозумілий: метод MoSCoW є простим і зрозумілим, що робить його легким для розуміння і застосування всіма членами команди;

- визначення пріоритетності критично важливих завдань: визначаючи та встановлюючи пріоритетність критично важливих завдань або вимог, метод

MoSCoW гарантує, що найбільш важливі елементи проекту будуть завершені в першу чергу.

Недоліки методу:

- обмеженість категорій: метод MoSCoW передбачає лише чотири категорії для визначення пріоритетів, чого може бути недостатньо для складних проектів;

- суб'єктивність: визначення пріоритетності завдань або вимог базується на суб'єктивній оцінці проектної команди, що може призвести до упередженості та розбіжностей;

- брак деталізації: метод MoSCoW ніяк не враховує проектні ризики та складність виконуваних задач;

- надмірний акцент на критично важливих завданнях: зосередженість методу MoSCoW на критично важливих завданнях може призвести до того, що інші важливі завдання або вимоги будуть проігноровані або залишені поза увагою.

Таким чином, хоча метод визначення пріоритетів MoSCoW має свої переваги, він може не підходити для всіх проектів або команд. Керівники проектів повинні враховувати обмеження та потенційні недоліки методу, перш ніж впроваджувати його у свої процеси управління проектами.

### 1.2.2 Метод RICE

Метод RICE – це популярний фреймворк, який використовується організаціями для пріоритезації вимог. Це аббревіатура, що розшифровується як Reach, Impact, Confidence, Effort (охоплення, вплив, впевненість та зусилля). Метод RICE передбачає присвоєння балів кожній вимозі на основі цих чотирьох факторів, а потім обчислення загального балу для ранжування вимог.

Reach – це кількість користувачів або зацікавлених сторін, на яких вплине вимога. Воно враховує потенційну аудиторію, на яку вплине виконання певної вимоги. Чим вище охоплення, тим значніший вплив вимоги.

Impact – це рівень впливу вимоги на бізнес, якщо її буде впроваджено. Цей фактор враховує потенційні вигоди, які вимога принесе організації, наприклад, збільшення доходів або підвищення продуктивності. Чим вищий вплив, тим ціннішою є вимога для організації.

Confidence – це рівень впевненості в тому, що вимога принесе очікувані вигоди. Цей фактор враховує, наскільки організація впевнена в успіху вимоги, беручи до уваги такі фактори, як можливість реалізації та потенційні ризики чи перешкоди. Чим вища впевненість, тим більша ймовірність того, що вимога буде успішною.

Effort – це рівень зусиль, необхідних для виконання вимоги. Цей фактор враховує час, ресурси та складність, пов'язані з виконанням вимоги. Чим менші зусилля, тим легше і економічно ефективніше буде реалізація вимоги.

Для ефективного використання методу RICE важливо залучити ключові зацікавлені сторони з різних підрозділів організації. Це можуть бути менеджери проектів, менеджери продуктів, розробники, дизайнери, бізнес-аналітики та топ-менеджмент. Кожен член команди може внести свій унікальний досвід і бачення в процес пріоритезації, що допоможе забезпечити оцінку вимог з різних сторін.

Першим кроком у використанні методу RICE є визначення всіх вимог, які потребують пріоритезації. Це можна зробити, проаналізувавши цілі та завдання проекту, відгуки клієнтів та бізнес-потреби. Після того, як вимоги визначені, команда повинна присвоїти бали кожній вимозі на основі факторів RICE.

Фактор Reach (R) можна оцінити, підрахувавши кількість користувачів або зацікавлених сторін, на яких вплине кожна вимога. Це можна визначити, переглянувши дані про клієнтів, дослідження ринку або відгуки користувачів. Фактор Impact (I) можна оцінити, розглянувши потенційні вигоди для бізнесу

від кожної вимоги, такі як збільшення доходу або підвищення ефективності. Фактор Confidence (C) можна оцінити, проаналізувавши технічну реалістичність, потенційні ризики та інші фактори, які можуть вплинути на ймовірність успіху. Нарешті, фактор Effort (E) можна оцінити, визначивши ресурси, час і складність, необхідні для виконання кожної вимоги. Для цього може стати в нагоді попередньо проведена оцінка вимог.

Після того, як кожній вимозі присвоєно бали за всіма чотирма факторами, пріоритетність вимоги за методом RICE визначається за формулою:

$$\frac{R * I * C}{E} = RICE\ Score, \quad (1)$$

де R – оцінка параметру Reach (від 1 до 10);

I – оцінка параметру Impact (від 1 до 10);

C – оцінка параметру Confidence (від 1 до 10);

E – оцінка параметру Effort (від 1 до 10);

RICE Score – пріоритетність вимоги.

Переваги методу RICE:

- фактори, які впливають на результат пріоритезації, легко зрозуміти, тому цей метод є прозорим для команди та зацікавлених сторін;
- метод заохочує підхід до визначення пріоритетів на основі даних, коли бали присвоюються на основі об'єктивних критеріїв;
- метод дозволяє визначати пріоритетність вимог на основі комбінації факторів, а не на основі суб'єктивних думок;
- метод дозволяє команді зосередитись на тих вимогах, які мають найбільшу цінність для проєкту.

Недоліки методу RICE:

- метод може надмірно спрощувати складні вимоги і не враховувати їхній повний вплив або реалістичність;

- метод не враховує залежності між вимогами, а це важливий фактор, який може значно вплинути на успіх проєкту;
- система оцінювання може бути упередженою, оскільки окремі члени команди можуть виставляти бали по-різному, виходячи з власних поглядів та досвіду;
- метод може не підходити для всіх типів проєктів, особливо тих, які вимагають більш якісного та ретельного підходу до пріоритезації;
- згідно методу, RICE Score обернено пропорційно відноситься до оцінки зусиль (Effort) на виконання вимоги, що може знизити пріоритет важливих але складних задач на користь менш складних.

Загалом, метод RICE трохи складніший за метод MoSCoW, так як потребує об'єктивної оцінки декількох критеріїв, але при цьому залишається простим у використанні. Проте, простота у використанні виливається в недоліки, пов'язані з менш ретельним визначенням пріоритетів, аніж у інших методах.

### 1.2.3 Модель Кано

Модель Кано - це популярний інструмент, який використовується в розробці та управлінні продуктами для визначення пріоритетності вимог на основі потреб користувачів. Вона складається з п'яти категорій вимог, які класифікуються за ступенем їхнього впливу на задоволеність клієнтів: Must-be (обов'язкові), One-dimensional/Performance (вимоги продуктивності), Attractive (привабливі), Indifferent (байдужі) та Reverse (зворотні).

Згідно моделі Кано, реалізація різних функцій по різному впливає на задоволення користувачів. Графік залежності задоволення користувачів від ступеня реалізації функцій згідно моделі Кано представлений нижче (рис. 1).

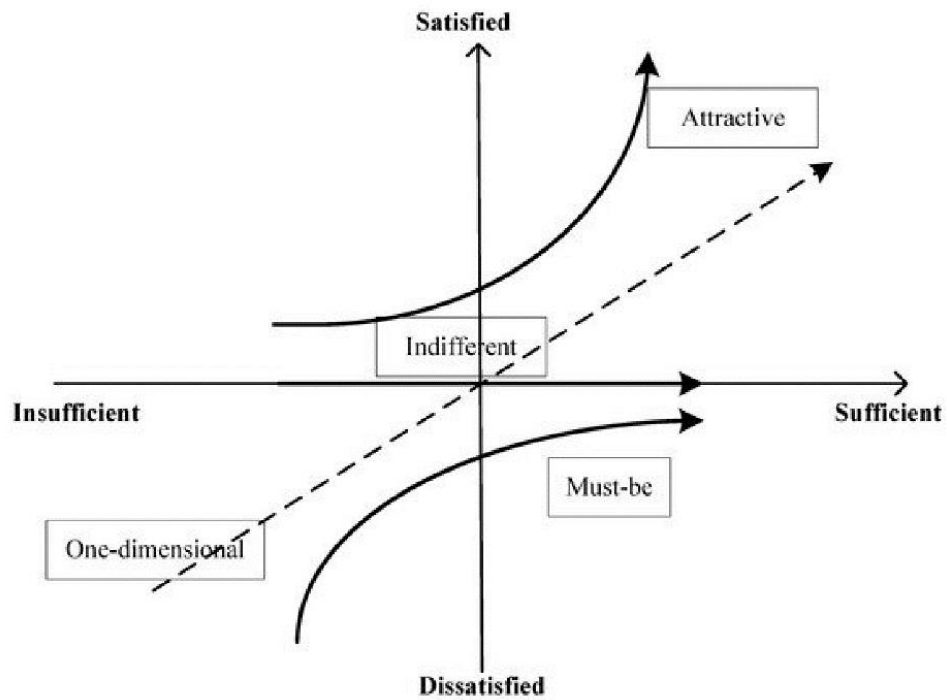


Рисунок 1 – Графік моделі Канона

Для використання моделі Канона проводиться опитування для вимірювання важливості та рівня задоволеності кожної вимоги. Питання опитування розроблені таким чином, щоб отримати відповіді, які вказують на те, наскільки користувачі очікують певну функцію і наскільки вони були б задоволені, якби ця функція була присутня в продукті.

Опитування є критично важливим компонентом моделі Канона, оскільки воно надає дані, необхідні для розподілу вимог користувачів за п'ятьма категоріями. Питання опитування розроблені таким чином, щоб отримати відповіді, які вказують на те, наскільки користувачі очікують певну функцію і наскільки вони були б задоволені, якби ця функція була присутня в продукті.

Опитування зазвичай проводиться серед репрезентативної вибірки користувачів або потенційних користувачів. Розмір вибірки та демографічна складова визначаються залежно від продукту чи послуги, що розробляється або вдосконалюється. Добре сплановане опитування гарантує, що зібрані дані будуть точними і репрезентативними для цільової аудиторії.

Питання, включені до опитування згідно моделі Кано, є парними. По кожній з функцій, яка планується до реалізації у кінцевому продукті, задаються питання:

- «яка у вас буде реакція, якщо [функція] буде реалізована у кінцевому продукті?»;
- «яка у вас буде реакція, якщо [функція] НЕ буде реалізована у кінцевому продукті?».

На кожне з запитань надаються варіанти відповідей:

- мені це сподобається;
- я очікую, що це буде реалізовано;
- мені все одно;
- я потерплю;
- мені це не сподобається.

Відповіді на пари запитань по кожній з функцій дають нам відповідь, до якої з категорій відноситься дана функція для цього користувача, згідно таблиці нижче (таблиця 1).

Таблиця 1 – Обробка відповідей на опитування згідно моделі Кано

		Функція не реалізована				
		Подобається	Очікую	Все одно	Потерплю	Не подобається
Функція реалізована	Подобається	Q	A	A	A	P
	Очікую	R	Q	I	I	M
	Все одно	R	I	I	I	M
	Потерплю	R	I	I	Q	M
	Не подобається	R	R	R	R	Q

Категорії, до яких потрапляють функції згідно опитування:

- Must-be (M) – це функції, які клієнти очікують бачити в продукті або послугі, і їх відсутність призведе до незадоволення;
- Performance (P) - це функції, які підвищують задоволеність клієнтів, оскільки покращують функціональність продукту;
- Attractive (A) - це функції, які клієнти не очікують, але дуже хочуть мати;
- Indifferent (I) - це ті функції, що не мають значного впливу на задоволеність клієнтів;
- Reverse (R) - це ті функції, що знижують задоволеність клієнтів, якщо їх реалізувати;
- Questionable (Q) - відповіді користувача на парні запитання суперечать одна одній і не враховуються.

Після того, як зібрані результати опитування, вимогам надається пріоритет на основі їх впливу на задоволеність користувачів та ресурсів, необхідних для їх виконання. Найвищий пріоритет надається вимогам по реалізації функцій категорії must-be, за ними йдуть performance, а потім - attractive. Вимоги по реалізації функцій в категорії indifferent мають низький пріоритет, а функції категорії reverse взагалі не розглядаються.

Модель Кано забезпечує системний підхід до розробки та управління продуктом, гарантуючи, що ресурси розподіляються на критичні вимоги, які мають найбільш значний вплив на задоволеність клієнтів. Вона допомагає організаціям визначити потенційні конкурентні переваги шляхом виявлення особливостей, які клієнти вважають приємними. Крім того, її можна використовувати для сегментування клієнтів на основі їхніх потреб і визначення пріоритетності вимог на основі важливості кожного сегмента.

Переваги моделі Кано:

- модель Кано допомагає організаціям зрозуміти потреби клієнтів і визначити пріоритетність вимог на основі їх важливості для клієнтів;

- вона забезпечує системний підхід до розробки та управління продуктом, гарантуючи, що ресурси розподіляються на критичні вимоги;
- модель може бути використана для визначення потенційних конкурентних переваг шляхом виявлення особливостей, які клієнти вважають приємними;
- модель легко зрозуміти і донести до зацікавлених осіб, що поліпшує комунікацію команди зі стейкхолдерами.

Недоліки моделі Кано:

- модель залежить від точності та повноти результатів опитування. Погано розроблені опитування або упереджені вибірки можуть призвести до неточних результатів;
- модель не враховує вартість реалізації кожної вимоги, що може призвести до нереалістичної розстановки пріоритетів;
- модель припускає, що потреби клієнтів є статичними, і не враховує мінливі потреби або тенденції;
- модель не враховує компроміси між вимогами. Пріоритет однієї вимоги над іншою може мати непередбачувані наслідки.

Модель Кано особливо корисна в наступних сценаріях:

- розробка нового продукту: модель Кано може допомогти визначити пріоритетність вимог і функцій для нового продукту, гарантуючи, що ресурси будуть спрямовані на виконання критично важливих вимог;
- покращення продукту: модель Кано може бути використана для визначення сфер вдосконалення існуючого продукту або послуги, гарантуючи, що найбільш критичні вимоги будуть виконані;
- аналіз конкурентів: модель Кано може бути використана для визначення потенційних конкурентних переваг шляхом виявлення особливостей, які клієнти вважають приємними;

– сегментація клієнтів: модель Кано можна використовувати для сегментації клієнтів на основі їхніх потреб і визначення пріоритетності вимог на основі важливості кожного з сегментів.

Модель Кано є цінним інструментом для пріоритезації вимог при розробці та управлінні продуктом. Вона забезпечує системний підхід до розуміння потреб користувача та розподілу ресурсів для виконання найважливіших вимог. Однак вона підходить лише для тих проектів, де є прямий контакт з користувачами, тому що оцінка критеріїв від зацікавлених сторін не буде позбавлена суб'єктивності і буде спотворювати результат пріоритезації.

#### 1.2.4 Метод Buy-a-function

Метод Buy-a-function - це гра, за результатами якої можна визначити пріоритет вимог до ІТ-продукту на основі їхньої відносної важливості для зацікавлених сторін. Метод передбачає виділення гіпотетичного бюджету на кожну вимогу, після чого, зацікавленим сторонам пропонується "купити" ті вимоги, які вони вважають найбільш важливими.

Нижче наведено кроки по проведенню гри Buy-a-Function.

Крок 1 – визначення вимог. Визначте усі ІТ-вимоги, пріоритет яких необхідно визначити.

Крок 2 – визначення ціни і бюджету. Для кожної з вимог необхідно визначити її ціну. Задля цього можна використати оцінку реалізації даної функції, яка має включати в себе повний цикл робіт – від аналізу вимоги до її релізу. Також необхідно визначити, який бюджет буде в кожного зі стейкхолдерів. Бюджет має бути складений відносно проміжку часу, на який буде зроблена пріоритизація (це може бути період від декількох спринтів до декількох місяців).

Крок 3 – вибір зацікавлених сторін, які прийматимуть участь у грі. Запросіть зацікавлені сторони до участі в процесі визначення пріоритетів. Зацікавлені сторони можуть бути як внутрішніми, так і зовнішніми по відношенню до організації і можуть включати працівників, клієнтів, постачальників або партнерів.

Крок 4 – поясніть метод. Поясніть стейкхолдерам правила гри. Опишіть, як вони можуть використати виділений їм бюджет для "купівлі" вимог, які вони вважають найбільш важливими. При цьому, важливо наголосити, що необов'язково витратити увесь бюджет.

Крок 5 – дозвольте зацікавленим сторонам "купити". Дайте стейкхолдерам певний час, щоб "купити" вимоги, які вони вважають найбільш важливими. Вони можуть витратити свій бюджет на будь-яку кількість вимог поки вистачає бюджету. Після "купівлі" попросіть пояснити, чому купили саме ці вимоги.

Крок 6 – зведіть результати в таблицю. Після того, як зацікавлені сторони завершили "купівлю", зведіть результати в таблицю, щоб визначити найважливіші вимоги.

Переваги методу:

- метод Buy-a-function простий для розуміння і може бути реалізований без спеціальної підготовки або знань;
- метод залучає зацікавлені сторони до процесу пріоритезації, надаючи їм відчуття причетності та залученості;
- метод гарантує, що всі зацікавлені сторони мають рівні можливості для висловлення своїх думок та уподобань;
- результати методу "Купи функцію" дають чітке уявлення про найважливіші вимоги для зацікавлених сторін, тому метод є ефективним.

Недоліки методу:

- метод спирається на суб'єктивні думки та вподобання зацікавлених сторін, на які можуть впливати різні фактори;

- обмежений бюджет може призвести до того, що деякі з важливих вимог можуть оминати увагою;

- метод може зайняти багато часу, особливо якщо приймає участь багато зацікавлених сторін або пріоритезації потребує завеликий перелік вимог.

Отже, метод Buy-a-function – це простий і ефективний метод пріоритезації вимог до ІТ-продукту. Він залучає зацікавлені сторони до процесу визначення пріоритетів і дає чітке уявлення про найважливіші вимоги. Однак, важливо відзначити суб'єктивний характер методу та затрати часу на підготовку та проведення пріоритезації за ним. Загалом, метод "Купи функцію" може бути корисним інструментом у процесі визначення пріоритетності ІТ-вимог, особливо коли залучення зацікавлених сторін є важливим фактором.

#### 1.2.5 Метод зваженої оцінки (Weighted Scoring)

Метод зваженої оцінки передбачає присвоєння ваг різним критеріям, а потім оцінку кожної альтернативи залежно від того, наскільки вона відповідає цим критеріям. Потім бали множаться на вагу і підсумовуються для отримання зваженого балу для кожної альтернативи. Відповідно, пріоритетність задачі за цим методом визначається згідно зваженої оцінки – від найбільшої оцінки до найменшої.

Першим кроком у використанні методу зваженої оцінки для визначення пріоритетності вимог ІТ-проекту є визначення критеріїв, які будуть використовуватися для оцінки вимог. Ці критерії повинні бути специфічними для проекту і відповідати його цілям, завданням і потребам зацікавлених сторін. Наприклад, критерії оцінки вимог до проекту з розробки ІТ-продукту можуть включати такі фактори, як функціональність, зручність використання, масштабованість, безпека і вартість розробки.

Наступним кроком є розробка системи оцінювання, яка буде використовуватися для оцінки кожної вимоги на основі того, наскільки вона відповідає визначеним критеріям. При цьому, для кожного з визначених критеріїв може використовуватись різна система оцінювання.

Щоб забезпечити узгодженість оцінювання критеріїв за всіма вимогами, важливо надати експертам чіткі вказівки та провести навчання щодо системи оцінювання. Це допоможе гарантувати, що критерії оцінюються послідовно за всіма вимогами і що оцінка буде об'єктивною.

Останнім кроком перед безпосередньою оцінкою є вибір ваги критеріїв. Вибір відповідних ваг для критеріїв є вирішальним аспектом методу зваженого оцінювання. Вага, присвоєна кожному критерію, визначає його відносну важливість у процесі оцінювання. Таким чином, вага, присвоєна критерію, повинна відображати рівень його важливості для досягнення цілей проекту.

Вибір вагових коефіцієнтів для критеріїв вимагає участі зацікавлених сторін та експертів, які повинні добре розуміти цілі, обмеження та фактори успіху проекту. Вклад зацікавлених сторін повинен бути використаний для присвоєння відповідної ваги кожному критерію на основі їх відносної важливості для проекту.

Існує кілька методів визначення ваги критеріїв. Один з них полягає у використанні методу прямого зважування, коли зацікавлені сторони просять присвоїти кожному критерію певну вагу у відсотках, виходячи з їхньої відносної важливості. Інший підхід - метод попарного порівняння, коли стейкхолдерів просять порівняти пари критеріїв і вибрати, який з них є більш важливим. Результати попарних порівнянь потім використовуються для визначення ваг.

Незалежно від методу, який використовується для присвоєння ваг, важливо переконатися, що ваги, присвоєні кожному критерію, складають 100%. Це гарантує, що критерії оцінюються збалансовано і всебічно, і жоден критерій не буде переоцінений або недооцінений.

Важливо також періодично переглядати і коригувати вагу, присвоєну критеріям, особливо в міру зміни вимог і цілей проекту. Вагу, присвоєну кожному критерію, слід коригувати, щоб відобразити його поточну актуальність і важливість для проекту.

Після того, як кожна вимога оцінена, бали множаться на відповідну вагу та підсумовуються для отримання зваженої оцінки за кожен вимогу. Вимоги з найвищою зваженою оцінкою визначаються як пріоритетні і відбираються як найбільш важливі для успіху проекту.

Метод зваженої оцінки має кілька переваг, серед яких:

- прозорість: метод зваженої оцінки є прозорим, і процес прийняття рішень можна легко пояснити зацікавленим сторонам;
- налаштовуваність: метод можна налаштувати відповідно до конкретного рішення, що приймається, забезпечуючи гнучкість процесу прийняття рішень;
- об'єктивність: використання об'єктивних критеріїв і ваг гарантує, що рішення ґрунтується на фактичній інформації, а не на особистих упередженнях.

Однак, метод зваженої оцінки також має певні обмеження:

- можливість суб'єктивності: процес присвоєння ваг і балів критеріям може бути суб'єктивним, що може призвести до варіацій в остаточному рішенні;
- обмеженість: метод обмежений обраними критеріями та вагами, і важливі фактори, які не були включені в аналіз, можуть бути проігноровані.

Метод зваженої оцінки є корисним інструментом для визначення пріоритетності вимог на основі їхньої відносної важливості. Метод передбачає присвоєння ваг критеріям і оцінку кожної вимоги залежно від того, наскільки добре вона задовольняє ці критерії. Вимоги з найвищими зваженими оцінками є найбільш пріоритетними. Метод має низку переваг, серед яких прозорість,

можливість налаштування та об'єктивність. Однак він також має і недоліки, такі як суб'єктивність та обмеженість.

## 2 ПОРІВНЯЛЬНА ХАРАКТЕРИСТИКА МЕТОДІВ ПРІОРИТЕЗАЦІЇ ВИМОГ ДО ІТ-ПРОДУКТІВ

### 2.1 Основні характеристики методів пріоритезації

Як бачимо з результатів дослідження в розділі 1, кожен з методів пріоритезації має свої переваги та недоліки. Виходячи з результатів дослідження методів пріоритезації, можемо виділити характеристики, за якими можна характеризувати кожен з методів. Ці характеристики є вирішальними при виборі методу пріоритезації для використання у практичній ситуації:

а) об'єктивність. За об'єктивністю методи пріоритезації можна розділити на дві категорії – об'єктивні та суб'єктивні:

- оцінка пріоритетів за об'єктивними методами пріоритезації спирається на дослідження, які можливо обчислити; або на декілька факторів, в результаті чого поступово втрачається суб'єктивність;
- суб'єктивні методи пріоритезації повністю спираються на думку зацікавленої сторони або групи зацікавлених сторін. При цьому, у випадку залучення достатньо великої групи зацікавлених сторін, метод може стати об'єктивним в наслідок меншого впливу суб'єктивності зацікавлених сторін;

б) прозорість. За прозорістю методи пріоритезації можна розділити на дві категорії – прозорі та непрозорі:

- результати пріоритезації за прозорими методами пріоритезації можна дослідити і зробити висновки з причин пріоритету тієї чи іншої вимоги. Прозорість процесу пріоритезації є великою перевагою при роботі з зацікавленими сторонами, які необхідно інформувати про результати роботи та які не приймають

безпосередньої участі у внутрішніх процесах (наприклад, донори), а також при необхідності проведення ретроспективного аналізу;

- результати пріоритезації за непрозорими методами не дають можливості провести ретроспективний аналіз з причин прийняття рішень по пріоритетності вимог;

в) складність. За складністю методи пріоритезації можна поділити на дві категорії – прості та складні:

- прості методи пріоритезації не вимагають проведення спеціальних підготовчих заходів. Такі методи легко впровадити в організації навіть в тому випадку, коли раніше процес пріоритезації не був формалізований;

- складні методи пріоритезації вимагають проведення спеціальних підготовчих заходів – таких як додаткові дослідження ринку, опитування користувачів тощо;

г) витрати часу. Витрати часу на застосування методу пріоритезації можна оцінити таким чином – малі витрати часу, середні витрати часу та великі витрати часу:

- витрати часу можна вважати малими у тому випадку, коли результат процесу пріоритезації можна отримати за одну сесію з зацікавленими сторонами, а для проведення сесії пріоритезації не потрібна додаткова підготовка;

- витрати часу можна вважати середніми у тому випадку, коли для отримання результату пріоритезації необхідно декілька сесій з зацікавленими сторонами, або ж у тому випадку, коли для проведення сесії з зацікавленими сторонами необхідна додаткова підготовка;

- витрати часу можна вважати великими у тому випадку, коли результат пріоритезації неможливо отримати без проведення додаткових заходів, які займають великий проміжок часу або

залежать не тільки від команди розробки – такі як зовнішні дослідження або опитування;

д) гнучкість. За гнучкістю методи пріоритезації можна поділити на дві категорії – гнучкі та негнучкі:

- гнучкі методи пріоритезації не вимагають специфічних умов для їх застосування. Такі методи пріоритезації легко налаштувати, виходячи зі специфіки проекту;

- негнучкі методи пріоритезації вимагають специфічних умов на проєкті, при відсутності яких метод є недієвим;

е) чіткість. За чіткістю методи пріоритезації можна поділити на дві категорії – чіткі та нечіткі:

- результат чітких методів пріоритезації виражений числами, що дозволяє більш влучно оцінити пріоритет вимог. Чим більший список вимог необхідно пріоритезувати, тим більшу перевагу мають чіткі методи, так як вони дають можливість оцінити, які з вимог є найпершим пріоритетом;

- результат нечітких методів пріоритезації виражений категоріями, які допомагають зрозуміти загальну картину, але вимоги в одній і тій же категорії можуть потребувати додаткового розгляду пріоритетів;

ж) орієнтованість. Методи пріоритезації вимог можуть бути внутрішньо- та зовнішньо-орієнтовані:

- внутрішньо-орієнтовані методи пріоритезації відображають пріоритетність вимог згідно інтересів зацікавлених сторін та команди розробки;

- зовнішньо-орієнтовані методи пріоритезації відображають пріоритетність вимог згідно інтересів користувачів.

Категоризація методів пріоритезації вимог до ІТ-продуктів за даним переліком характеристик дає можливість оцінити найважливіші переваги та

недоліки методів. Завдяки цьому, можна зробити висновки щодо доречності того чи іншого методу у практичній ситуації. Однак, варто зазначити, що кожен з методів може мати специфічні ознаки, які не піддаються категоризації. Через це, даний перелік дає змогу швидко скоротити список методів пріоритезації, які варті розгляду, але не може дати остаточну відповідь щодо того, який саме метод пріоритезації необхідно використати.

## 2.2 Порівняння методів пріоритезації вимог до ІТ-продуктів за наданими характеристиками

Виходячи з досліджень методів пріоритезації вимог до ІТ-продуктів, проведених в рамках розділу 1, а також з характеристик, які були відокремлені в розділі 2.1, можемо характеризувати кожен з розглянутих методів пріоритезації:

### а) метод MoSCoW:

- за об'єктивністю: суб'єктивний. У методі немає ніяких механізмів по запобіганню суб'єктивних суджень зацікавлених сторін, які можуть напряду впливати на результат пріоритезації. Вплив суб'єктивних поглядів на результат найбільший серед розглянутих методів;
- за прозорістю: непрозорий. Результати методу ніяк не відповідають на питання «чому?» і без додаткової фіксації пояснень пріоритету вимог контекст втрачається;
- за складністю: простий. Найбільш простий метод з досліджених, не вимагає нічого окрім розподілу вимог між категоріями пріоритетності;
- витрати часу: малі. Завдяки простоті у використанні, метод не вимагає великих витрат часу;

- за гнучкістю: гнучкий. Не вимагає специфічних умов для використання, може бути використаний у будь-яких ситуаціях;
- за чіткістю: нечіткий. Згідно методу MoSCoW, є всього три категорії вимог, які варті уваги (Must have, Should have, Could have). За умови великої кількості вимог, які потребують пріоритетизації, метод може надати лише вектор руху, але не виділить одну або декілька найпріоритетніших вимог, проте за умови маленької кількості вимог, метод буде ефективним;
- за орієнтованістю – внутрішньо-орієнтований;

б) метод RICE:

- за об'єктивністю: об'єктивний. Завдяки використанню декількох параметрів для визначення пріоритету, ступінь впливу суб'єктивної думки на результат пріоритетизації знижується, завдяки чому метод можна назвати об'єктивним;
- за прозорістю: прозорий. Параметри, які використовуються для пріоритетизації згідно методу RICE, дають чітке розуміння, чому у вимоги саме такий пріоритет і дозволяють провести ретроспективний аналіз;
- за складністю: простий. Метод RICE є дещо складнішим за MoSCoW за рахунок необхідності оцінки вимог з боку декількох параметрів, однак ця оцінка не потребує додаткових зусиль і спирається на інформацію, яка має бути відома учасникам проекту до початку процесу пріоритетизації вимог;
- витрати часу: середні. За рахунок того, що кожна вимога потребує обговорень зі сторони чотирьох факторів, однієї сесії з пріоритетизації вимог за методом RICE може не вистачити для розгляду повного скоупу;
- за гнучкістю: гнучкий. Параметри оцінки пріоритетності вимог за методом RICE актуальні для будь-якого проекту з розробки ІТ-

продукту, тому цей метод не потребує специфічних умов для його застосування;

- за чіткістю: чіткий. Завдяки обчисленню RICE Score, за яким визначається пріоритет вимоги, метод дає чітке розуміння з приводу відносної пріоритетності вимог;

- за орієнтованістю: внутрішньо-орієнтований;

в) модель Кано:

- за об'єктивністю: об'єктивний. Завдяки тому, що модель Кано орієнтована на потреби користувачів, а також завдяки великій виборці даних опитування, оцінка пріоритетів за моделлю Кано є об'єктивною;

- за прозорістю: прозорий. Процес пріоритезації вимог за моделлю Кано спирається на опитування, завдяки чому досягається прозорість процесу для усіх зацікавлених сторін;

- за складністю: складний. Проведення опитування користувачів, яке є невід'ємною частиною процесу пріоритезації за цим методом, є складним процесом, що вимагає ретельної підготовки;

- витрати часу: великі. Використання цього метода передбачає підготовку і проведення опитування користувачів, що є зовнішнім фактором, а це може значно збільшити час, необхідний для завершення процесу;

- за гнучкістю: негнучкий. Модель Кано може бути застосована лише на тих проектах, де відомі кінцеві користувачі ІТ-продукту і наявна база користувачів для зворотного зв'язку. Через необхідність наявності цих обставин метод не є гнучким;

- за чіткістю: чіткий. Хоча за цим методом вимоги поділяються на п'ять категорій, за допомогою кількісного аналізу відповідей респондентів отримуємо чіткі пріоритети вимог. Зі збільшенням

кількості респондентів, що приймали участь в опитуванні, результат пріоритезації стає дедалі чіткішим;

- за орієнтованістю: зовнішньо-орієнтований, що значно виділяє цей метод на фоні інших;

г) метод Buy-a-function:

- за об'єктивністю: суб'єктивний. Метод спирається лише на суб'єктивну думку зацікавлених сторін з приводу пріоритезованих вимог. Суб'єктивність методу зменшується зі збільшенням залучених стейкхолдерів, але при цьому збільшуються витрати часу, тому таке вирішення проблеми не є ефективним;

- за прозорістю: прозорий. Завдяки тому, що дані з приводу "купування" функцій різними стейкхолдерами фіксуються (тому що ці дані є вирішальними для визначення пріоритетів) під час процесу пріоритезації, процес є прозорим і може бути ретроспективно проаналізованим;

- за складністю: простий. Для пріоритезації вимог за методом Buy-a-function, необхідно тільки провести оцінку витрат на розробку кожної з вимог. Це окремий процес, який впроваджується на більшості проєктів з розробки ІТ-продуктів. За умови, що процес оцінки витрат вже впроваджений на проєкті, витрати на впровадження методу Buy-a-function будуть мінімальними;

- витрати часу: малі. Процес пріоритезації за методом Buy-a-function достатньо простий і зрозумілий для учасників процесу, а також не потребує додаткових обговорень між учасниками, тому що кожен зі стейкхолдерів особисто приймає рішення з "купування" функцій. Витрати часу можуть зростати в тому випадку, коли задля підвищення рівню об'єктивності збільшується кількість залучених стейкхолдерів;

- за гнучкістю: гнучкий. Метод може бути застосований на будь-якому проєкті, так як не вимагає специфічних умов;
- за чіткістю: чіткий. Завдяки використанню вартості вимог, яка відображає складність їх реалізації, після завершення етапу "купування" отримуємо чітку картину, які вимоги є більш пріоритетними;
- за орієнтованістю: внутрішньо-орієнтований;

д) метод зваженої оцінки:

- за об'єктивністю: об'єктивний. Метод зваженої оцінки дає можливість налаштовувати параметри, які впливають на пріоритетність вимоги. Завдяки використанню оцінки багатьох факторів, вплив суб'єктивних думок на результат пріоритезації зменшується. Однак, як наслідок необережного вибору факторів пріоритезації, об'єктивність методу може знижуватись;
- за прозорістю: прозорий. Метод явно зображує, які саме фактори і як вплинули на пріоритет тієї чи іншої вимоги, тому він є повністю прозорим і дає можливість ретроспективного аналізу;
- за складністю: простий/складний. Даний метод цілком налаштовується при його використанні, тому процес пріоритезації може бути як простим, так і складним, залежно від критеріїв оцінювання пріоритетності та методів оцінки;
- витрати часу: середні/великі. Метод вимагає однієї або більше підготовчих сесій для підготовки до використання – для визначення критеріїв оцінки пріоритетності та вагів кожного з критеріїв, після чого вже можна приступати до безпосередньої пріоритезації вимог. Окрім того, враховуючи те, що метод підлягає налаштуванню, залежно від критеріїв оцінювання пріоритетності та методів оцінки, час на пріоритезацію може збільшуватись;

- за гнучкістю: гнучкий. Завдяки тому, що метод налаштовується командою, яка його використовує, він є найбільш гнучким серед усіх розглянутих методів, тому що під кожний проєкт можливо визначити різні критерії, які є більш або менш важливими для визначення пріоритетів;
- за чіткістю: чіткий. Результатом пріоритезації за методом зваженої оцінки є число, що відображає пріоритетність вимоги, що дає можливість чітко зрозуміти, які вимоги є більш або менш важливими за інші;
- за орієнтованістю: внутрішньо-орієнтований.

Відобразимо результати аналізу методів пріоритезації вимог до ІТ-продуктів за визначеними у цьому розділі характеристиками у вигляді таблиці для порівняння (табл. 2).

Таблиця 2 – Результат порівняння методів пріоритезації

Метод	Характеристика		
	За об'єктивністю	За прозорістю	За складністю
MoSCoW	суб'єктивний	непрозорий	простий
RICE	об'єктивний	прозорий	простий
Модель Кано	об'єктивний	прозорий	складний
Buy-a-function	суб'єктивний	прозорий	простий
Зважена оцінка	об'єктивний	прозорий	простий/складний

Продовження таблиці 2

Метод	Характеристика			
	Витрати часу	За гнучкістю	За чіткістю	Орієнтованість
MoSCoW	малі	гнучкий	нечіткий	внутрішня
RICE	середні	гнучкий	чіткий	внутрішня
Модель Кано	великі	негнучкий	чіткий	зовнішня

Buy-a-function	малі	гнучкий	чіткий	внутрішня
Зважена оцінка	середні/ великі	гнучкий	чіткий	внутрішня

Таким чином, можемо побачити, що кожен з розглянутих методів унікальний за своїми характеристиками, що дозволяє порівнювати методи пріоритезації вимог до ІТ-продуктів використовуючи дані характеристики. Отже, робимо висновок, що визначені у даному розділі характеристики, хоча і не дають повного розуміння методу, можуть бути використані для визначення найбільш доречного методу у практичній ситуації.

## **3 ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА ЕФЕКТИВНОСТІ МЕТОДІВ ПРІОРИТЕЗАЦІЇ ВИМОГ ДО ІТ-ПРОДУКТІВ**

### **3.1 Умови експерименту**

При застосуванні методів пріоритизації вимог до ІТ-продуктів важливо розуміти, що вони є ефективними. Ефективністю методів пріоритизації вимог вважаємо можливість отримати рівнозначні результати в наслідок застосування методу при однакових вхідних даних. В рамках експерименту, спробуємо дослідити ефективність розглянутих у роботі методів пріоритизації вимог.

Робоча гіпотеза: кожен з розглянутих методів є ефективним, тому вибір конкретного методу залежить вже від умов проєкту, наявних ресурсів, інформації та інших факторів.

Порядок проведення експерименту:

- а) формування контрольних результатів. Для контрольних результатів використаємо метод Вігера для пріоритизації вимог, як метод, який не підлягає дослідженню в рамках даного експерименту;
- б) застосування методів пріоритизації вимог:
  - 1) метод MoSCoW;
  - 2) метод Buy-a-function;
  - 3) метод RICE;
  - 4) метод зваженої оцінки;
  - 5) модель Кано;
- в) порівняння отриманих результатів з контрольними даними, висновок з ходу експерименту.

В якості вхідних даних для експерименту використаємо гіпотетичний проєкт з розробки веб-застосунку для проведення онлайн сесій з інтелектуальної гри «Мафія». Враховуючи гіпотетичність проєкту, для експерименту сформуємо вимоги високого рівня та визначимо фактори, які

будуть важливі в рамках процесу пріоритезації. Важливо зазначити, що в рамках даної роботи не розглядаються проєктні процеси, які не мають прямого впливу на процес пріоритезації.

Так як вхідні дані мають бути однаковими для перевірки ефективності методів пріоритезації, кожен з методів буде використаний однією і тією ж самою групою осіб, які виступатимуть в якості стейкхолдерів проєкту. Так як проєкт є гіпотетичним, в якості стейкхолдерів були обрані особи, які мають експертні знання в обраному домені, а саме – судді і ведучі інтелектуальної гри «Мафія».

Так як один з методів, а саме модель Кано, вимагає проведення користувацького опитування, також була обрана група осіб, які можуть виступати в якості користувачів подібного застосунку, а саме – гравці в інтелектуальну гру «Мафія».

Для чистоти експерименту, першими будуть застосовані суб'єктивні методи пріоритезації вимог, щоб запобігти впливу результатів застосування об'єктивних методів на результати застосування суб'єктивних методів. Через це, хоча за логічним порядком проведення експерименту контрольні результати мають бути сформовані першими, фактичний порядок застосування методів буде відрізнятися.

### 3.2 Вхідні дані

Сформували ключові характеристики проєкту, які матимуть вплив на прийняття рішень з приводу пріоритетів:

- розробкою займається компанія-стартап;
- на даний момент у компанії немає зовнішнього фінансування, тому бюджет проєкту суворо обмежений;
- в рамках проєкту – розробка рішення з нуля, з можливістю використання готових фреймворків для розробки;

- бюджету проєкту вистачає на три місяці, за результатом яких необхідно представити MVP (Minimal Valued Product);
- команда розробки складається з front-end розробника, back-end розробника, дизайнера та бізнес-аналітика/тестувальника;
- кожен з членів команди є висококваліфікованим спеціалістом у своїй галузі і має усі необхідні знання для розробки;
- процес розробки включатиме в себе аналіз вимог, формування документації, розробку дизайну, розробку веб-застосунку (і серверної, і клієнтської частин) і, за необхідності, розробку бази даних.

Для пріоритезації сформували наступний перелік вимог високого рівня до веб-застосунку для онлайн гри в інтелектуальну гру «Мафія»:

- аутентифікація користувачів. Веб-застосунок повинен дозволяти користувачам створювати обліковий запис, входити і виходити з нього. Користувачі повинні мати доступ до своїх профілів та оновлювати інформацію в акаунті;
- ігрове лобі. На сайті має бути ігрове лобі, де гравці можуть приєднуватися до ігор або створювати нові ігри. В ігровому лобі повинні відображатися деталі гри, такі як назва гри, кількість гравців, правила гри тощо;
- налаштування гри. Веб-застосунок повинен надавати інтерфейс для творців ігор для налаштування деталей гри, таких як правила гри, та кількість гравців;
- ігровий інтерфейс. Веб-застосунок повинен надавати зручний інтерфейс для гри у інтелектуальну гру «Мафія». Інтерфейс гри повинен відображати ігрову інформацію, таку як ролі гравців, результати голосування, таймер гри тощо;
- спілкування в режимі реального часу. Веб-застосунок повинен дозволяти гравцям спілкуватися один з одним в режимі реального часу за допомогою системи чату або голосового чату;

- елементи керування модератором. Веб-застосунок повинен надавати модераторам можливість керувати ігровим потоком і забезпечувати чесний ігровий процес. Модератори повинні мати можливість ставити гру на паузу або завершувати її, вилучати гравців або призначати ролі;
- сповіщення. Веб-застосунок повинен надавати гравцям сповіщення про те, що гра ось-ось почнеться або коли настане їхня черга;
- рейтингова система. На сайті повинна бути система рейтингу для відстеження результатів гравців і відображення таблиці лідерів;
- мобільна сумісність. Веб-застосунок повинен бути оптимізований для мобільних пристроїв, щоб гравці могли грати на ходу;
- безпека. Веб-застосунок повинен забезпечувати безпеку даних користувачів і запобігати несанкціонованому доступу до сайту;
- кастомізація. Веб-застосунок повинен дозволяти творцям ігор змінювати правила гри та ролі відповідно до їхніх уподобань;
- система зворотного зв'язку. Веб-застосунок повинен мати систему зворотного зв'язку, яка дозволяє гравцям залишати відгуки та пропозиції щодо покращення.

### 3.3 Формування контрольних результатів

Згідно порядку проведення експерименту, застосували метод Вігерса для пріоритезації вимог з вхідних даних. Результати застосування методу Вігерса наведені у табл. 3.

Таблиця 3 – Контрольні результати

Вимога	Характеристика						
	Вигода	Шкода від відсутності	Сумарна цінність	Відносна цінність	Витрати	Відносні витрати	Показник пріоритету
Система зворотного зв'язку	5	3	8	0.066	2	0.034	1.934

Ігровий інтерфейс	9	9	18	0.148	5	0.085	1.741
-------------------	---	---	----	-------	---	-------	-------

Продовження таблиці 3

Вимога	Характеристика						
	Вигода	Шкода від відсутності	Сумарна цінність	Відносна цінність	Витрати	Відносні витрати	Показник пріоритету
Система зворотного зв'язку	5	3	8	0.066	2	0.034	1.934
Ігрове лобі	9	9	18	0.148	6	0.102	1.451
Спілкування в режимі реального часу	9	9	18	0.148	7	0.119	1.244
Аутентифікація користувачів	7	6	13	0.107	7	0.119	0.898
Елементи керування модератором	2	7	9	0.074	5	0.085	0.870
Кастомізація	3	4	7	0.057	4	0.068	0.846
Налаштування гри	4	6	10	0.082	6	0.102	0.806
Сповідання	2	4	6	0.049	4	0.068	0.725
Безпека	3	7	10	0.082	8	0.136	0.605
Рейтингова система	2	3	5	0.041	5	0.085	0.484
Мобільна сумісність	4	3	7	0.057	9	0.153	0.376

Відсортували результати застосування методу Вігера у порядку пріоритетності для отримання контрольних даних. Як бачимо, основна функціональність веб-застосунку (а саме – ігровий інтерфейс, ігрове лобі, спілкування у режимі реального часу) отримала високий пріоритет розробки. За рахунок малих витрат на реалізацію і великої вигоди, високий пріоритет також отримала система зворотного зв'язку.

Ці дані використаємо як контрольні результати для перевірки інших методів пріоритезації вимог до ІТ-продуктів.

### 3.4 Застосування методів пріоритезації вимог

#### 3.4.1 Метод MoSCoW

Застосували метод MoSCoW на вхідних даних з урахуванням усіх важливих факторів проєкту.

Під час застосування методу група стейкхолдерів доходила консенсусу з приводу категорії пріоритетності вимог, після чого обрана категорія фіксувалась як результат. Результати пріоритезації згідно методу MoSCoW представлені у табл. 4.

Таблиця 4 – Результат застосування методу MoSCoW

Категорія пріоритету	Вимога
Must have	Ігрове лобі
	Ігровий інтерфейс
	Спілкування в режимі реального часу
Should have	Аутентифікація користувачів
	Елементи керування модератором
	Система зворотного зв'язку
Could have	Налаштування гри
	Кастомізація
	Сповідання
	Безпека
Won't have	Мобільна сумісність
	Рейтингова система

До категорії must have віднесли вимоги, без яких неможливий ігровий процес – ігровий інтерфейс, ігрове лобі та спілкування в режимі реального часу.

Аутентифікація користувачів була внесена у категорію should have як функція, яка не є необхідною для ігрового процесу, але значно поліпшить особистий досвід користувача по використанню застосунку. Система зворотного зв'язку та елементи керування модератором віднесли до цієї

категорії з тієї причини, що обидві функції дозволять ефективніше адмініструвати веб-застосунок.

Налаштування гри, кастомізацію та сповіщення віднесли до категорії *could have* з тієї причини, що кожна з цих функцій може бути корисною для гравців, але напряду не впливають на можливість користування веб-застосунком. Безпека також віднесена до цієї категорії з тієї причини, що витрати бюджету на закриття усіх можливих вразливостей є завеликими, і в рамках розробки MVP не вважаємо це доречним.

Мобільна сумісність також була визнана недоречною витратою бюджету в рамках розробки MVP. Відносно вимог з безпеки, мобільна сумісність була визнана менш пріоритетною, тому вона віднесена до категорії *won't have*.

До категорії *won't have* також потрапила рейтингова система через значні витрати на автоматичне рахування рейтингу гравців щодо розробки алгоритму розрахунків.

### 3.4.2 Метод Buy-a-function

Для застосування методу Buy-a-function, один зі стейкхолдерів виступив в якості представника команди розробників, завдяки чому визначили вартість кожної з вимог. Для визначення вартості використали приблизні затрати часу в людино-годинах. Вартості вимог представлені у табл. 5.

Таблиця 5 – Вартості вимог для методу Buy-a-function

Вимога	Вартість
Ігрове лобі	300
Ігровий інтерфейс	250
Спілкування в режимі реального часу	350
Аутифікація користувачів	380
Елементи керування модератором	200

Система зворотного зв'язку	80
Налаштування гри	240

Продовження таблиці 5

Вимога	Вартість
Кастомізація	160
Сповідання	120
Безпека	400
Мобільна сумісність	420
Рейтингова система	220

Сумарна вартість усіх вимог становила 3120.

Бюджет, який був виділений кожному зі стейкхолдерів, що приймали участь у купівлі вимог, становив 1680, що дорівнює людино-годинам з розрахунку: 3 робочих місяці по 20 днів; 3 члена команди з 100% залученням на проєкті (8 годин на день) і 1 член команди з 50% залученням на проєкті (4 години на день).

Результати застосування методу наведені у табл. 6.

Таблиця 6 – Результат застосування методу Buy-a-function

Вимога	Витрати			
	Стейкхолдер 1	Стейкхолдер 2	Стейкхолдер 3	Сума
Спілкування в режимі реального часу	350	350	350	1050
Ігрове лобі	300	300	300	900
Аутентифікація користувачів	0	380	380	760
Ігровий інтерфейс	250	250	250	750
Елементи керування модератором	200	200	0	400
Система зворотного зв'язку	80	80	80	240
Сповідання	120	120	0	240
Налаштування гри	0	0	240	240

Рейтингова система	220	0	0	220
Кастомізація	160	0	0	160

Продовження таблиці 6

Вимога	Витрати			
	Стейкхолдер 1	Стейкхолдер 2	Стейкхолдер 3	Сума
Безпека	0	0	0	0
Мобільна сумісність	0	0	0	0
	1680	1680	1600	

Бачимо, що результати використання даного методу сильно залежать від оцінки вартості. Хоча вимоги з налаштування гри купив лише один стейкхолдер, вимоги зі сповіщення – двоє і вимоги зі зворотного зв'язку купили усі стейкхолдери – пріоритет при цьому у цих трьох вимог виявився однаковим за рахунок вартості. В такому випадку, більш пріоритетними вважаємо вимоги, які були куплені більшою кількістю стейкхолдерів.

Також, результати явно відображують суб'єктивність стейкхолдерів. Один зі стейкхолдерів вирішив витратити 380 очок на купівлю вимог з рейтингової системи та кастомізації замість аутентифікації користувачів, що значно вплинуло на результати пріоритезації, внаслідок участі групи з лише трьох стейкхолдерів.

### 3.4.3 Метод RICE

Для використання методу RICE, двоє з чотирьох стейкхолдерів виступили в якості представників бізнесу, які оцінювали параметри reach та impact, в той час як інші двоє виступили в якості представників команди розробки, які оцінювали параметри confidence та effort.

Оцінка відбувалась за консенсусом представників, відповідальних за оцінку параметрів.

Параметр Reach оцінювали за відносним показником того, на яку кількість користувачів впливатиме реалізація тих чи інших вимог. Наприклад, вимоги з приводу ігрового інтерфейсу отримали оцінку 10 згідно параметру Reach, так як реалізація впливатиме на всіх гравців. Тоді як вимоги до рейтингової системи отримали оцінку 2, так як більшості гравців рейтинг не цікавий.

Параметр Impact оцінювали відповідно до того, наскільки реалізація цієї вимоги важлива для успіху проєкту на стадії MVP. Вимоги, що стосуються безпосередньо ігрового процесу отримали 10, тоді як вимоги до аутентифікації користувачів отримали лише 8 – тому що ігровий процес можливий і без аутентифікації, а лише з використанням «гостьового» доступу.

Параметр Confidence оцінювався відносно складності розробки. Оцінка цього параметру враховувала, наскільки багато досвіду у розробників з розробки подібного функціоналу, а також, чи існують відкриті фреймворки, які можуть спростити розробку.

Параметр Effort оцінювався за допомогою відносної оцінки необхідних людино-годин для розробки.

Результати застосування методу наведені у табл. 7.

Таблиця 7 – Результат застосування методу RICE

Вимога	Оцінка параметрів				RICE Score
	Reach	Impact	Confidence	Effort	
Ігровий інтерфейс	10	10	10	5	200.000
Система зворотного зв'язку	5	6	10	2	150.000
Ігрове лобі	10	10	7	6	116.667
Спілкування в режимі реального часу	10	10	6	7	85.714
Аутентифікація користувачів	10	8	8	8	80.000
Елементи керування модератором	6	8	6	5	57.600
Сповідання	8	4	7	4	56.000
Налаштування гри	3	5	6	6	15.000

Кастомізація	5	2	5	4	12.500
Безпека	10	2	5	9	11.111
Рейтингова система	2	3	7	5	8.400
Мобільна сумісність	6	3	4	10	7.200

Як бачимо, кожен з параметрів значно впливає на RICE Score. Важливі вимоги з високими затратами на розробку отримують низький пріоритет, тоді як вимоги середньої важливості, але які вимагають невеликих зусиль – отримують вищий пріоритет.

В цілому, пріоритетність вимог майже не змінилась порівняно з попередніми методами.

#### 3.4.4 Метод зваженої оцінки

Використання методу передбачає попереднє визначення критеріїв, згідно яких буде відбуватись пріоритезація і визначення відносної ваги кожного з критеріїв.

Критерії, які були обрані для пріоритезації згідно методу зваженої оцінки, та їх відносна вага:

- вплив на ігровий процес – 0.3; найвища оцінка за цим критерієм означає, що без реалізації даної вимоги ігровий процес неможливий, найнижча – вимога ніяк не впливає на ігровий процес;
- вподобання користувачів – 0.15; найвища оцінка за цим критерієм означає, що користувачам точно сподобаються можливості, які надає реалізація даної вимоги, найнижча – користувачам байдуже на реалізацію цієї вимоги;
- швидкість розробки – 0.2; найвища оцінка за цим критерієм означає, що реалізація даної вимоги займе не більше тижня, найнижча – витрати часу настільки великі, що їх складно оцінити;

- вигода для компанії – 0.35; найвища оцінка за цим критерієм означає, що якісна реалізація вимоги гарантує привернення уваги нових користувачів або є корисною для майбутньої розробки, найнижча – реалізація вимоги матиме мінімальний вплив на успішність проєкту.

Кожен з критеріїв оцінюється за шкалою від 1 до 5, де 1 – найменша оцінка згідно критерію, а 5 відповідно – найкраща.

Після оцінювання, підсумовуємо добутки оцінки за критерієм та ваги критерію і в результаті отримуємо оцінку пріоритетності вимоги у проміжку від 1 до 5.

Результати застосування методу наведені у табл. 8.

Таблиця 8 – Результат застосування методу зваженої оцінки

Вимога	Критерій				Оцінка
	Вплив на ігровий процес	Вподобання користувачів	Швидкість розробки	Вигода для компанії	
	0.3	0.15	0.2	0.35	
Ігровий інтерфейс	5	2	3	5	4.15
Ігрове лобі	5	2	3	5	4.15
Спілкування в режимі реального часу	5	3	2	5	4.1
Елементи керування модератором	4	4	3	3	3.45
Система зворотного зв'язку	1	3	5	4	3.15
Аутентифікація користувачів	3	3	2	3	2.8
Налаштування гри	4	3	2	2	2.75
Кастомізація	2	5	3	2	2.65
Сповіщення	3	4	4	1	2.65
Безпека	3	2	1	3	2.45
Рейтингова система	1	5	3	1	2
Мобільна сумісність	2	3	1	2	1.95

Завдяки тому, що найбільша вага була надана критеріям, що відображають вплив від реалізації вимоги на ігровий процес і на успіх проєкту (вигоду для компанії), найвищий пріоритет був наданий тим вимогам, які є

критичними для успіху проєкту. Тоді як пріоритет вимог до аутентифікації користувачів та системи зворотнього зв'язку став трішки нижчим, аніж при застосуванні інших методів.

### 3.4.5 Модель Кано

Для застосування моделі Кано провели опитування невеликої вибірки потенційних користувачів веб-застосунку для онлайн гри в інтелектуальну гру «Мафія».

Опитування складалось з 24 питань – на кожен вимогу було зроблено два парних запитання – відчуття від присутності і відсутності даної функції.

Після проведення опитування, обробили результати опитування згідно таблиці 1, внаслідок чого отримали агреговані результати, відображені в табл. 9.

Таблиця 9 – Результат застосування моделі Кано

Вимога	Категорія					
	М	Р	А	І	Р	Q
Спілкування в режимі реального часу	8	4	0	0	0	0
Ігрове лобі	4	2	4	2	0	0
Ігровий інтерфейс	3	4	4	1	0	0
Рейтингова система	3	2	3	4	0	0
Елементи керування модератором	3	1	4	4	0	0
Аутентифікація користувачів	2	4	2	4	0	0
Система зворотного зв'язку	2	1	4	5	0	0
Сповідання	2	0	7	3	0	0
Безпека	1	3	4	4	0	0
Мобільна сумісність	0	2	3	7	0	0
Налаштування гри	0	0	3	6	3	0
Кастомізація	0	0	3	5	4	0

У таблиці відображена кількість респондентів, за відповідями яких вимога належить до тієї або іншої категорії. Всього в опитуванні прийняли

участь 12 респондентів. Для визначення пріоритету кожної з вимог, відсортували записи з наступними налаштуваннями (у порядку важливості):

- за збільшенням кількості відповідей категорії reverse (R);
- за зменшенням кількості відповідей категорії must-do (M);
- за зменшенням кількості відповідей категорії performance (P);
- за зменшенням кількості відповідей категорії attractive (A);
- за збільшенням кількості відповідей категорії indifferent (I).

Таким чином, найбільший пріоритет ми віддаємо тим вимогам, які є необхідними для найбільшої кількості користувачів, а найменший пріоритет – тим вимогам, які відлякують частину гравців.

Як бачимо, за результатами застосування методу, пріоритет вимог з реалізації рейтингової системи значно вищий, аніж при використанні інших методів, в той час як пріоритет вимог щодо аутентифікації та системи зворотного зв'язку – навпаки, менший. Також, значно впав пріоритет вимог з налаштування та кастомізації, так як деяких користувачів ці функції відштовхують. З цього можна зробити висновок, що інтереси стейкхолдерів і користувачів частково не співпадають, або ж на результати повпливала суб'єктивність опитаних респондентів.

### 3.5 Висновок з проведення експерименту

Для того щоб зробити висновки, порівнюємо контрольні дані, отримані за методом Вігера, з даними, отриманими в ході застосування інших методів пріоритезації. Для порівняння використаємо таблицю, в яку внесемо пріоритет кожної з вимог за різними методами. Табл. 10 відображає порівняння результатів застосування різних методів пріоритезації.

Таблиця 10 – Порівняння пріоритетів, отриманих в ході проведення експерименту

Вимога	Пріоритетність згідно методу			
	Контрольна	MoSCoW	Buy-a-function	RICE
Система зворотного зв'язку	1	4-6	6	2
Ігровий інтерфейс	2	1-3	4	1
Ігрове лобі	3	1-3	2	3

Продовження таблиці 10

Вимога	Пріоритетність згідно методу			
	Контрольна	MoSCoW	Buy-a-function	RICE
Безпека	10	7-10	11	10
Рейтингова система	11	11-12	9	11
Мобільна сумісність	12	11-12	12	12
Спілкування в режимі реального часу	4	1-3	1	4
Аутентифікація користувачів	5	4-6	3	5
Елементи керування модератором	6	4-6	5	6
Кастомізація	7	7-10	10	9
Налаштування гри	8	7-10	7	8
Сповідання	9	7-10	8	7

Продовження таблиці 10

Вимога	Пріоритетність згідно методу		
	Контрольна	Зважена оцінка	Модель Кано
Система зворотного зв'язку	1	5	7
Ігровий інтерфейс	2	1	3
Ігрове лобі	3	2	2
Спілкування в режимі реального часу	4	3	1
Аутентифікація користувачів	5	6	6
Елементи керування модератором	6	4	5

Кастомізація	7	8	12
Налаштування гри	8	7	11
Сповідання	9	9	8
Безпека	10	10	9
Рейтингова система	11	11	4

Кінець таблиці 10

Вимога	Пріоритетність згідно методу		
	Контрольна	Зважена оцінка	Модель Кано
Мобільна сумісність	12	12	10

З результатів порівняння можемо зробити висновок щодо відносної ефективності застосування методів.

Результати кожного з методів ідентифікують одні і ті самі вимоги в якості найпріоритетніших, а саме – вимоги до ігрового інтерфейсу, ігрового лобі та спілкування в режимі реального часу. Також, результати застосування кожного з методів явно відображають найменш пріоритетні вимоги для успіху проекту, а саме вимоги до мобільної сумісності.

Результат застосування методу MoSCoW виявився ефективним, так як пріоритети майже цілком співпадають з контрольними даними, не дивлячись на нечіткість пріоритетів. Це підтверджує гіпотезу, що метод MoSCoW є ефективним за умови маленької кількості вимог, висунуту в підрозділі 2.2 в рамках характеристики методу MoSCoW.

Результат застосування методу Buy-a-function визначають пріоритетними (з 1 по 6) ті самі вимоги, що були визначені пріоритетними згідно контрольного методу, хоча відносна пріоритетність цих вимог відрізняється.

Результати застосування методів, що спираються на оцінку декількох факторів для визначення пріоритету вимог, а саме методів RICE та зваженої оцінки, виявилися майже однозначними одне одному та контрольним даним.

Це підтверджує порівняльну характеристику, надану цим двом методам у підрозділі 2.2, яка також була майже ідентичною.

Результат застосування моделі Кано значно відрізняється від контрольних результатів. Це пояснюється тим фактом, що орієнтованість методу, застосованого для контрольних результатів, та орієнтованість моделі Кано – різна (внутрішня і зовнішня відповідно) і підтверджує гіпотезу, висунуту в підрозділі 2.2 про те, що зовнішня орієнтованість значно виділяє модель Кано на фоні інших досліджених методів. Не дивлячись на значну різницю в пріоритетах згідно моделі Кано та контрольних результатів, найважливіші пріоритети співпадають (з урахуванням похибки на вимоги до рейтингової системи), що дає нам змогу зробити висновок про ефективність метода.

Результати експерименту доводять робочу гіпотезу про ефективність кожного з методів. Можемо зробити висновок, що використання кожного з досліджених методів у практичній ситуації цілком залежить від тих факторів, на які команда розробки має спиратись під час процесу пріоритезації.

## 4 ВИКОРИСТАННЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

Зважаючи на актуальність проблеми пріоритизації вимог до ІТ-продукту і різноманіття методів, які були розроблені для вирішення цієї проблеми, результати дослідження в рамках даної роботи можуть бути широко застосовані у сфері управління проектами з розробки ІТ-продуктів.

Результати дослідження в рамках підрозділу 2.1 даної роботи можуть бути використані для надання характеристики будь-якому методу пріоритизації вимог до ІТ-продуктів. Надані характеристики дадуть змогу оцінити доречність метода пріоритизації у практичній ситуації не вимагаючи від особи, що займається впровадженням, детального дослідження кожного з характеризованих методів. Такий підхід до вибору методу пріоритизації може бути широко використаний на проектах, що не мають впровадженого і формалізованого методу пріоритизації вимог.

Також, використання характеристик, наданих у підрозділі 2.1 може стати основою для майбутніх досліджень методів пріоритизації.

Переваги і недоліки досліджених методів, наведені у підрозділах 1.2.1-1.2.5, а також порівняльний аналіз методів пріоритизації, наведений у підрозділі 2.2, можуть бути використані в якості:

- посібника для оцінки доречності досліджених методів, а саме методів MoSCoW, RICE, Buy-a-feature, моделі Кано та методу зваженої оцінки, при впровадженні метода на проекті з розробки ІТ-продукту;
- основи для більш глибокого наукового дослідження одного або декількох з наведених методів пріоритизації.

Результати експериментального дослідження, що наведені у розділі 3 даної роботи, можуть виступати в якості доказової бази ефективності досліджених методів пріоритизації вимог до ІТ-продукту при впровадженні будь-якого з розглянутих методів на проекті з розробки ІТ-продукту.

## ВИСНОВКИ

В рамках роботи було проведено дослідження методів пріоритезації вимог до ІТ-продуктів, що включало в себе:

а) вивчення та систематизація існуючих відомостей про методи пріоритезації MoSCoW, RICE, Buy-a-function, модель Кано та метод зваженої оцінки;

б) аналіз розглянутих методів на предмет переваг та недоліків;

в) визначення загальних характеристик методів пріоритезації вимог до ІТ-продуктів;

г) порівняльний аналіз розглянутих методів пріоритезації згідно визначених характеристик;

д) експериментальне дослідження розглянутих методів пріоритезації на предмет ефективності застосування.

Розділ 1 даної роботи був присвячений вивченню, систематизації та аналізу методів пріоритезації вимог до ІТ-продуктів. Був наданий вичерпний перелік переваг та недоліків кожного з розглянутих методів. Результати дослідження в рамках розділу 1 лягли в основу досліджень в рамках розділів 2 і 3.

Розділ 2 даної роботи був присвячений визначенню загальних характеристик методів пріоритезації і їх використанню для проведення порівняльного аналізу розглянутих у розділі 1 методів. Зокрема, під час проведення порівняльного аналізу, були висунуті гіпотези щодо ефективності використання розглянутих методів залежно від умов, що були доведені в розділі 3 в рамках дослідження.

Розділ 3 даної роботи був присвячений експериментальному дослідженню розглянутих методів на предмет ефективності використання. За результатами експериментального дослідження дійшли висновків щодо ефективності кожного з методів. Також, результати дослідження показали, що,

не дивлячись на ефективність розглянутих методів, доречність їх використання сильно залежить від критеріїв, які вважаються найбільш важливими для компанії-розробника і, зокрема, в рамках проєкту з розробки ІТ-продукту.

В розділі 4 даної роботи були наведені приклади щодо використання результатів даної роботи на практиці.

Розглянута тема є великою та може досліджуватись далі в рамках наукової роботи. Наприклад, наступні наукові роботи можуть розглянути такі теми як визначення типів проєктів, для яких використання методів з певними характеристиками є доречним або недоречним, або дослідження методів пріоритезації, що не були розглянуті в рамках даної роботи.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Методичні вказівки щодо розробки та оформлення магістерської кваліфікаційної роботи за спеціальністю 122 Комп'ютерні науки (освітня програма «Управління проєктами в галузі інформаційних технологій» освітньо-кваліфікаційного рівня «магістр» / Упоряд.: Петров К.Е., Левикін В.М., Чалий С.Ф., Євланов М.В., Саєнко В.І., Міхнов Д.К., Міхнова А.В., Чала О.В. – Харків: ХНУРЕ, 2021. – 28 с
2. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура і правила оформлювання. – Чинний від 22.06.2015. – Київ: ДП «УкрНДНЦ», 2016. – 31
3. ДСТУ 8302:2015. Інформація та документація. Бібліографічні посилання. Загальні положення та правила складання. – Чинний від 04.03.2016. – Київ: ДП «УкрНДНЦ», 2016. – 20 с.
4. Chen, T., Zhang, Z., & Zou, Y. (2018). AHP-Based Requirements Prioritization: A Case Study. *IEEE Software*.
5. Hussain, M., Ali, M., & Ali, M. (2018). A Comparative Analysis of Requirement Prioritization Techniques. *Proceedings of the 11th International Conference on Emerging Ubiquitous Systems and Pervasive Networks*.
6. Hong, S. W., Kim, K., & Lee, J. H. (2019). Evaluating the Kano Model for Prioritizing User Requirements: A Case Study. *International Journal of Industrial Engineering Computations*.
7. Khan, M. U., Isa, N. A. M., & Mohamad, N. R. (2019). Comparison of Prioritization Techniques for Requirements Engineering: A Systematic Review. *Journal of Engineering Science and Technology*.
8. Ali, S., Abid, S. S., & Iqbal, H. (2019). An Analysis of Software Requirements Prioritization Techniques. *Proceedings of the 7th International Conference on Frontiers of Information Technology*.

9. Zaman, I., Khan, M. A., & Hussain, S. (2019). An Investigation of Prioritization Techniques for Software Requirements. Proceedings of the 2019 International Conference on Computer and Information Sciences.

10. Niazi, M. A., Malik, F. A., & Malik, A. I. (2018). A Comparative Analysis of Requirement Prioritization Techniques. Proceedings of the 4th International Conference on Innovations in Computer Science and Engineering.

11. Khan, M. A., & Ahmad, R. J. (2019). Comparative Study of Requirements Prioritization Techniques in Agile Software Development. Proceedings of the 3rd International Conference on Computer and Information Sciences.

12. Reinheimer, D., & Ritter, C. (2018). A Comprehensive Overview of Requirements Prioritization Techniques. Journal of Computing Sciences in Colleges.

13. Sedghi, H., Salehi-Abari, A., & Bagheri, E. (2018). Requirements Prioritization for Agile Software Development: A Systematic Literature Review. Journal of Systems and Software.

14. Januszek, O., & Kocur, D. (2020). A Comparative Study of Requirements Prioritization Techniques for Software Development. Journal of Software Engineering and Applications.

15. Khorsandi, O., & Naji, A. A. (2018). Prioritization of Requirements in Agile Software Development: A Comparative Analysis. Journal of Software Engineering and Applications.