

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 виявлення дублювання функціональних вимог до ІТ - продукту у вигляді математичних виразів»  
(тема)

Виконав:  
Студент 2 курсу, групи УПГІТм-20-1  
Кадигроб В.А.  
(прізвище, ініціали)

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

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

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

Керівник проф. Євланов М.В.  
(посада, прізвище, ініціали)

Допускається до захисту  
Зав. кафедри

\_\_\_\_\_ Петров К.Е.  
(підпис) (прізвище, ініціали)

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-професійна \_\_\_\_\_  
(освітньо-професійна або освітньо-наукова)  
Освітня програма Управління проектами в галузі інформаційних технологій  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_

(підпис)

«\_\_\_\_\_» \_\_\_\_\_ 2021 р.

### ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові \_\_\_\_\_ Кадигробу Владиславу Андрійовичу \_\_\_\_\_  
(прізвище, ім'я, по батькові)

1. Тема роботи «Дослідження методів виявлення дублювання функціональних вимог до ІТ -продукту у вигляді математичних виразів». затверджена наказом по університету від 05.11.2021 р. № 1646 Ст
2. Термін подання студентом роботи до екзаменаційної комісії 14.12.2021 р.
3. Вихідні дані до роботи науково-технічні публікації, джерела інтернету, що стосуються теми атестаційної роботи.
4. Перелік питань, що потрібно опрацювати в роботі: Вступ; Аналіз існуючих методів виявлення дублювання функціональних вимог; Дослідження методів виявлення дублювання функціональних вимог; Розробка вдосконаленого методу виявлення дублювання функціональних вимог; Практичне використання вдосконаленого методу; Висновки.

## КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз матеріалів з теми роботи	08.11.2021 - 12.11.2021	виконано
2	Постановка задачі атестаційної роботи	13.11.2021 - 15.11.2021	виконано
3	Обробка матеріалу	16.11.2021 - 17.11.2021	виконано
4	Попереднє дослідження галузі завдання	18.11.2021 - 21.11.2021	виконано
5	Аналіз існуючих методів виявлення дублювання функціональних вимог	22.11.2021 - 27.11.2021	виконано
6	Дослідження розробленого методу виявлення дублювання функціональних вимог	28.11.2021 - 03.12.2021	виконано
7	Практичне використання вдосконаленого методу виявлення дублювання функціональних вимог	04.12.2021 - 07.12.2021	виконано
8	Оформлення пояснювальної записки та графічного матеріалу	08.12.2021 - 10.12.2021	виконано
9	Підготовка презентації	04.12.2021 - 07.12.2021	виконано
10	Подання студентом роботи для перевірки на плагіат	08.12.2021 - 09.12.2021	виконано
11	Подання студентом роботи для перевірки на нормоконтроль	10.12.2021 - 11.12.2021	виконано
12	Надання роботи на підпис науковому керівнику	12.12.2021 - 13.12.2021	виконано
13	Попередній захист роботи	14.12.2021	виконано
14	Захист	15.12.2021	

Дата видачі завдання \_\_\_\_\_ 2021 р.

Студент \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_ Д.Т.Н., доцент Євланов  
М.В.

(підпис)

(посада, прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи містить: 66 с., 2 рис., 10 табл., 27 джерел, 1 додаток.

АНАЛІЗ ВИМОГ, ДУБЛЮВАННЯ ВИМОГ, ІНФОРМАЦІЙНА СИСТЕМА, ПУБЛІКАЦІЯ ВИМОГИ, ФУНКЦІОНАЛЬНА ВИМОГА.

Мета магістерської роботи – провести дослідження можливостей застосування методу виявлення дублювання математичних описів вимог до інформаційної системи для різних варіантів математичних описів вимог.

Об'єктом дослідження в рамках магістерської кваліфікаційної роботи, є дослідження методів виявлення дублювання функціональних вимог до ІТ - продукту у вигляді математичних виразів для замовника.

Предметом дослідження, є методи виявлення дублювання функціональних вимог до ІТ-продукту.

Теоретичними результатами дослідження є описи удосконаленого методу оптимізації виявлення дублювання при виконанні робіт.

Практичними результатами, є впровадження удосконаленого методу оптимізації виявлення дублювання при виконанні робіт у систему замовника.

Новизна дослідження полягає у вивченні існуючих методів в розробці, на підставі отриманої інформації, удосконаленого методу виявлення дублювання функціональних вимог до ІТ -продукту у вигляді математичних виразів.

## ABSTRACT

The explanatory note to the qualification work contains: 66 pages, 2 images, 10 tables, 27 sources, 1 attachment.

### ANALYSIS OF REQUIREMENTS, DUPLICATION OF REQUIREMENTS, INFORMATION SYSTEM, PUBLICATION OF REQUIREMENTS, FUNCTIONAL REQUIREMENT.

The purpose of the master's work is to study the possibilities of applying the method of detecting duplication of mathematical descriptions of requirements to the information system for different variants of mathematical descriptions of requirements.

The object of research in the master's qualification work is the study of methods for identifying duplication of functional requirements for the IT product in the form of mathematical expressions for the customer.

The subject of research is methods for detecting duplication of functional requirements for the IT product.

The theoretical results of the study are descriptions of an improved method for optimizing the detection of duplication in the performance of work.

The practical results are the introduction of an improved method for optimizing the detection of duplication when performing work in the customer's system.

The novelty of the study is the study of existing methods in the development, based on the information obtained, an improved method of detecting duplication of functional requirements for the IT product in the form of mathematical expressions.

## ЗМІСТ

Скорочення та умовні позначки.....	8
Вступ.....	9
1 Аналіз сучасного стану моделей і методів аналізу вимог до інформаційної системи.....	10
1.1 Дослідження моделей вимог до інформаційної системи.....	10
1.2 Дослідження сучасних моделей і методів аналізу вимог до ІТ-продуктів.....	14
1.3 Постановка задачі дослідження.....	17
2 Дослідження існуючих методів виявлення дублювання функціональних вимог до ІТ-продукту.....	21
2.1 Дослідження особливостей методу сценарного аналізу.....	21
2.2 Дослідження особливостей вдосконаленого методу синтезу варіантів опису архітектури створюваної інформаційної системи.....	24
2.3 Висновки до другого розділу.....	27
3 Дослідження можливості використання існуючих методів для виявлення дублювання функціональних вимог, опублікованих як математичні вирази.....	29
3.1 Дослідження можливості використання вдосконаленого методу синтезу варіантів опису архітектури створюваної інформаційної системи.....	29
3.2 Дослідження можливості використання базового методу сценарного аналізу.....	30
3.3 Порівняльний аналіз переваг та недоліків застосування досліджених методів.....	32
3.4 Висновки до третього розділу.....	35
4 Практична апробація отриманих результатів.....	36

4.1	Опис потреб користувачів та функціональних вимог, які підлягають аналізу.....	36
4.2	Результати застосування вдосконаленого методу синтезу варіантів опису архітектури створеної інформаційної системи.....	39
4.3	Перевірка отриманих результатів.....	43
4.4	Висновки до четвертого розділу.....	46
	Висновки.....	48
	Перелік джерел посилання.....	50
	Додаток А. Графічний матеріал кваліфікаційної роботи на тему «Дослідження методів виявлення дублювання функціональних вимог до ІТ -продукту у вигляді математичних виразів».....	54

## СКОРОЧЕННЯ, ТА УМОВНІ ПОЗНАКИ

АІС - автоматизована інформаційна система;

ІБ - Інформаційна безпека;

ІС - інформаційна система;

ОС – операційна система;

ПЗ – програмне забезпечення;

ПК – персональний комп'ютер.

## ВСТУП

Відповідно до визначення поняття «система», наведеного в [1], будь-який ІТ-продукт можна представити у вигляді комбінації взаємодіючих функцій, причому ця взаємодія здійснюється шляхом передачі та прийому структурованих масивів даних. Виявлення суперечливих функціональних вимог до ІТ-продукту на ранніх стадіях ІТ-проекту дозволяє зменшити витрати на реалізацію ІТ-проекту для створення цього ІТ-продукту. Ці витрати передусім включають час, витрачений на виявлення конфліктуючих функцій ІТ-продукту, усунення виявлених невідповідностей та перепроєктування функціональної структури, бази даних та програмного забезпечення ІТ-продукту. Тому дослідження в галузі автоматизації роботи з аналізу функціональних вимог до ІТ-продукту і, зокрема, роботи з аналізу вимог до ІТ-продукту на узгодженість слід визнати актуальними.

Метою кваліфікаційної роботи є дослідження методів виявлення дублювання функціональних вимог до ІТ –продукту, опублікованих у вигляді математичних виразів.

Кваліфікаційна робота виконується відповідно до вимог методичних вказівок з організації та виконання кваліфікаційної роботи [2] та національного стандарту ДСТУ 3008:2015 «Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлення» [3].

Оформлення переліку джерел посилання здійснюється відповідно до національного стандарту ДСТУ 8302:2015 «Бібліографічне посилання. Загальні положення та правила складання» [4].

# 1 АНАЛІЗ СУЧАСНОГО СТАНУ МОДЕЛЕЙ І МЕТОДІВ АНАЛІЗУ ВИМОГ ДО ІНФОРМАЦІЙНОЇ СИСТЕМИ

## 1.1 Дослідження моделей вимог до інформаційної системи

Сучасні уявлення терміну «вимога» базуються на визначенні, яке зафіксовано у стандарті ISO/IEC/IEEE 29148:2018. За цим визначенням вимога – це твердження, яке перекладає або виражає яку-небудь потребу та пов'язані з нею обмеження та умови. Під терміном «обмеження» у стандарті ISO/IEC/IEEE 29148:2018 розуміються якісь граничні умови, які накладаються на систему, її дизайн або реалізацію, та які можуть обмежувати або змінювати систему, її дизайн або реалізацію. Під терміном «умови» у стандарті ISO/IEC/IEEE 29148:2018 розуміються вимірювані кількісні або якісні ознаки вимоги, які вказують на обставини або події, за якими ця вимога використовується.

У стандарті ISO/IEC/IEEE 29148:2018 до цього визначення терміну «вимога» надано такі доповнення:

- а) вимоги існують на різних рівнях в структурі системи;
- б) вимога – це вираз однієї або кількох конкретних потреб дуже конкретним, точним та однозначним образом;
- в) вимога завжди відноситься до системи, програмного забезпечення або послуги або іншому об'єкту, до якого виникає зацікавленість.

Зараз існують такі засоби опису вимог:

- а) у вигляді текстів, написаних природною мовою, або структурованих текстових документів;
- б) у вигляді візуальних моделей;
- в) у вигляді текстів, написаних формальною мовою (у тому числі у вигляді одноразових чи багаторазових програмних прототипів).

Як правило, під час створення ІТ-продукту зазвичай використовують описи вимог, які зроблено, як мінімум, двома з розглянутих засобів.

Такий підхід до представлення вимог дозволяє ввести термін «модель вимог». Цей термін означає комплекс описів вимог тими засобами, які дозволяють на основі даних описів виконувати необхідні роботи в межах життєвого циклу ІТ-продукту з використанням конкретних методологій та інформаційних технологій [5].

Класичною моделлю вимог до такого різновиду ІТ-продуктів, як інформаційні системи (ІС), є модель вимог, яка була розроблена в межах методології SSADM. Ця модель являє собою набір описів вимог, наведений у табл. 1.1. [6, 7].

Таблиця 1.1 – Описи вимог до системи, які утворюють модель вимог методології SSADM

Засіб представлення (модель або документ)	Предмет опису в моделі вимог	Призначення опису
Документ «Каталог вимог»	Повний опис вимоги мовою учасників ІТ- проекту створення системи	Централізоване представлення основних аспектів вимоги до системи
Реляційна модель даних	Дані, які входять, виходять та оброблюються в ІС	Опис даних, які харак- теризують інформаційне середовище, що оточує ІС, і саму ІС
Модель потоків даних	Хто використовує систему, та які дії вони в ній здійснюють	Опис використання системи

Кінець таблиці 1.1

Засіб представлення (модель або документ)	Предмет опису в моделі вимог	Призначення опису
Об'єктно-подійні моделі	Робочий процес і обмін відомостями в процесі взаємодії процесів обробки даних і сховищ даних	Опис шляхів доступу процесів обробки даних до даних
Логічні моделі даних	Обмін відомостями в процесі взаємодії процесів обробки даних і сховищ даних	Опис шляхів доступу запитів до даних
Прототипи	Упрощений опис вимоги мовою програмування	Подальше дослідження і перевірка правильності вимоги
Документ «Каталог користувача»	Хто використовує систему	Опис користувачів системи
Матриця «Роль користувача / вимога»	Хто які вимоги висуває до системи	Опис користувачів як джерел вимог

Зараз найбільш розповсюдженою є сценарна модель вимог. Прикладом такої моделі є модель вимог компанії «Microsoft». Вона зводиться до набору описів вимог у вигляді візуальних моделей і текстових документів, які супроводжують ці моделі. Цей набір описів розглянуто в табл. 1.2 [5].

Як можливу заміну моделей, що описують окремі аспекти вимог до системи, «Microsoft» пропонує розглядати прототипи окремих функцій системи, які розробляються в упрощеній формі на початкових стадіях ІТ-проекту [5].

Таблиця 1.2 – Описи вимог до системи, які утворюють модель вимог компанії «Microsoft»

Засіб представлення (модель або документ)	Предмет опису в моделі вимог	Призначення опису
Модель варіантів використання	Хто використовує систему, які дії вони в ній здійснюють	Опис використання системи
Модель діяльності	Робочий процес і обмін відомостями в процесі взаємодії користувачів та системи або її частин	Відображення робочого процесу між користувачами і системою
Модель послідовностей	Послідовність взаємодій між користувачами і системою або її частинами (альтернативне пред- ставлення моделі діяльності)	Відображення взаємодій між користувачами та системою
Додаткові документи або робочі елементи	Критерії продуктивності, безпеки, корисності та надійності	Опис вимог до якості обслуговування
Додаткові документи або робочі елементи	Обмеження і правила, які не відносяться до конкретного варіанту використання	Відображення бізнес- правил

## 1.2 Дослідження сучасних моделей і методів аналізу вимог до ІТ-продуктів

Основне розуміння методів і методів аналізу вимог було сформульовано наприкінці ХХ - початку ХХІ ст. Приклади цих методів наведено в [8]. Однак ефективність цих методів на практиці часто виявляється недостатньою. У той же час використання сучасних методів Agile для управління роботою з вимогами суттєво не підвищує ефективність застосування методів аналізу вимог [9].

Сучасні дослідження моделей та методів аналізу вимог до ІТ-продуктів, у тому числі – функціональних вимог, здійснюються у двох основних напрямках. Один із цих напрямків передбачає дослідження та створення методів аналізу вимог, заснованих на взаємодії учасників ІТ-проекту. Так, у [10] пропонується метод аналізу вимог, який допомагає подолати комунікаційний розрив між замовником та розробником. Запропонований у [11] метод аналізу вимог до розробки програмного забезпечення ґрунтується на спільній участі представників стейкхолдерів ІТ-проекту. Однак наведені в [12] результати показують, що подібні методи можна покращити шляхом структурування сприйняття створюваної системи учасниками ІТ-проекту у формальній концептуальній ментальній моделі. Ця модель є візуальним втіленням переконань і очікувань користувача створюваної системи. Перевагою цієї моделі слід визнати, як показано в [12], покращення внаслідок її застосування правильності, повноти та передбачуваності сценаріїв виконання вимог. Однак дана модель є також додатковим джерелом можливих помилок, які виникають через розрив між сприйняттями вимоги, що формується замовником і аналітиком. Крім того, створення подібної моделі потребує додаткових витрат часу, що призводить до вкрай небажаного збільшення тривалості ІТ-проекту.

Ще одним способом формального опису поведінки користувачів є застосування математичного апарату теорії категорій, який покладено в основу запропонованої у [13] спеціальної декларативної мови. Перевагою цього підходу можна вважати можливість проведення автоматичного семантичного аналізу сценаріїв виконання вимог. Однак цей підхід не позбавлений серйозних вад. Головним із цих недоліків слід визнати орієнтацію розглянутого в [13] підходу до формального опису поведінки на вирішення завдання аналізу виключно публікацій окремих вимог конкретного ІТ-проекту. Даний підхід не дозволяє надалі застосовувати накопичений досвід у наступних ІТ-проектах аналогічного призначення.

Використання для опису робіт з аналізу вимог методів сучасної кібернетики на основі представлення процесу встановлення вимог до даних як системи управління із зворотним зв'язком розглянуто у [14]. Головною перевагою даного підходу слід визнати можливість автоматизації робіт зі збору вимог до експлуатованих ІС за умови, що описи сервісів таких ІС та їх поведінки сильно формалізовані. Однак застосування такого підходу у практиці управління експлуатацією пов'язане з цілою низкою проблем. Зокрема, залишається незрозумілою множина змінних, які мають точно описувати керовані послуги ІС, запити на їх зміну, сформульовані з урахуванням цих запитів вимоги і завдання управління експлуатацією ІС загалом. Погано узгоджена частина пропозиції, що допускає різні тлумачення. Зокрема, залишається незрозумілою множина змінних, які повинні точно описувати:

- а) експлуатовані послуги ІС;
- б) запити на зміну експлуатованих сервісів ІС;
- в) вимоги до нових версій сервісів ІС, які сформульовані на основі запитів на зміну сервісів ІС, що експлуатуються;
- г) завдання управління експлуатацією ІС загалом.

Інший напрямок досліджень передбачає розробку методів виявлення та аналізу вимог, в основу яких покладено формальні описи цих вимог. Так, в

[15] запропоновано аналізувати моделі поведінки програмного забезпечення на основі їх ймовірнісних описів. Перевагою цього методу є можливість проведення кількісного аналізу вимог на основі ймовірнісного моделювання поведінки мобільного програмного забезпечення. Однак цей і подібні до нього методи вимагають накопичення досить великого масиву даних для статистичних досліджень можливих гіпотез закономірностей. У той самий час створення нових, модернізація чи розвиток існуючих ІС неминуче призводить до нових вимог, зумовлених розвитком бізнес-процесів автоматизованих підприємств чи організацій. Такі нові вимоги далеко не завжди можуть бути описані на основі раніше відомих закономірностей або гіпотез щодо поведінки програмного забезпечення ІС. Тому даний варіант аналізу вимог до ІС поки не набув досить широкого поширення.

Найбільш поширеним способом формального опису вимог до ІТ-продуктів слід визнати знання-орієнтовані описи. При цьому значну увагу приділяють удосконаленню вже існуючих моделей та методів аналізу вимог [16]. Зокрема, [17] розглядається варіант порівняльного аналізу елементів системи на основі їх описів у вигляді діаграм Use Case, Activity і діаграм потоків даних. Перетворення публікацій вимог стейкхолдерів у модель системи, що виконується на основі діаграм Activity і State UML, розглянуто в [18]. Процедурна модель специфікації функціональних вимог до програмного забезпечення запропонована у [19]. У [20] розглянуто модель інноваційного сервісного проектування, використання якої в ході роботи з вимогами передбачає поєднання формальних знань-орієнтованих методів та неформальних методів теорії вирішення винахідницьких завдань. У цілому розглянуті в [17-20] моделі та методи є окремими випадками одного й того ж підходу. Цей підхід передбачає виділення та подальший аналіз знання-орієнтованих описів вимог до ІС на основі публікацій цих вимог у вигляді існуючих візуальних моделей. Головною перевагою методів аналізу, що ґрунтуються на цьому підході, слід визнати можливість кількісного оцінювання характеристик аналізованих вимог. Таке оцінювання дозволяє

підвищити об'єктивність значень характеристик вимог, що формуються аналітиком. Іншою, не менш важливою перевагою, є використання як формальних описів аналізованих вимог їх знання-орієнтованих моделей. Подібні моделі дозволяють забезпечити багаторазове використання описів окремих вимог, у тому числі у різних ІТ-проектах.

Однак такий підхід і, отже, засновані на ньому моделі та методи, розглянуті в [17–20], мають низку недоліків. Ці недоліки значною мірою обмежують застосування цих методів. Серед цих недоліків насамперед слід виділити [21]:

а) практично повну відсутність досліджень формальних описів характеристик аналізованих вимог;

б) прив'язку формальних описів вимог та методів їх аналізу до конкретних візуальних уявлень цих вимог, що обмежує сферу використання методів аналізу.

Тому слід визнати необхідність досліджень, спрямованих на вдосконалення існуючих методів аналізу функціональних вимог до ІТ-продуктів.

### 1.3 Постановка задачі дослідження

Результати досліджень, описаних у підрозд. 1.1 та підрозд. 1.2, дозволяють зробити такі висновки.

По-перше, вимоги до ІТ-продукту – це перші описи цього ІТ-продукту, з яких формуються описи всіх інших компонентів цього продукту. При цьому будь-який ІТ-продукт представлений як система – поєднання взаємодіючих елементів, організованих для досягнення однієї або кількох поставлених цілей [1]. Таке представлення ІТ-продукту визначає два перші з основних процесів життєвого циклу системи, які слід виконувати: процес визначення потреб і

вимог зацікавленої сторони та процес визначення системних вимог [1]. Обидва ці процеси мають одну спільну рису: спочатку виконується робота зі збору індивідуальних вимог, а потім виконується робота з аналізу повного набору вимог. У той же час основні характеристики такого аналізу в [1] означають, що вимоги є необхідними, вільно реалізованими, послідовними, змістовними, повними, відображають специфіку, здійсненні, простежуються, перевіряються, прийнятні та мають межі.

По-друге, використання розглянутих характеристик аналізованих вимог значно ускладнює автоматизацію аналізу вимог. Справа в тому, що переважна більшість цих характеристик не мають формального опису. Отже, наявність цих характеристик в аналізованих вимогах неможливо підтвердити чи спростувати стандартними методами аналізу даних, структурованими чи неструктурованими текстами. Зокрема, ця складність пов'язана з такою характеристикою вимог до ІТ-продукту, як узгодженість. При цьому аналіз повного набору функціональних вимог до узгодженості дуже важливий при створенні таких ІТ-продуктів, як ІС.

По-третє, основним способом представлення вимог до ІТ-продуктів зараз є візуальні моделі. Ці моделі можуть бути структурними або об'єктно-орієнтованими. Але незалежно від виду конкретних візуальних моделей, їх дослідження найбільш розповсюдженими методами кількісного аналізу без адаптації цих методів до особливостей цих візуальних моделей слід визнати неможливим.

По-четверте, результати аналізу особливостей, переваг та недоліків розглянутих досліджень методів аналізу вимог дозволяють такі висновки [21]:

- а) більшість методів аналізу функціональних вимог незалежно від напрямку досліджень спираються на будь-які формалізовані описи;
- б) одним із основних підходів до формального опису вимог до ІС, стейкхолдерів як джерел цих вимог, а також поведінки учасників ІТ-проекту в ході роботи з вимогами є знання-орієнтовані моделі;

в) знання-орієнтовані описи вимог найчастіше втілюються у вигляді візуальних діаграм або математичних моделей;

г) використання знання-орієнтованих описів для аналізу вимог до ІС та їх окремих характеристик, як і раніше, спрямоване на вдосконалення відомих методів аналізу.

Особливу увагу в дослідженні моделей і методів аналізу вимог слід приділити вимогам, які опубліковані у вигляді математичних виразів. Слід відзначити, що таке представлення функціональних вимог до ІТ-продукту майже не досліджувалося. Між тим, використання математичних виразів для описів потреб користувачів ІТ-продуктів та функціональних вимог, які розроблено на основі цих потреб, є дуже корисним, оскільки дозволяє ще на перших стадіях ІТ-проекту точно сформулювати правила виконання функції у вигляді формули розрахунку окремих показників, які характеризують об'єкти предметної галузі.

Серед усіх можливих видів аналізу слід преш за все звернути увагу на виявлення дубльованих функціональних вимог до ІТ-продукту. Результати цього різновиду аналізу дуже сильно впливають на всі подальші роботи з планування ІТ-проекту створення ІТ-продукту, тому що дозволяють вилучити з подальшого розгляду ті функціональні вимоги і, відповідно, ті функції ІТ-продукту, які вже були описані раніше. Внаслідок такого вилучення стає можливим зекономити достатньо значну кількість ресурсів ІТ-проекту за рахунок відмови від реалізації повторюваних у ІТ-продукті функцій.

Тому метою кваліфікаційної роботи є дослідження методів виявлення дублювання функціональних вимог до ІТ –продукту, опублікованих у вигляді математичних виразів.

Для досягнення мети в кваліфікаційній роботі пропонується вирішити такі задачі дослідження:

– проаналізувати сучасний стан моделей і методів аналізу вимог до інформаційної системи;

- дослідити існуючі методи виявлення дублювання функціональних вимог до ІТ-продукту;
- дослідити можливість використання існуючих методів для виявлення дублювання функціональних вимог, опублікованих як математичні вирази;
- провести апробацію отриманих рішень на конкретному прикладі.

## **2 ДОСЛІДЖЕННЯ ІСНУЮЧИХ МЕТОДІВ ВИЯВЛЕННЯ ДУБЛЮВАННЯ ФУНКЦІОНАЛЬНИХ ВИМОГ ДО ІТ-ПРОДУКТУ**

### **2.1 Дослідження особливостей методу сценарного аналізу**

Сучасна модель вимог компанії «Microsoft» заснована на публікаціях вимог до ІТ-продукту у вигляді сценаріїв виконання окремих функцій цього ІТ-продукту. Але використання такого підходу, який отримав назву «сценарний підхід», до публікацій вимог ускладнює рішення задач аналізу вимог до ІТ-продукту. Використання сценарного підходу до опису та публікації вимог серйозно ускладнює визначення меж створюваного або модернізованого ІТ-продукту. Крім того, використання UML і, зокрема, діаграм варіантів використання (Use Case) для опису сценаріїв виконання вимог правовласників, вимагає розробки спеціальних моделей, методів та інформаційних технологій, що дозволяють формалізувати та автоматизувати роботи з аналізу вимог. Здійснення аналізу сценаріїв виконання вимог правовласників вручну неминуче породжує високий ризик невиявлення помилок. Ці помилки можуть бути викликані невірним визначенням сценаріїв виконання окремих вимог правовласників, а також синтезом опису архітектури ІТ-продукту без узгодження описів сценаріїв виконання окремих вимог до функцій цього продукту.

Базовий метод сценарного аналізу вимог правовласників до системи розглянуто у [22] як послідовність таких етапів.

Етап 1. Виділення для функціональної вимоги базового сценарію, який описує дії системи, що повторюються найчастіше незалежно від типу запиту актора та умов виконання цих запитів.

Етап 2. Опис зв'язків вимог між собою на основі діаграм Use Case.

Етап 3. Виявлення взаємовиключних вимог та декомпозиція окремих сценаріїв.

Етап 4. Формування підсумкової діаграми Use Case, яка описує сформульовані функціональні вимоги до системи та перетворення даної діаграми на нумерований список функціональних вимог до системи.

Приклад виділення базового сценарію, який описує дії системи, що повторюються найчастіше незалежно від типу запиту актора та умов виконання цих запитів, наведено на рис. 2.1 [22].

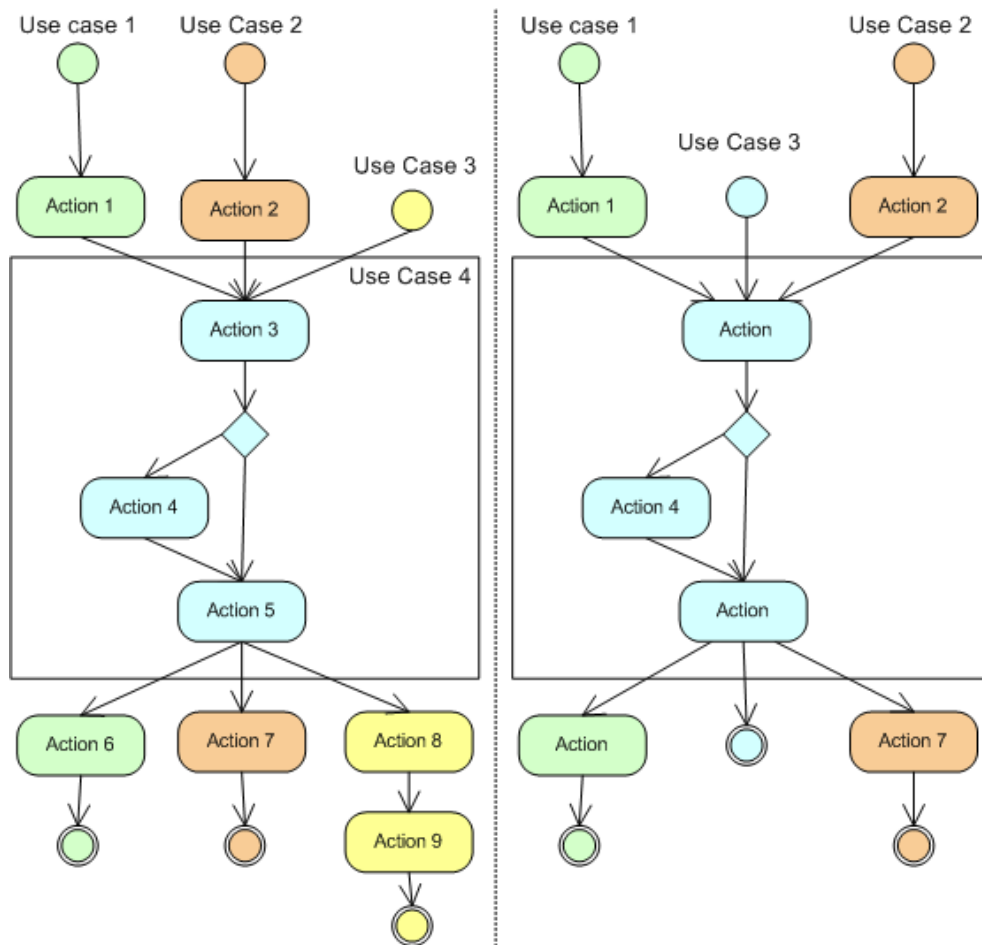


Рисунок 2.1 – Приклад виділення базового сценарію, який описує дії системи, що повторюються найчастіше

Приклад підсумкової діаграми Use Case, яка описує сформульовані функціональні вимоги до системи, наведено на рис. 2.2 [22].

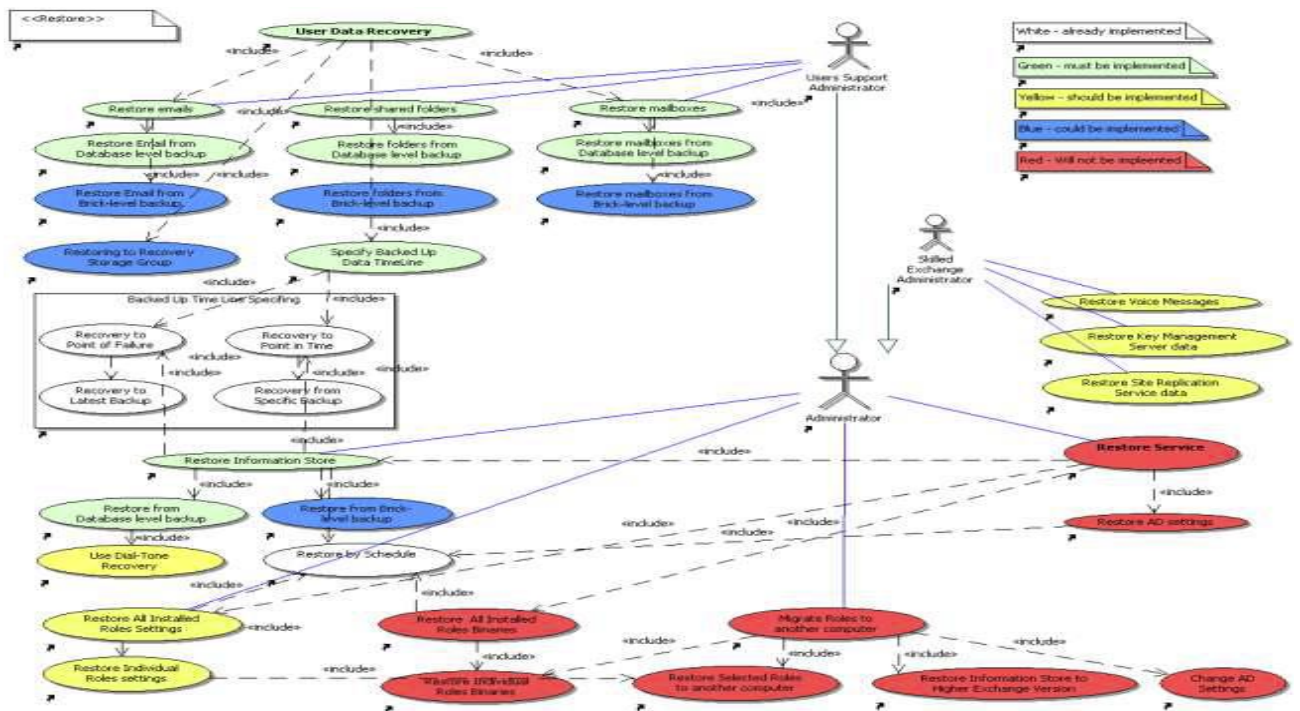


Рисунок 2.2 – Приклад підсумкової діаграми Use Case, яка описує сформульовані функціональні вимоги до системи

Виконання даного методу засноване на припущенні про можливість представлення вимоги правовласника у вигляді одного базового сценарію та множині додаткових сценаріїв, що розширюють можливості базового сценарію для різних ситуацій предметної області [22].

Виявлення дубльованих функціональних вимог відбувається у методі сценарного аналізу під час виконання Етапу 1. За результатами виконання цього етапу дві вимоги будуть вважатися дубльованими, якщо їх базові сценарії співпадають повністю або із заздалегідь визначеним ступенем.

Але до розглянутого базового методу сценарного аналізу слід висунути такі зауваження. Базовий метод сценарного аналізу в ІТ-проектах створення ІС в даний час виконується аналітиками переважно вручну. При цьому Етапи 1 та 2 виконуються аналітиками на основі загальних рекомендацій щодо збирання та публікації вимог правовласників. Етапи 3 та 4 виконуються аналітиками із застосуванням методів візуального аналізу діаграм UseCase, які

описують сформульовані функціональні вимоги до системи. Така організація робіт пов'язана з тим, що CASE-засоби візуального моделювання, а також інформаційні технології управління вимогами не мають спеціальних функцій аналізу діаграм Use Case. Крім того, обмеження розміру екранів дисплеїв призводить до неминучого виключення з зору аналітиків великої групи елементів підсумкової діаграми Use Case, особливо у випадках опису архітектури середніх і великих ІС [23].

## 2.2 Дослідження особливостей вдосконаленого методу синтезу варіантів опису архітектури створюваної інформаційної системи

Як метод пошуку дубльованих функціональних вимог пропонується дослідити вдосконалений метод синтезу варіантів опису архітектури створюваної ІС. Цей метод був описаний у [24, 25] та вдосконалений у [26]. Метод складається з наступних етапів і кроків.

Етап 1. Сформувати початковий варіант опису архітектури створеної ІС  $Arch_{base}$ .

Крок 1.1. Визначити кількість загальносистемних представлень функціональних вимог до ІС на рівні знань  $n = |K_i^{fIS}|$ .

Крок 1.2. Сформувати набір описів ІТ-послуг  $IT_{act}$  шляхом виконання операції  $IT_{act_j} = K_i^{fIS}$ .

Крок 1.3. Для набору  $\{K_i^{fIS}\}$  утворити матрицю  $Arch_{base}$  наступної форми:

$$Arch_{base} == \begin{bmatrix} t_{11}(IT_{act_1}) & \dots & t_{1n}(IT_{act_n}) \\ \dots & \dots & \dots \\ t_{n1}(IT_{act_n}) & \dots & t_{nn}(IT_{act_n}) \end{bmatrix}, \quad (2.1)$$

де  $t_{ij}(IT_{act_j})$  - кількість входжень загальносистемних представлень функціональних вимог до ІС на рівні знань в описі ІТ-сервісу (функції) створеної ІС,  $i, j=1, \dots, n$ . Завершити виконання кроку методу.

Значення  $t_{ij}(IT_{act_j})$  під час формування матриці  $Arch_{base}$  визначаються таким чином:

-  $t_{ij}(IT_{act_j}) = 1$  якщо для запропонованого варіанту включення  $K_i^{fIS}$  в кластер  $IT_{act_j}$  значення  $Profit(IT_{act_j}, r)$  не розраховано;

- якщо  $t_{ij}(IT_{act_j}) = 1$ , то  $t_{ji}(IT_{act_j}) = 0$ .

Етап 2. Встановити значення коефіцієнта відштовхування  $r$  і обчислити значення функції прибутку  $Profit(IT_{act}, r)$  за виразом:

$$Profit = \frac{\sum_{j=1}^k \frac{S(IT_{act_j}) |IT_{act_j}|}{W(IT_{act_j})^r}}{\sum_{j=1}^k |IT_{act_j}|} \quad (2.2)$$

де  $S(IT_{act_j})$  – кількість елементів в орієнтованому на знання описі ІТ-послуги  $IT_{act_j}$ ;

$W(IT_{act_j})$  – кількість унікальних елементів в знання-орієнтованому описі ІТ-послуги  $IT_{act_j}$ ;

$r$  – коефіцієнт відштовхування, значення якого залежить від типу архітектури ІС;

$|IT_{act_j}|$  – кількість загальносистемних представлень функціональних вимог до ІС на рівні знань в описі ІТ-послуги  $IT_{act_j}$ ;

$k$  – кількість ІТ-послуг (функцій) ІС.

Етап 3. Провести синтез оптимальних та/або прийнятних варіантів опису архітектури створюваної ІС.

Крок 3.1. Взяти  $Profit_{max} = Profit(IT_{act}, r)$ ,  $i = 1, j = i + 1$ .

Крок 3.2. Якщо  $t_{ij}(IT_{act_j}) = 1$ , то виключити  $K_i^{fIS}$  з  $IT_{act_i}$  та включити  $K_i^{fIS}$  в  $IT_{act_j}$ . Інакше, йдемо до Шагу 3.8.

Крок 3.3. Для отриманого варіанту опису архітектури обчислити значення  $Profit(IT_{act}, r)$  за виразом (2.2).

Крок 3.4. Якщо  $Profit(IT_{act}, r) > Profit_{max}$ , тоді взяти  $t_{jj}(IT_{act_j}) = t_{jj}(IT_{act_j}) + t_{ii}(IT_{act_i})$ ,  $t_{ij}(IT_{act_i}) = 0$  та  $t_{ji}(IT_{act_i}) = 0$  для  $j = 1, \dots, n$ ,  $K_j^{fIS} = K_j^{fIS} \cup K_i^{fIS}$  та йдемо до Кроку 3.1.

Крок 3.5. Якщо  $Profit(IT_{act}, r) \in [Profit_{max} - \varepsilon; Profit_{max}]$ , тоді взяти  $t_{ij}(IT_{act_i}) = 1$ . Тут  $i$  надалі рекомендується визначити значення  $\varepsilon = Profit_{max} \times 0,1$  [24, 25].

Крок 3.6. Якщо  $Profit(IT_{act}, r) < Profit_{max} - \varepsilon$ , тоді взяти  $t_{ij}(IT_{act_i}) = 0$ .

Крок 3.7. Взяти  $j = j + 1$ . If  $j \leq n$ , потім йдемо до Кроку 3.2.

Крок 3.8. Взяти  $i = i + 1, j = i + 1$ . If  $i < n$ , потім йдемо до Кроку 3.2. В іншому випадку завершити виконання кроку методу

Етап 4. Виключити з розгляду всі варіанти опису архітектури створеної ІС  $IT_{act}$ , які зафіксовано на Етапі 3, для яких не виконується умова  $Profit(IT_{act}, r) \in [Profit_{max} - \varepsilon; Profit_{max}]$ . Завершити застосування методу.

Крок 4.1. Сформуванати версію опису архітектури створеної ІС, включаючи ті  $IT_{act_i}$ , для яких  $t_{ii}(IT_{act_i}) = 1$ .

Крок 4.2. Для кожного  $t_{ij}(IT_{act_i}) = 1$  матриці  $Arch_{base}$  сформуванати варіант опису архітектури створеної ІС, взявши за моментом формування  $K_i^{fIS} = K_i^{fIS} \cup K_j^{fIS}$ ,  $IT_{act_j} = \emptyset$  та  $t_{jj}(IT_{act_i}) = 0$ . Закінчення застосування методу.

Результатом цього методу є множина описів архітектури створеної ІС. У цій множині кожна версія опису архітектури міститиме лише унікальні функціональні вимоги або функціональні вимоги, які дублюють одна одну до

мінімально допустимого ступеня. Таким чином, цей метод дозволяє виявити та усунути функціональні вимоги, які повністю або значною мірою дублюють одна одну.

Але до розглянутого вдосконаленого методу синтезу варіантів опису архітектури створюваної ІС слід висунути такі зауваження. Цей метод для великої кількості функцій ІС слід виконувати тільки автоматизованим способом, для чого слід розробити відповідне програмне забезпечення. Застосування цього методу аналітиком вручну ускладнюється великою кількістю обчислень, які слід виконувати на Етапі 2 та Етапі 3. Крім того, цей метод не є наглядним, а результати вилучення дубльованих функціональних вимог записуються не у вигляді підсумкової візуальної моделі, а аналітично.

### 2.3 Висновки до другого розділу

В другому розділі було проведено дослідження двох основних методів пошуку дубльованих функціональних вимог до ІТ-продукту. Першим було досліджено базовий метод сценарного аналізу, який був запропонований Алістером Коберном у [22]. Цей метод базується на моделі вимог, яка запропонована компанією «Microsoft».

Було розглянуто основні етапи застосування методу, які полягають у виділенні для функціональних вимог базових сценаріїв, які описують дубльовані дії системи, визначенню та описі зв'язків вимог між собою на основі діаграм Use Case, виявленню взаємовиключних вимог та декомпозиції окремих сценаріїв, а також у формуванні підсумкової діаграми Use Case, яка описує сформульовані функціональні вимоги до системи. Визначено головні переваги та недоліки використання базового методу сценарного аналізу для пошуку дубльованих функціональних вимог до ІТ-продукту.

Другим було досліджено вдосконалений метод синтезу варіантів опису архітектури створюваної ІС. Цей метод полягає у формуванні знання-орієнтованого опису архітектури створюваної ІС, визначенню значення функції корисності (2.2), та циклічного розрахування нових значень функції корисності (2.2) для варіантів переводу знання-орієнтованого опису кожної окремої функції з одного кластеру в інший. Слід зазначити, що значення функції корисності (2.2) буде збільшуватися тільки у випадку, якщо пара функцій, описи яких розміщуються в одному й тому ж кластері, дублюють одна іншу. За результатами застосування цього методу залишаються тільки ті варіанти опису архітектури створюваної ІС, для яких дублювання функцій або виключено зовсім, або зведено до прийняттого мінімуму. Визначено головні переваги та недоліки використання вдосконаленого методу синтезу варіантів опису архітектури створюваної ІС для пошуку дубльованих функціональних вимог до ІТ-продукту.

### **3 ДОСЛІДЖЕННЯ МОЖЛИВОСТІ ВИКОРИСТАННЯ ІСНУЮЧИХ МЕТОДІВ ДЛЯ ВИЯВЛЕННЯ ДУБЛЮВАННЯ ФУНКЦІОНАЛЬНИХ ВИМОГ, ОПУБЛІКОВАНИХ ЯК МАТЕМАТИЧНІ ВИРАЗИ**

3.1 Дослідження можливості використання вдосконаленого методу синтезу варіантів опису архітектури створюваної інформаційної системи

Проведення дослідження можливості використання вдосконаленого методу синтезу варіантів опису архітектури створюваної ІС для виявлення дублювання функціональних вимог, опублікованих як математичні вирази, пропонується здійснювати шляхом визначення можливості виконання кожного з етапів цього методу.

На першому етапі методу пропонувалося сформувати початковий варіант опису архітектури створеної ІС  $Arch_{base}$ . Для цього необхідно розробити описи загальносистемних представлень функціональних вимог до ІС на рівні знань  $K_i^{fIS}$ . Для цього давайте спочатку опублікуємо ці функціональні вимоги. Публікація кожної з цих вимог є математичним виразом, що визначає сценарій обчислення результату відповідних функцій.

Знання-орієнтований опис для таких функціональних вимог пропонується представити у вигляді множини. Елементами цієї множини будуть окремі елементи математичного виразу та знаки дії, причому кожен з елементів множини повинен мати самостійне значення.

Але такий підхід до формування знання-орієнтованого опису функціональної вимоги на основі її публікації у вигляді математичного виразу породжує проблему, яку пропонується назвати «проблема надмірної деталізації». Суть цієї проблеми полягає в тому, що кожен конкретний математичний вираз містить унікальні елементи, які насправді є окремими випадками одного і того ж елемента більш загального характеру.

Для подолання проблеми надмірної деталізації пропонується в процесі формування знання-орієнтованого опису функціональної вимоги  $K_i^{fIS}$  на основі її публікації у вигляді математичного виразу замінити окремі випадки окремих елементів узагальненими описами цих елементами.

Такий підхід дозволяє використовувати вдосконалений метод синтезу варіантів опису архітектури створюваної ІС для виявлення дублювання функціональних вимог, опублікованих як математичні вирази, без суттєвої зміни етапів і кроків цього методу.

### 3.2 Дослідження можливості використання базового методу сценарного аналізу

Проведення дослідження можливості використання базового методу сценарного аналізу для виявлення дублювання функціональних вимог, опублікованих як математичні вирази, пропонується здійснювати шляхом визначення можливості виконання кожного з етапів цього методу.

Для виконання першого етапу базового методу сценарного аналізу слід також опублікувати функціональні вимоги у вигляді математичних виразів. Як і для вдосконаленого методу синтезу варіантів опису архітектури створюваної ІС, публікація кожної з цих вимог є математичним виразом, що визначає сценарій обчислення результату відповідних функцій. Ці вирази фіксуються у публікаціях як назви прецедентів діаграм Use Case, які описують функції ІТ-продукту.

На третьому етапі можлива декомпозиція окремих математичних виразів за умови можливості їх розділення на окремі елементи. Але така декомпозиція призведе до втрати семантичної цілісності похідного математичного виразу і буде ускладнювати переклад результатів застосування

базового методу сценарного аналізу на мову, яку зрозуміють і розробники, і замовники.

Слід зазначити, що основним засобом виявлення дубльованих функціональних вимог у базовому методі сценарного аналізу є візуальний аналіз діаграм Use Case, який проводиться аналітиком. Тому в результаті публікацій функціональних вимог у вигляді математичних виразів додаткових переваг не виявляється, а складність застосування методу збільшується внаслідок необхідності розуміння аналітиком семантики конкретних математичних виразів, якими описано сценарії окремих прецедентів.

З метою усунення визначених у підрозд. 2.1 недоліків базового методу сценарного аналізу у [23] було розроблено удосконалений метод сценарного аналізу. Цей метод можна представити як сукупність таких етапів [22].

Етап 1. Виділення для вимоги правовласника базового сценарію, який описує дії ІС, що повторюються найчастіше незалежно від типу запиту актора та умов виконання цих запитів.

Етап 2. Формування множини публікацій сценаріїв виконання вимог правовласників на основі діаграм Use Case.

Етап 3. Формування множини уявлень сценаріїв виконання вимог правовласників на рівні знань.

Етап 4. Виявлення суперечливих вимог шляхом виконання методу аналізу окремих фреймів уявлень вимог на несуперечність.

Етап 5. Автоматичне формування варіантів підсумкової діаграми Use Case, яка визначає архітектуру ІС як сукупність сформульованих функціональних вимог до ІС.

Етап 6. Вибір раціонального опису архітектури ІС за результатами порівняльного аналізу сформованих на Етапі 5 варіантів підсумкової діаграми Use Case.

Етап 7. Перетворення обраного раціонального опису архітектури ІС на нумерований список функціональних вимог до системи.

Методи формування представлень сценаріїв виконання вимог правовласників на рівні знань у загальному випадку аналогічні методам, розглянутим у [24-26].

Але для виконання Етапу 5 удосконаленого методу сценарного аналізу авторами [22] запропоновано використовувати розглянутий вище вдосконалений метод синтезу варіантів опису архітектури створюваної ІС. Тому використання удосконаленого методу сценарного аналізу не дає значних переваг перед вдосконаленим методом синтезу варіантів опису архітектури створюваної ІС.

### 3.3 Порівняльний аналіз переваг та недоліків застосування досліджених методів

Проведемо порівняльний аналіз переваг та недоліків застосування базового методу сценарного аналізу та вдосконаленого методу синтезу варіантів опису архітектури створюваної ІС. Для цього визначимо такі критерії порівняння.

Критерій «Складність використання аналітиком» використовується для порівняльного оцінювання методів за витратами часу, який буде потрібен людині-аналітику на засвоєння методів та правил їх застосування.

Критерій «Можливість автоматизованої реалізації методу» використовується для порівняльного оцінювання методів за оцінкою можливості автоматизації виконання кожного з етапів досліджуваних методів.

Критерій «Витрати обчислювальних ресурсів» використовується для порівняльного оцінювання методів за оцінками можливих витрат обчислювальних ресурсів під час автоматизованої реалізації досліджуваних методів.

Критерій «Вплив на застосування обмежень використовуваного комплексу технічних засобів» використовується для порівняльного оцінювання методів за можливістю негативного впливу на точність результатів виконання етапів досліджуваних методів будь-яких обмежень використовуваного комплексу технічних засобів (автоматизованого робочого місця аналітика).

Критерій «Обмеження публікацій функціональних вимог» використовується для порівняльного оцінювання методів за можливістю публікувати функціональні вимоги з використанням різних візуальних моделей.

Критерій «Точність результатів застосування» використовується для порівняльного оцінювання методів за оцінками можливої кількості помилок, які можуть виникнути наприкінці застосування досліджуваних методів.

Результати порівняльного аналізу базового методу сценарного аналізу та вдосконаленого методу синтезу варіантів опису архітектури створюваної ІС наведено у табл. 3.1.

Таблиця 3.1 – Результати порівняння досліджуваних методів

Назва порівняння	критерію	Базовий метод сценарного аналізу	Вдосконалений метод синтезу варіантів опису архітектури створюваної ІС
Складність використання аналітиком		Дуже простий для використання (+)	Складний для використання (-)

Кінець таблиці 3.1

Назва критерію порівняння	Базовий метод сценарного аналізу	Вдосконалений метод синтезу варіантів опису архітектури створюваної ІС
Можливість автоматизованої реалізації методу	Неможливо (-)	Можливо (+)
Витрати обчислювальних ресурсів	Мінімальні (+)	Середні (-)
Вплив на застосування обмежень використовуваного комплексу технічних засобів	Негативно впливає розмір дисплею (-)	Не впливає (+)
Обмеження публікацій функціональних вимог	Тільки у вигляді діаграм Use Case (-)	У вигляді будь-яких структурних чи об'єктно-орієнтованих візуальних моделей (+)
Точність результатів застосування	Залежить від результатів візуального аналізу аналітика (-)	Є наслідком розрахунків за формулами (+)
Кількість позитивних оцінок	2	4
Кількість негативних оцінок	4	2

За результатами порівняльного аналізу можна зробити висновок, що вдосконалений метод синтезу варіантів опису архітектури створюваної ІС

більше підходить до виявлення дубльованих функціональних вимог, опублікованих у вигляді математичних виразів, ніж базовий метод сценарного аналізу.

### 3.4 Висновки до третього розділу

Було проведено дослідження можливості використання вдосконаленого методу синтезу варіантів опису архітектури створюваної ІС для виявлення дублювання функціональних вимог до ІТ-продукту, які опубліковано у вигляді математичних виразів. Визначено, що даний метод можливо адаптувати для аналізу подібних публікацій функціональних вимог.

Було проведено дослідження можливості використання базового методу сценарного аналізу для виявлення дублювання функціональних вимог до ІТ-продукту, які опубліковано у вигляді математичних виразів. Визначено, що даний метод також можливо адаптувати для аналізу подібних публікацій функціональних вимог.

Було проведено порівняльний аналіз переваг та недоліків застосування базового методу сценарного аналізу та вдосконаленого методу синтезу варіантів опису архітектури створюваної ІС. Для цього були визначені критерії порівняльного аналізу та проведено оцінювання переваг та недоліків застосування кожного з досліджуваних методів за цими критеріями. Результати порівняльного аналізу свідчать, що вдосконалений метод синтезу варіантів опису архітектури створюваної ІС більше підходить до виявлення дубльованих функціональних вимог, опублікованих у вигляді математичних виразів, ніж базовий метод сценарного аналізу.

## 4 ПРАКТИЧНА АПРОБАЦІЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

4.1 Опис потреб користувачів та функціональних вимог, які підлягають аналізу

Для опису функціональних вимог у вигляді математичної моделі використано модель задачі «Формування та ведення індивідуального плану науково-педагогічного працівника кафедри». Ця модель була розглянута в [27] і має такий вигляд

$$\alpha_e C_e = \sum_{a=1}^k \sum_{b=1}^l \sum_{c=1}^m t_{abc} + \sum_{d=1}^{p_1} t_d + \sum_{f=1}^{p_2} t_f + \sum_{g=1}^r \frac{a_g t_g}{q_g} + \sum_{i=1}^s t_i \quad (4.1)$$

де  $\alpha_e$  – частина ставки вчителя;

$C_e$  – кількість годин, відведених на одну ставку вчителя на планований навчальний рік;

$t_{abc}$  – кількість годин навчальної роботи, яка запланована на групи студентів і на навчальну дисципліну;

$t_d$  – кількість годин методичної роботи, яка планується для успішного завершення навчальної роботи;

$t_f$  – кількість годин методичної роботи, яка планується підвищити якість навчальної роботи;

$t_g$  – кількість годин видів наукової роботи, що планується на навчальний рік;

$a_g$  – запланована кількість результатів виду наукової роботи;

$q_g$  – запланована кількість співавторів за результатом виду наукової роботи;

$t_i$  – кількість годин організаційно-виховної роботи, яка планується на навчальний рік.

Рівняння (4.1) має виконуватися за умови [27]:

$$\left\{ \begin{array}{l} 0 < \alpha_e \leq 1; \\ t_{abc}, t_d, t_f, t_g, t_i, a_g > 0; \\ q_g \geq 1; \\ t_{abc}, t_d, t_f, t_g, t_i, a_g, q_g \in Z. \end{array} \right. \quad (4.2)$$

В результаті збирання вимог до функціональної задачі «Формування та ведення індивідуального плану науково-педагогічного працівника кафедри» в [12] було виділено такі функції:

- формування розділу документа «Позиція та ставка»;
- кількісне визначення годин навчальної роботи, запланованих на навчальний рік;
- формування розділу документа «Виховна робота»;
- кількісне визначення годин методичної роботи, яка планується для успішного завершення навчальної роботи;
- кількісне визначення годин методичної роботи, яка планується підвищити якість навчальної роботи;
- формування розділу документа «виховно-методична робота»;
- кількісне визначення годин наукової роботи, запланованих на навчальний рік;
- формування розділу документа «наукова робота»;
- кількісне визначення годин організаційно-виховної роботи, запланованих на навчальний рік;
- формування розділу документа «Організаційно-виховна робота»;
- формування розділу документа «Підсумковий розподіл часу за видами робіт на навчальний рік».

Деякі з цих публікацій описують функції, що формують окремі розділи документа «Індивідуальний план науково-педагогічного працівника кафедри». Однак інша частина цих публікацій описує функції для розрахунку кількості годин для різних видів роботи співробітника. Важко оцінити ступінь

дублювання таких функцій звичайним способом, оскільки їх опис є елементами моделі (4.1). Тому слід використати для виявлення дублювання вдосконалений метод синтезу варіантів опису архітектури створеної ІС.

Виділимо спочатку ті публікації, які описують обчислювальні функції задачі «Формування та ведення індивідуального плану науково-педагогічного працівника кафедри». Ці публікації включають [28]:

- кількість годин навчальної роботи, запланованих на навчальний рік;
- кількісна кількість годин методичної роботи, яка запланована на успішне завершення навчальної роботи;
- кількість годин методичної роботи, яка планується підвищити якість навчальної роботи;
- кількість годин наукової роботи, запланованих на навчальний рік;
- кількісна оцінка годин організаційно-виховної роботи, запланованих на навчальний рік.

Публікація кожної з цих вимог є математичним виразом, що визначає сценарій обчислення результату відповідних функцій (див. табл. 4.1).

Таблиця 4.1 – Публікації функціональних вимог

Позначення вимоги	Публікація вимоги
$f_{IS_1}$	$\sum_{a=1}^k \sum_{b=1}^l \sum_{c=1}^m t_{abc}$
$f_{IS_2}$	$\sum_{d=1}^{p_1} t_d$
$f_{IS_3}$	$\sum_{f=1}^{p_2} t_f$
$f_{IS_4}$	$\sum_{g=1}^r \frac{a_g t_g}{q_g}$
$f_{IS_5}$	$\sum_{i=1}^s t_i$

## 4.2 Результати застосування вдосконаленого методу синтезу варіантів опису архітектури створеної інформаційної системи

Тепер давайте виділимо заснований на знаннях  $K_i^{fIS}$  для кожної з функціональних вимог, публікації яких наведено в табл. 4.1. Як зазначено у підрозд. 3.1, знання-орієнтований опис для таких функціональних вимог пропонується представити у вигляді множини. Елементами такої множини будуть окремі елементи математичного виразу та знаки дії, причому кожен з елементів множини повинен мати самостійне значення. Використання цієї рекомендації дозволяє представити знання-орієнтовані описи публікацій функціональних вимог із табл. 4.1 наступним чином (див. табл. 4.2).

Наведені в табл. 4.2 представлення знання-орієнтованих описів розглянутих вимог можуть бути використані в ході застосування вдосконаленого методу синтезу варіантів опису архітектури створеної ІС.

Таблиця 4.2 – Знання-орієнтовані описи функціональних вимог

Позначення опису	Зміст опису
$K_1^{fIS}$	$(a, =, 1, \Sigma, k, b, =, 1, \Sigma, l, c, =, 1, \Sigma, m, t_{abc})$
$K_2^{fIS}$	$(d, =, 1, \Sigma, p_1, t_d)$
$K_3^{fIS}$	$(f, =, 1, \Sigma, p_2, t_f)$
$K_4^{fIS}$	$(g, =, 1, \Sigma, r, a_g, t_g, /, q_g)$
$K_5^{fIS}$	$(i, =, 1, \Sigma, s, t_i)$

Розглянемо основні результати реалізації етапів удосконаленого методу синтезу варіантів опису архітектури створеної ІС. Результатом першого етапу буде сформована матриця (2.1), яка в нашому випадку матиме вигляд

$$Arch_{base} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (4.3)$$

Результатом другого етапу буде значення функції виплат  $Profit(IT_{act}, r)$  для варіанту опису архітектури задачі як набору виділених функціональних вимог. Вихідні дані та результати розрахунків другого етапу наведені в табл. 4.3.

На третьому етапі визначаються можливі ситуації дублювання орієнтованих на знання описів обраних функціональних вимог. У нашому випадку результатом третього етапу буде виправлена матриця (4.3), яка має такий вигляд

Таблиця 4.3 – Вихідні дані та результати розрахунку другого етапу вдосконаленого методу

Позначення опису	Значення $S(IT_{act,j})$	Значення $W(IT_{act,j})$
$K_1^{fIS}$	16	10
$K_2^{fIS}$	6	6
$K_3^{fIS}$	6	6
$K_4^{fIS}$	9	9
$K_5^{fIS}$	6	6

Розрахунок вартості  $Profit(IT_{act}, r)$

$$\left| IT_{act,j} \right| = 5 \quad r = 2,5 \quad Profit(IT_{act}, r) = 0,058352$$

(сервіс – орієнтована архітектура)

$$Arch_{base} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (4.4)$$

Форма матриці (4.4) дозволяє стверджувати, що серед виявлених прецедентів немає ситуацій дублювання. Отже, результатом виконання етапу 4 вдосконаленого методу буде лише одна версія опису архітектури. Цей параметр міститиме описи кожного з виділених випадків використання.

Проведені розрахунки були зроблені без врахування проблеми надмірної деталізації публікацій функціональних вимог у вигляді математичних виразів. Для подолання цієї проблеми пропонується в процесі формування знання-орієнтованих описів  $K_i^{fIS}$  з таблиці 2 замінити окремі випадки окремих елементів загальними елементами, наведеними в табл. 4.4.

Таблиця 4.4 – Описи загальних елементів публікацій функціональних вимог

Приватні елементи	Узагальнений елемент	Семантика узагальнених елементів
$(c, d, f, g, i)$	$index_1$	Елемент, що позначає індекс першої операції
$(m, p_1, p_2, r, s)$	$top_1$	Елемент, що позначає верхню межу робочого діапазону
$(t_d, t_f, t_g, t_i)$	$t_{index_1}$	Елемент, що позначає змінну, що використовується в операції

З урахуванням занесених в табл. 4.4 описів узагальнених елементів, знання-орієнтовані описи функціональних вимог до обчислювальних функцій аналізованої задачі будуть мати такий вигляд (див. табл. 4.5).

Таблиця 4.5 – Скориговані описи функціональних вимог на основі знань

Позначення опису	Зміст опису
$K_1^{fIS}$	$(a, =, 1, \Sigma, k, b, =, 1, \Sigma, l, index_1, =, 1, \Sigma, top_1, t_{abc})$
$K_2^{fIS}$	$(index_1, =, 1, \Sigma, top_1, t_{index_1})$
$K_3^{fIS}$	$(index_1, =, 1, \Sigma, top_1, t_{index_1})$
$K_4^{fIS}$	$(index_1, =, 1, \Sigma, top_1, a_g, t_{index_1}, /, q_g)$
$K_5^{fIS}$	$(index_1, =, 1, \Sigma, top_1, t_{index_1})$

Результати реалізації першого та другого етапів удосконаленого методу для вихідних даних, наведених у таблиці 4.5, збігатимуться з результатами, розглянутими вище для вихідних даних, наведених у таблиці 4.2, і наведені у вигляді матриці (4.3) і табл. 4.3.

Результатом третього етапу вдосконаленого методу для вихідних даних, наведених у табл. 4.5, буде скоригована матриця (5). У цьому випадку ця матриця буде мати такий вигляд:

$$Arch_{base} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 3 \end{bmatrix}. \quad (4.5)$$

Форма матриці (4.5) дозволяє стверджувати, що серед обраних прецедентів є ситуації дублювання. Тому в результаті застосування Етапу 4

вдосконаленого методу сформувалися два варіанти опису архітектури. Перший варіант міститиме описи наступних трьох функціональних вимог:

- а) кількість годин навчальної роботи, запланованих на навчальний рік;
- б) кількість годин наукової роботи, запланованих на навчальний рік;
- в) кількість годин методичної роботи, яка запланована на успішне завершення навчальної роботи; кількість годин методичної роботи, яка планується підвищити якість навчальної роботи; кількісна оцінка годин організаційно-виховної роботи, запланованих на навчальний рік.

Остання із сформованих функціональних вимог першого варіанту опису архітектури включає другу, третю і п'яту з раніше визначених публікацій обчислювальних функцій.

Другий варіант міститиме описи наступних двох функціональних вимог:

- а) кількість годин навчальної роботи, запланованих на навчальний рік; кількість годин наукової роботи, запланованих на навчальний рік;
- б) кількісна оцінка годин методичної роботи, яка запланована на успішне завершення навчальної роботи; кількість годин методичної роботи, яка планується підвищити якість навчальної роботи; кількісна оцінка годин організаційно-виховної роботи, запланованих на навчальний рік.

Перша з вимог другого варіанту опису архітектури включає першу і четверту із раніше визначених публікацій обчислювальних функцій.

#### 4.3 Перевірка отриманих результатів

Щоб перевірити отриманий результат, проведемо пошук дубльованих функціональних вимог за їх публікаціями у вигляді тексту, написаного природною мовою. В якості таких публікацій використаємо назви виділених функцій. Тоді описи цих вимог, засновані на знаннях, будуть сформовані шляхом обробки публікацій стемером Портера, а потім вилученням стоп-слів.

Результати формування знання-орієнтованих описів функціональних вимог наведено в табл. 4.6.

Для цих описів на основі знань результат першого етапу вдосконаленого методу також матиме вигляд матриці (4.3). Результат другого етапу вдосконаленого методу наведено в табл. 4.7.

В результаті третього етапу застосування вдосконаленого методу була отримана скоригована матриця (4.3), яка має такий вигляд:

Таблиця 4.6 – Знання-орієнтовані описи функціональних вимог

Позначення опису	Зміст опису
$K_1^{fIS}$	Кількість годин навчального плану роботи на навчальний рік
$K_2^{fIS}$	Кількість годин методолог план роботи Успіх завершити навчальну роботу
$K_3^{fIS}$	План роботи методиста кількісних годин покращення якості навчальної роботи
$K_4^{fIS}$	Кількість годин план наукової роботи на навчальний рік
$K_5^{fIS}$	Кількість годин організації навчального плану роботи на навчальний рік

Таблиця 4.7 – Вихідні дані та результати розрахунку Етапу 2 вдосконаленого методу

Позначення опису	$S(IT_{act_j})$ значення	$W(IT_{act_j})$ значення
$K_1^{fIS}$	7	7
$K_2^{fIS}$	9	8
$K_3^{fIS}$	9	8
$K_4^{fIS}$	7	7
$K_5^{fIS}$	8	8

Розрахунок вартості  $Profit(IT_{act}, r)$

$ IT_{act_j}  = 5$	$r = 2,5$	$Profit(IT_{act}, r)$
	(сервіс – орієнтована архітектура)	$= 0,05032$

$$Arch_{base} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 \end{bmatrix}. \quad (4.6)$$

Форма матриці (4.6) дозволяє стверджувати, що серед обраних вимог є ситуації дублювання. Тому в результаті впровадження Етапу 4 вдосконаленого методу сформувалися два варіанти опису архітектури. Перший варіант міститиме описи наступних трьох функціональних вимог:

- а) кількість годин методичної роботи, яка планується для успішного завершення навчальної роботи;
- б) кількісне визначення годин методичної роботи, яка планується підвищити якість навчальної роботи;

в) кількість годин навчальної роботи, запланованих на навчальний рік; кількість годин наукової роботи, запланованих на навчальний рік; кількісна оцінка годин організаційно-виховної роботи, запланованих на навчальний рік.

Остання із сформованих функціональних вимог першого варіанту опису архітектури включає першу, четверту і п'яту з раніше ідентифікованих публікацій обчислювальних функцій.

Другий варіант міститиме описи наступних двох функціональних вимог:

а) кількість годин методичної роботи, яка планується для успішного завершення навчальної роботи; кількість годин методичної роботи, яка планується підвищити якість навчальної роботи;

б) кількість годин навчальної роботи, запланованих на навчальний рік; кількість годин наукової роботи, запланованих на навчальний рік; кількісна оцінка годин організаційно-виховної роботи, запланованих на навчальний рік.

Перша з функціональних вимог другої версії опису архітектури включає другу і третю з виділених раніше публікацій обчислювальних функцій.

Таким чином, підтверджуються факти наявності дубльованих функціональних вимог до обчислювальних функцій аналізованої задачі.

Цей висновок дозволяє підтвердити можливість виявлення повторюваних функціональних вимог на основі їх публікації у вигляді математичних виразів за умови усунення зайвої деталізації цих виразів.

#### 4.4 Висновки до четвертого розділу

Під час проведення ініціації IT-проекту було зроблено збирання та публікація функціональних вимог до функціональної задачі «Формування та ведення індивідуального плану науково-педагогічного працівника кафедри». Для цієї задачі було отримано математичну модель розрахунків індивідуального плану працівника кафедри та визначені основні функції цієї

задачі. З визначених функцій було для подальшого аналізу обрано ті функції, які могли бути описані математичними виразами.

Проведено застосування вдосконаленого методу синтезу варіантів опису архітектури створеної ІС для пошуку дубльованих функціональних вимог, опублікованих у вигляді математичних виразів. Наведено, що з врахуванням рішення проблеми надмірної деталізації стає можливим знайти дубльовані функціональні вимоги.

Наявність дубльованих функціональних вимог серед їх публікацій було підтверджено також застосуванням вдосконаленого методу синтезу варіантів опису архітектури створеної ІС до публікацій тих же самих вимог, але зроблених у вигляді текстів природною мовою.

## ВИСНОВКИ

В ході виконання кваліфікаційної роботи було проведене дослідження методів виявлення дублювання функціональних вимог до ІТ –продукту, опублікованих у вигляді математичних виразів. Отримані результати дозволяють свідчити про можливість пошуку подібних дубльованих вимог існуючими методами аналізу функціональних вимог до ІТ-продукту.

Було проведено аналіз сучасних моделей вимог до ІТ-продуктів. Наведено, що основним способом опису функціональних вимог до ІТ-продуктів є візуальні моделі. Проведено аналіз сучасних досліджень в галузі аналізу функціональних вимог. За результатами цього дослідження встановлено основні недоліки існуючих методів аналізу вимог до ІТ-продуктів. За результатами досліджень визначено мету та основні задачі кваліфікаційної роботи.

Проведено дослідження можливості застосування базового методу сценарного аналізу та вдосконаленого методу синтезу варіантів опису архітектури створеної ІС для пошуку дубльованих функціональних вимог. Розглянуто основні етапи зазначених методів, визначено їх переваги та недоліки.

Проведено дослідження можливості застосування базового методу сценарного аналізу та вдосконаленого методу синтезу варіантів опису архітектури створеної ІС для пошуку дубльованих функціональних вимог, опублікованих у вигляді математичних виразів. Визначені особливості використання зазначених методів для такого аналізу. Проведено порівняльний аналіз, за результатами якого обгрунтовано використання для пошуку дубльованих вимог вдосконаленого методу синтезу варіантів опису архітектури створеної ІС.

Проведено практичну апробацію отриманих результатів дослідження. Як тестовий приклад було використано публікації функціональних вимог

функціональної задачі «Формування та ведення індивідуального плану науково-педагогічного працівника кафедри». Наведено результати застосування вдосконаленого методу синтезу варіантів опису архітектури створеної ІС для пошуку дубльованих функціональних вимог без врахування дії проблеми надмірної деталізації та з врахуванням дії цієї проблеми. Отримані результати пошуку було підтверджено за результатами застосування вдосконаленого методу синтезу варіантів опису архітектури створеної ІС до аналізу публікацій функціональних вимог цієї ж функціональної задачі, зроблених у вигляді текстів природною мовою.

Отримані результати кваліфікаційної роботи рекомендується використовувати під час виконання робіт із аналізу функціональних вимог до ІТ-продуктів на стадіях ініціації та планування ІТ-проектів.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. GOST R 57193–2016. System and program engineering. System life cycle processes (ISO/IEC/IEEE 15288:2015, NEQ). (2016). Introduced 31–10–2016. М.: Standartinform, 98.
2. Методичні вказівки щодо розробки та оформлення кваліфікаційної роботи (для студентів усіх форм навчання другого (магістерського) рівня вищої освіти спеціальності 122 Комп'ютерні науки освітньо-професійної програми «Управління проектами в галузі інформаційних технологій») / 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. – 30 с.
3. ДСТУ 3008:2015. Державний стандарт України. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлення. – К.: ДП «УкрНДНЦ», 2016. – 31 с.
4. ДСТУ 8302:2015. Бібліографічне посилання. Загальні положення та правила складання. / Видання офіційне. – К.: ДП «УкрНДНЦ», 2016 – 20 с.
5. Моделирование требований пользователей. Сайт «Microsoft Developer Network». URL: <https://msdn.microsoft.com/ru-ru/library/dd409376.aspx>.
6. Петров Э.Г., Чайников С.И., Овезгельдыев А.О. Методология структурного системного анализа и проектирования крупномасштабных ИУС. Концепции и методы. Харьков: Рубикон, 1997. Часть 1. 140 с.
7. SSADM Version 4 Reference Manual. Oxford: NCC Blackwell, 1990. 1400 p.
8. M. Maguire, N. Bevan (2002). User requirements analysis. A review of supporting methods. Proceedings of IFIP 17th World Computer Congress, Montreal, Canada, 25-30 August 2002, 133-148. doi: 10.1007/978-0-387-35610-5\_9

9. H.F. Martins, A.C. de Oliveira, E.D. Canedo etc. (2019). Design thinking: Challenges for software requirements elicitation. *Information*, 10, 12, 371. doi: <https://doi.org/10.3390/info10120371>
10. Stowell F., Cooray S. (2017). The appreciative system, learning, and its impact on information systems design. *Communications of the Association for Information Systems*, 40, 93-119.
11. Ali N., Lai R. (2017). A method of requirements elicitation and analysis for Global Software Development. *Journal of Software: Evolution and Process*, 29, 4, 18-30. doi: 10.1002/smr.1830
12. Beimel D., Kedmi S. (2019). Improving the identification of functional system requirements when novice analysts create use case diagrams: the benefits of applying conceptual mental models. *Requirements Engineering*, 24, 4, 483-502. doi: <https://doi.org/10.1007/s00766-018-0296-z>
13. Asteasuain F., Braberman V. (2017). Declaratively building behavior by means of scenario clauses. *Requirements Engineering*, 22, 2, 239-274. doi: 10.1007/s00766-015-0242-2
14. Liu L., Zhou Q., Liu J., Cao Z. (2017). Requirements cybernetics: Elicitation based on user behavioral data. *Journal of Systems and Software*, 124, 187-194. doi: 10.1016/j.jss.2015.12.030
15. Yu Y.-J., Liu C. (2017). Little Model in Big Data: An Algebraic Approach to Analysing Abstract Software Behaviours. *Ruan Jian Xue Bao/Journal of Software*, 28 (6), 1488-1497. doi: 10.13328/j.cnki.jos.005229
16. T. Ambreen, N. Ikram, M. Usman, M. Niazi (2018). Empirical research in requirements engineering: trends and opportunities. *Requirements engineering*, 23, 1, 63-95. doi: 10.1007/s00766-016-0258-2
17. Kaiya, K. Adachi, Y. Chubachi (2019) Requirements Exploration by Comparing and Combining Models of Different Information Systems. In: M. Virvou, F. Kumeno, K. Oikonomou (eds) *Knowledge-Based Software Engineering: 2018. JCKBSE 2018. Smart Innovation, Systems and Technologies*, vol 108. Pp. 64-74. doi: 10.1007/978-3-319-97679-2\_7.

18. S.-K. Kim, T. Myers, M.-F. Wendland, P.A. Lindsay (2012) Execution of natural language requirements using state machines synthesised from Behavior Trees. *Journal of Systems and Software*. 85, 11, 2652-2664. doi: 10.1016/j.jss.2012.06.013.
19. V. Cornejo-Aparisio, S. Flores-Silva, N. Bedregal-Alpaca, D. Tupacyupanqui-Jaen. Procedural model for the specification of functional requirements in software construction [Modelo procedimental para la especificación de requisitos funcionales en la construcción de software]. *RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao*, 2020, E26, 571-586.
20. C.-H. Lee, C.-H. Chen, A.J.C. Trappey (2019). A structural service innovation approach for designing smart product service systems: Case study of smart beauty service. *Advanced Engineering Informatics*, 40, 154-167. doi: <https://doi.org/10.1016/j.eswa.2019.113062>.
21. Vasilcova, N., Panforova, I., & Neumyvakina, O. (2020). Improving a method to analyze the requirements for an information system for consistency. *Eastern-European Journal of Enterprise Technologies*, 3(2 (105), 17–27. <https://doi.org/10.15587/1729-4061.2020.205518>
22. Kobern, A. (2002). *Writing Effective Use Cases*. M.: «Lory», 2002, 288.
23. Mohammed, M. Q., Muhamed, S. Q., Ievlanov, M., & Gazetdinova, Z. (2019). Improvement of the method of scenario analysis of functional requirements to an information system. *Eastern-European Journal of Enterprise Technologies*, 3(2 (99), 25–35. <https://doi.org/10.15587/1729-4061.2019.170351>
24. M.V. Yevlanov, N.V. Vasilcova, I.Yu. Panforova. “Models and methods of synthesis of the description of rational architecture of information system”. *Bulletin of the Scientific University "Lviv Polytechnic". Information Systems and Networks Series* (2015). No. 829: 135–152 .
25. M.V. Evlanov (2016). Development of the model and method of selecting the description of rational architecture of information system. *Eastern-*

European Journal of Enterprise Technologies, 1(2(79), 4–12.  
<https://doi.org/10.15587/1729-4061.2016.60583>.

26. M.V. Yevlanov. “Improved method of synthesis of variants of description of architecture of the created information system”. ACS and automation devices (2018). No 175: 32–41.

27. Vasytsova N., Panforova I., Kuzma Ye. Formation of Function Use Cases Based on Its Mathematical Model // Fourth International Scientific and Technical Conference “COMPUTER AND INFORMATION SYSTEMS AND TECHNOLOGIES”. Kharkiv: DISA PLUS LLC, 2020: 59-60.