

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 оцінювання змін часу, необхідного для виконання
задач, за результатами реалізації довгострокового ІТ-проєкту
(тема)

Виконала:

Студентка 2 курсу, групи УПІТМ-22-3

Попова Анастасія Вікторівна

(прізвище, ім'я, по батькові)

Спеціальність 122 Комп'ютерні науки

(код і повна назва спеціальності)

Тип програми освітньо-наукова

Освітня програма Управління проєктами в
галузі інформаційних технологій

(повна назва освітньої програми)

Керівник професор кафедри ІУС

Наталія ВАСИЛЬЦОВА

(посада, власне ім'я, прізвище)

Допускається до захисту

Зав. кафедри



(підпис)

Костянтин ПЕТРОВ

(власне ім'я, прізвище)

2024 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 проектами в галузі інформаційних технологій
(повна назва)

ЗАТВЕРДЖУЮ: 
Зав. кафедри _____
(підпис)
« 01 » квітня 20 24 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентці Поповій Анастасії Вікторівні
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів оцінювання змін часу, необхідного для виконання задач, за результатами реалізації довгострокового ІТ-проекту затверджена наказом університету від 01 квітня 2024 р. № 258 Сп
2. Термін подання студентом роботи до екзаменаційної комісії 7 червня 2024 р.
3. Вихідні дані до роботи: науково-технічна література, публікації та інтернет-ресурси, що стосуються теми кваліфікаційної роботи, дані про результати виконання довгострокового проекту «Web Components»
4. Перелік питань, що потрібно опрацювати в роботі аналіз особливостей управління довгостроковими ІТ-проектами, опис та аналіз управління змінами при реалізації довгострокових ІТ-проектів, опис та аналіз сучасних методів управління змінами при реалізації проектів, постановка задачі дослідження, опис дескрипторного підходу для оцінювання змін на рівні задач ІТ-проекту, опис комбінованого методу оцінювання змін часу виконання задач з використанням дескрипторного підходу, сценарії використання комбінованого методу оцінювання змін часу виконання задач при реалізації довгострокового ІТ-проекту, класифікація та опис дескрипторів, які використовуються при реалізації довгострокового ІТ-проекту, аналіз довгострокового проекту розробки інформаційної системи для створення сайтів, аналіз використання комбінованого методу оцінювання змін часу виконання задач ІТ-проекту розробки ІС для створення сайтів, експериментальна перевірка (апробація) розробленого комбінованого методу оцінювання змін часу при реалізації довгострокових ІТ-проектів.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз сучасного стану об'єкта дослідження	01.04.2024-03.04.2024	виконано
2	Аналіз існуючих методів управління змінами при реалізації довгострокових ІТ-проектів	04.04.2024-09.04.2024	виконано
3	Розробка комбінованого методу оцінювання змін часу при реалізації довгострокових ІТ-	10.04.2024-17.04.2024	виконано
4	Опис і аналіз особливостей комбінованого методу оцінювання змін часу при реалізації ІТ-	18.04.2024-20.04.2024	виконано
5	Практична реалізація комбінованого методу	21.04.2024-30.04.2024	виконано
6	Реалізація дослідження дескрипторного підходу при використанні комбінованого	01.05.2024-08.05.2024	виконано
7	Опис даних для дослідження та перевірки розробленого комбінованого методу та дескрипторного підходу	09.05.2024-14.05.2024	виконано
8	Експериментальна перевірка (апробація) розробленого комбінованого методу оцінювання змін часу при реалізації ІТ-	15.05.2024-24.05.2023	виконано
9	Оформлення пояснювальної записки до кваліфікаційної роботи	25.05.2023-29.05.2024	виконано
10	Підготовка презентаційних матеріалів	30.05.2024-02.06.2024	виконано
11	Перевірка роботи на плагіат	03.06.2024	виконано
12	Попередній захист	07.06.2024	виконано
13	Захист кваліфікаційної роботи в екзаменаційній комісії	10.06.2024	виконано

Дата видачі завдання 01 квітня 2024 р.

Студентка _____
(підпис)

Керівник роботи _____ професор каф. ІУС Наталія ВАСИЛЬЦОВА
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи: 127 сторінок, 29 рисунків, 40 таблиць, 2 додатки, 65 джерел.

ДЕСКРИПТОР, ДОВГОСТРОКОВИЙ ІТ-ПРОЄКТ, ЗАДАЧА, ЗМІНА, ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ, ОЦІНЮВАННЯ ЗМІН, УПРАВЛІННЯ ЗМІНАМИ.

Об'єкт дослідження кваліфікаційної роботи – процеси оцінювання змін на рівні задач при виконанні довгострокового ІТ-проєкту.

Мета дослідження – розробка комбінованого методу оцінювання змін часу виконання задач при реалізації довгострокових ІТ-проєктів.

Під час виконання кваліфікаційної роботи було проаналізовано особливості реалізації довгострокових ІТ-проєктів, проаналізовано та порівняно існуючі методи управління змінами в ІТ-проєктах. Було визначено, що в існуючих методах відсутні кількісні параметри для оцінювання змін часу. З метою оцінювання змін часу на рівні задач було розроблено та проаналізовано дескрипторний підхід. Даний підхід полягає у використанні двох процесів: збору даних, щодо виконуваних задач, у вигляді дескрипторів та їх використання.

Було розроблено комбінований метод оцінювання змін, який базується на методі, заснованого на моделі Бекхарда і Гарріса, та дескрипторного підходу.

Було запропоновано показники оцінювання успішності процесу управління змінами.

Комбінований метод був апробований при реалізації проєкту розробки ІС створення сайтів та показав свою ефективність при вирішенні задачі оцінювання змін часу на рівні задач.

ABSTRACT

The explanatory note to the qualification work: 127 pages, 29 images, 40 tables, 2 attachments, 65 sources.

CHANGE, CHANGE ESTIMATION, CHANGE MANAGEMENT, DESCRIPTOR, INFORMATION TECHNOLOGY, LONG-TERM IT PROJECT, TASK.

The object of the research of the qualification work is the processes of assessing changes at the task level during the execution of a long-term IT-project.

The research aim is to develop a combined method for evaluating changes in task execution time during the implementation of long-term IT projects.

During the execution of the qualification work, the features of implementing long-term IT projects were analyzed, and existing change management methods in IT projects were reviewed and compared. It was determined that existing methods lack quantitative parameters for evaluating changes over time. To assess changes over time at the task level, a descriptor-based approach was developed and analyzed. This approach involves two processes: data collection on performed tasks in the form of descriptors and their utilization.

A combined method of change evaluation was developed, which is based on the Beckhard and Harris model and the descriptor approach. Indicators for evaluating the success of the change management process were proposed. The combined method was tested during the implementation of the website development information system project and demonstrated its effectiveness in addressing the task of evaluating changes over time at the task level.

ЗМІСТ

Скорочення та умовні позначки.....	8
Вступ.....	9
1 Опис та аналіз методів управління змінами при реалізації довгострокових ІТ-проектів.....	11
1.1 Аналіз особливостей управління довгостроковими ІТ-проектами...	11
1.2 Опис та аналіз процесів управління змінами при реалізації довгострокових ІТ-проектів.....	13
1.3 Опис та аналіз сучасних методів управління змінами при реалізації проектів.....	20
1.4 Постановка задачі дослідження.....	32
2 Розробка комбінованого методу оцінювання змін часу виконання задач при реалізації ІТ-проекту.....	34
2.1 Опис дескрипторного підходу для оцінювання змін на рівні задач ІТ-проекту.....	34
2.2 Опис комбінованого методу оцінювання змін часу виконання задач з використанням дескрипторного підходу.....	37
2.3 Висновки з розробки комбінованого методу оцінювання змін часу виконання задач при реалізації довгострокового ІТ-проекту.....	45
3 Дослідження особливостей реалізації комбінованого методу оцінювання змін часу виконання задач при реалізації ІТ-проекту.....	46
3.1 Сценарії використання комбінованого методу оцінювання змін....	46
3.1.1 Основні етапи реалізації комбінованого методу	46
3.1.2 Реалізація етапу «Внутрішній проектний аналіз».....	47
3.1.3 Реалізація етапу «Збір інформації у вигляді дескрипторів на кожній ітерації проекту».....	49
3.1.4 Реалізація етапу «Визначення потреби в змінах».....	50
3.1.5 Реалізація етапу «Аналіз відмінностей між поточним станом	

та бажаним».....	51
3.1.6 Реалізація етапу «Визначення можливостей реалізації змін на поточній ітерації виконання проєкту».....	52
3.1.7 Реалізація етапу «Планування дій».....	67
3.1.8 Реалізація етапу «Управління переходом»	67
3.1.9 Реалізація етапу «Оцінювання успішності процесу управління змінами».....	68
3.2 Класифікація та опис дескрипторів, які використовуються при реалізації довгострокового ІТ-проєкту	69
3.3 Висновки з дослідження особливостей реалізації комбінованого методу оцінювання змін часу виконання задач.....	73
4 Апробація комбінованого методу оцінювання змін часу виконання задач при реалізації ІТ-проєкту.....	75
4.1 Аналіз довгострокового проєкту розробки інформаційної системи для створення сайтів.....	75
4.2 Апробація комбінованого методу оцінювання змін часу виконання задач ІТ-проєкту розробки ІС для створення сайтів.....	82
4.3 Оцінювання успішності процесу управління змінами.....	94
4.4 Висновки з четвертого розділу кваліфікаційної роботи.....	96
Висновки.....	98
Перелік джерел посилання.....	100
Додаток А Програмна реалізація етапів комбінованого методу.....	106
Додаток Б Графічний матеріал.....	112

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

IC – інформаційна система

IT – інформаційна технологія

ADKAR – Awareness Desire Knowledge Ability Reinforcement

AEM – Adobe Experience Manager

AIM – Accelerating Implementation Methodology

AWS – Amazon Web Services

BA – business analysis

BE – Back End

CR – change request

CSS – cascading Style Sheets

DevOps – development operations

DoD – definition of done

FE – Front End

HTML – hypertext mark-up language

JS – JavaScript

LOOCV – leave-one-out cross-validation

npm – node package manager

OLS – ordinary least squares

POC – proof of concept

QA – quality assurance

SA – solution architects

SQL – structured query language

TS – TypeScript

ВСТУП

Оцінювання змін часу виконання задач при реалізації довгострокових ІТ-проектів є важливою частиною процесу управління змінами та управління проектів в цілому.

Актуальність дослідження зумовлена зростанням ризиків які мають місце при виконанні сучасних довгострокових ІТ-проектів. Швидкий розвиток технологій призводить до необхідності гнучкого підходу впровадження та оцінювання змін з метою підтримання конкурентоспроможності компаній.

Якісне оцінювання змін часу виконання задач має прямий вплив на успішність реалізації ІТ-проектів. Не якісна оцінка змін часу може призвести до перевищення термінів виконання проекту, перенапруження ресурсів та непередбачуваних затримок у запуску продукту на ринок.

Дієве управління змінами у виконанні ІТ-проектів передбачає систематичне оцінювання та аналіз змін у часі виконання задач, щоб вчасно виявляти потенційні проблеми та ризики.

Зокрема, систематичне оцінювання змін часу виконання дозволяє управлінню проектами приймати обґрунтовані рішення щодо ресурсного планування, розподілу завдань та пріоритетів у випадку змін у вимогах чи умовах проекту. Такий підхід сприяє зменшенню ризиків затримок у виконанні проекту та забезпечує його успішне завершення в рамках встановлених термінів.

У цій роботі досліджується процес оцінювання змін часу виконання завдань під час реалізації довгострокових ІТ-проектів.

Об'єкт дослідження кваліфікаційної роботи – процеси оцінювання змін на рівні задач при виконанні довгострокового ІТ-проекту.

Мета дослідження – розробка комбінованого методу оцінювання змін часу виконання задач при реалізації довгострокових ІТ-проектів.

Для досягнення цієї мети робота передбачає вирішення таких завдань:

- проаналізувати особливості реалізації довгострокових ІТ-проектів;
- проаналізувати та узагальнити існуючі методи управління змінами на підприємствах;
- визначити критерії вибору методу управління змінами на підприємствах відповідно до мети його адаптації для використання на рівні задач з можливістю оцінювання змін;
- розробити підхід оцінювання змін часу при реалізації довгострокових проектів;
- розробити комбінований метод оцінювання змін з використанням існуючого методу управління змінами при реалізації ІТ-проектів;
- практично реалізувати запропонований метод на прикладі ІТ-проекту розробки системи створення сайтів;
- експериментально перевірити ефективність розробленого комбінованого методу управління змінами часу на рівні задач.

1 ОПИС ТА АНАЛІЗ МЕТОДІВ УПРАВЛІННЯ ЗМІНАМИ ПРИ РЕАЛІЗАЦІЇ ДОВГОСТРОКОВИХ ІТ-ПРОЄКТІВ

1.1 Аналіз особливостей управління довгостроковими ІТ-проєктами

На сьогоднішній день ІТ-проєкти охоплюють такі основні сфери діяльності: розробку і розвиток програмних застосунків, впровадження інформаційних систем і технологій (ІТ), розгортання ІТ-інфраструктури.

Множину таких проєктів можна класифікувати за різними ознаками: за спрямованістю (або сферою діяльності), за масштабом (за обсягами фінансування), за тривалістю реалізації, за складністю реалізації, за призначенням, за характером учасників, за залежністю від вимог до якості, за географічною ознакою тощо [1].

Для довгострокових проєктів, які розглядаються в кваліфікаційній роботі, кожна з цих ознак суттєво впливає на ефективність виконання усіх етапів життєвого циклу проєкту.

Зараз можна спостерігати широке використання довгострокових проєктів, термін виконання яких складає від 3 і більше років. Зокрема, серед них можна виділити мегапроєкти, тривалість реалізації яких складає від 5 років та більше. Такі проєкти є характерними й для різних галузей ІТ-сфери: розробки та впровадження ІТ, інформаційних системи (ІС), Т-інфраструктури, наукових досліджень в рамках ІТ тощо.

Аналіз багатьох довгострокових ІТ-проєктів показує, що основну увагу у таких проєктах треба приділяти наступним характеристикам: тривалості, складності, витратам ресурсів, варіативності, ризикам, етапності виконання.

Такому типу проєктів характерний високий рівень складності через велику кількість етапів, взаємопов'язаних задач та залучення багатьох учасників.

Зокрема, довгостроковим проєктам характерний високий рівень невизначеності та ризиків, що можуть виникати на різних етапах проєкту. Це

можуть бути фінансові ризики, технічні складнощі, проблеми з постачанням матеріалів тощо.

Такі проекти часто розбиваються на декілька етапів або фаз, кожна з яких має свої цілі, завдання та результати, що дозволяє краще контролювати прогрес і вчасно вносити необхідні корективи [2].

Аналіз перелічених характеристик сучасних довгострокових ІТ-проектів дозволяє виділити їх основні особливості [3]:

- складність опису, яка спричинена великою кількістю функцій, процесів, елементів, даних та зв'язків між ними;
- наявність сукупності підсистем, які мають тісні взаємозв'язки та мають свої локальні цілі та задачі;
- необхідність інтеграції існуючих і повторно розроблюваних застосувань;
- різноманітність окремих груп розробників за рівнем класифікації та сформованих традицій використання інструментальних засобів;
- постійний потік змін в сучасних інформаційних системах.

Довгострокові проекти часто вимагають комплексного підходу до управління ресурсами, їхнього планування та оптимізації.

До особливостей управління довгостроковими ІТ-проектів можна віднести наступні:

- детальне планування на початковому етапі, що включає визначення цілей, задач, ресурсів, бюджетів та термінів, а також важливість передбачувати можливість і необхідність коригувань планів у процесі реалізації проекту);
- постійний моніторинг прогресу проекту, контроль виконання задач та використання ресурсів, що включає регулярні звіти, ревію, зустрічі команди;
- впровадження ефективних процедур для управління змінами, що включають ідентифікацію, оцінку, затвердження та впровадження змін;
- визначення потенційних ризиків, їх оцінювання та розробка планів

для їхньої мінімізації або уникнення, регулярне оновлення планів ризик-менеджменту;

- забезпечення ефективної комунікації між усіма учасниками проєкту, що включає внутрішню комунікацію в команді та зовнішню комунікацію із зацікавленими сторонами;

- підтримка високого рівня мотивації в команді, організація навчань та розвиток навичок, управління конфліктами та забезпечення сприятливої робочої атмосфери.

1.2 Опис та аналіз процесів управління змінами при реалізації довгострокових ІТ-проєктів

Одним із важливих процесів в управлінні довгостроковими ІТ-проєктами є управління змінами, які вносяться при реалізації проєкту.

Під зміною в теорії управління проєктами розуміють перехід від одного рішення до іншого внаслідок впливу зовнішніх і внутрішніх чинників під час реалізації проєкту. Якісне управління цим переходом підвищує шанси на успішний результат та позитивно впливає на команду, проєкт та організацію в цілому [1].

Аналіз особливостей довгострокових ІТ-проєктів та особливостей управління цими проєктами показав, що зміни у вимогах та умовах, що відбуваються протягом життєвого циклу ІТ-проєкту, можуть бути зумовлені зовнішніми та внутрішніми чинниками.

До зовнішніх чинників змін в ІТ-проєктах відносяться:

- зміни ринкових умов (конкуренція та попит на ринку);
- технологічні фактори (нові технології, технологічні інновації та безпека);
- регуляторні вимоги (законодавство та стандарти);

- соціально-економічні фактори (економічні умови та соціальні тенденції);

- політичні фактори.

До внутрішніх чинників змін в ІТ-проєктах відносяться:

- зміни стратегічних рішень, бізнес-стратегій та пріоритетів;

- вимоги до ресурсів, в тому числі до трудових;

- управлінські процеси (методологія розробки, інструменти та технології);

- зміни вимог до якості та продуктивності (оцінка якості та ефективність команди);

- вимоги та специфікації (зворотній зв'язок від клієнтів та внутрішні ініціативи).

Зміни можуть мати наступний вплив на ІТ-проєкт: підвищувати витрати, спричиняти затримку виконання проєкту, знижувати продуктивність праці виконавців робіт, погіршувати стосунки між членами команди.

Причинами внесення змін, як правило, є неможливість прогнозування нових проєктних рішень, технологій тощо на етапі розробки проєкту, а також відставання від запланованих термінів та обсягів при реалізації проєкту через непередбачені обставини [3], [4].

Зміни можуть відбутися на будь-якій стадії реалізації ІТ-проєкту і мати такі значення та наслідки:

- зміни архітектури або обсягу проєкту на стадії розробки (це природно, але дуже часто вони приймаються без належної оцінки наслідків з точки зору часу та вартості, після затвердження дизайну ці зміни можуть бути значно дорожчими);

- пізні зміни в архітектурі (ці зміни є найдорожчими, створюються в результаті помилок на етапі розробки архітектури або для того, щоб спробувати реалізувати задачу відповідно до вимог часу, використовуючи останні досягнення техніки, що призводить до збільшення обсягу роботи);

– зміни, які вимагаються законодавством або безпековими обмеженнями (вони є обов'язковими до внесення та мають бути внесені на рівні управління проектами);

– зміни для підвищення рентабельності та фінансової віддачі від проекту (питання про доцільність цих змін вирішує топ-менеджмент компанії відповідно до своєї політики, зокрема дуже важко точно розрахувати вартість зміни та майбутні грошові потоки) [4].

Управління змінами відображається як процес оцінки та планування майбутніх змін, запис усіх виявлених змін (у змісті проекту, специфікації, вартості, плані, мережевому графіку тощо) для детального вивчення, оцінки впливу, затвердження або відхилення, моніторингу та координаційної організації працівників, що здійснюють зміни в проекті.

Ініціаторами змін можуть бути: замовник, інвестор, архітектор, менеджер, бізнес-аналітик, інженер з тестування (quality assurance, QA).

Замовник, як правило, вносить зміни, що покращують кінцеві техніко-економічні характеристики проекту, робить зміни в бюджеті або часі виконання завдань.

Архітектор може вносити зміни до початкової проектної архітектури або документації, методики та технології виконання робіт, послідовності (технологічної) побудови об'єктів тощо.

Менеджер під час реалізації проекту вносить зміни в календарний план або склад команди [5].

Управління змінами – це комплексний, циклічний і структурований підхід до зміни окремих осіб, груп і організацій із поточного стану на майбутній із очікуваними перевагами для бізнесу. Це допомагає організаціям інтегрувати та вирівнювати людей, процеси, структури, культуру та стратегію [6].

До розробки концепції управління змінами в IT-проекті відносяться: вироблення стратегії управління змінами, аналіз можливих змін,

формулювання принципів інтеграції процесів управління змінами та затвердження концепції.

До оцінки та планування змін можна віднести: вибір методів і засобів оцінки і планування змін та прогнозування змін, моніторинг зовнішнього середовища і тенденцій змін, планування можливих попереджувальних впливів для захисту ІТ-проєкту та розробка плану управління змінами в проєкті [5].

До організації та контролю змін в проєкті відносяться [5], [6]:

- розподіл ролей та визначення відповідальності для персоналу, який бере участь в управлінні змінами, і формування відповідної організаційної структури;

- встановлення процедур для впровадження змін у проєкті;
- впровадження системи управління змінами;
- надання інформаційної підтримки для управління змінами у проєкті;
- збір та аналіз запитів та пропозицій щодо внесення змін;
- прийняття рішень та впровадження змін в проєкт;
- ведення бази даних змін проєкту.

До аналізу та регулювання змін можна віднести:

- моніторинг впровадження змін у проєкті;
- аналіз та огляд динаміки змін в проєкті;
- постійну оцінку змін в проєкті та досягнутих результатів в результаті здійснення змін;
- подання звіту щодо виконання змін у проєкті та відхилень від плану управління змінами;

- пропозиції щодо внесення коректив до плану змін.

До етапу завершення управління змінами в проєкті можна віднести:

- проведення післяпроєктного аналізу та оцінку змін та їх результатів;
- підготовку заключного звіту про фактичні зміни в проєкті;
- формування архіву змін, які мали місце при реалізації проєкту;

– адаптацію стратегії на майбутнє та створення бази знань з управління змінами.

Вимоги процесу управління змінами мають бути закріплені в документах компанії та визначати наступне:

- перелік параметрів проєкту, що повинні бути обов'язково затверджені;
- розмір відхилень параметрів проєкту, при яких необхідне обов'язкове підтвердження планів проєкту;
- процедуру внесення змін в проєкт;
- порядок затвердження та реєстрацію змін.

У довгострокових ІТ-проєктах часто виникає необхідність змін на рівні вирішення задач через такі причини:

- зміни бізнес-вимог;
- технологічні оновлення;
- зворотний зв'язок від користувачів;
- виявлення нових можливостей для поліпшення ІТ-продукту.

Зміни на рівні задач суттєво впливають на всі рівні проєкту, включаючи емоційну складову кожного співробітника та інші, тому важливо мати ефективні механізми для їх кількісної оцінки [7].

На сьогодні в ІТ-проєктах часто можна спостерігати ситуації, коли відбувається зміна вимог до задач, які вже знаходяться в роботі або які вже оцінені командою. Зміни можуть бути пов'язані: з розробкою дизайну графічного інтерфейсу, з певними функціональними вимогами або технічними підходами, технологіями, що використовуються, тощо. Інколи складність задачі стає зрозумілою розробникам та архітекторам вже під час виконання самої задачі. В деяких випадках створюються спеціальні задачі для реалізації цих змін. Інколи зміни виконуються в ході реалізації самої задачі, в якій була помічена необхідність цих змін.

В проєктних командах зміни можуть бути пов'язані з тим, що відбувається скорочення ресурсів, скорочення виплат (наявні лікарняні або

незапланована відпустка, з іншої причини). Інколи зміни пов'язані з тим, що першочергове завдання було описано не повністю або не якісно та було невірно оцінено зі сторони бізнес-аналітиків або інших членів команди.

Ці зміни призводять: до ризиків та необхідності дій зі сторони менеджменту та команди, до прийняття рішень про можливе заохочення для понаднормової роботи або перенесення релізу, до створення нових задач після релізу для реалізації змін тощо.

Менеджерам в такому випадку треба приймати рішення про те, який саме підхід краще обрати та врахувати всі можливі ризики.

Зміни є значною сферою конфліктів, зокрема для ІТ-проектів.

Завданням керівника проекту є усунення непотрібних змін та встановлення чіткої межі між «потрібними» і «бажаними», вносячи лише ті зміни, які є необхідними для виконання визначеного обсягу задачі та вимог безпеки. Загальний контроль змін здійснюється: для оцінки впливу факторів, що призводять до позитивних чи негативних змін у проекті, для визначення, чи вже відбулися зміни в проекті, для керування змінами в проекті під час його реалізації.

Зміни в ІТ-проектах можуть відбуватись на всіх рівнях: стратегічний, тактичний, операційний, рівень спрінтів, командний рівень. Опис змін на різних рівнях ІТ-проектів наведений в таблиці 1.1.

Таблиця 1.1 – Опис змін на різних рівнях ІТ-проектів

Рівень	Зміни на заданому рівні
1	2
Стратегічний рівень	Перегляд і коригування довгострокової візії та цілей проекту відповідно до змін в бізнес-середовищі або стратегії компанії. Зміни у фінансуванні або ресурсах, доступних для проекту, що можуть впливати на його обсяги або строки. Визначення нових пріоритетів для функціональності або змін у продукті на основі ринкових досліджень, зворотного зв'язку від клієнтів або внутрішніх потреб бізнесу.

Кінець таблиці 1.1

1	2
Тактичний рівень	<p>Коригування планів релізів з метою включення нових функцій або відкладання запланованих через зміну пріоритетів або затримки.</p> <p>Перерозподіл команд і ресурсів для вирішення нових задач або відповідно до змін у графіку.</p> <p>Внесення змін до методологій роботи (наприклад, перехід з методології Waterfall на методологію Agile) для підвищення ефективності роботи команди.</p>
Операційний рівень (рівень задач)	<p>Зміна складу або пріоритетів задач, над якими працюють команди в спринтах, щоб адаптуватися до нових вимог або умов. Зміни на рівні задач.</p> <p>Перегляд і коригування технічних рішень, таких як: архітектура системи, інструменти, фреймворки та технології.</p> <p>Уточнення або зміна вимог і специфікацій функціональності продукту для врахування нових потреб або виправлення недоліків.</p>
Рівень спринтів	<p>Регулярне оновлення та визначення пріоритетів задач спринту на основі зворотного зв'язку, прогресу виконання задач та змін у бізнес-вимогах.</p> <p>Використання щоденних зустрічей для виявлення перешкод, координації дій та оперативного вирішення проблем.</p> <p>Регулярні оцінки прогресу та ретроспективи для аналізу виконання спринтів, що дозволяють вносити зміни для покращення майбутніх спринтів.</p>
Командний рівень	<p>Впровадження нових інструментів або процесів для поліпшення комунікації та співпраці в команді.</p> <p>Організація навчальних сесій або тренінгів для підвищення кваліфікації членів команди відповідно до нових вимог або технологій.</p>

В таблиці 1.2 наведені важливі метрики для вимірювання ефективності процесу управління змінами в ІТ-проектах [8].

Таблиця 1.2 – Метрики для вимірювання ефективності процесу управління змінами

Показник	Опис показника	Аналіз показника
Кількість несанкціонованих змін	Кількість неавторизованих змін, виявлених за певний період часу	Менше число вказує на те, що процес виявлення та затвердження змін є надійним і здатним використовуватися при управлінні всіма змінами.
Відсоток скасованих змін	Відсоток змін, які не використовувались через проблеми під час впровадження	Більше число свідчить про погано сплановані зміни
Рівень прийняття змін	Відсоток змін, які були схвалені комітетом змін	Результат вказує на ефективність запитів на зміни та планів змін. Більше число означає, що зміни та планування надійні
Відхилення від графіка	Відхилення між часом, який був витрачений на виконання робіт, і оціненим часом	Результат вказує на те, чи виконуються зміни вчасно і дотримуються плану змін
Кількість інцидентів, спричинених зміною	Кількість інцидентів, які виникли внаслідок певної зміни	Результат показує, чи впливає зміна на інші операції служб. Висока кількість інцидентів свідчить про те, що зміни потрібно краще обговорювати
Відсоток змін, завершених вчасно	Відсоток змін, які були завершені вчасно протягом певного періоду часу	Результат вказує на те, наскільки ефективно виконується процес управління змінами. Чим вище відсоток, тим краще виконується процес управління змінами

1.3 Опис та аналіз сучасних методів управління змінами при реалізації проєктів

На сьогоднішній день найбільш поширеними методами управління змінами у проєктах є:

- метод, заснований на моделі ADKAR та ADKAR PROSCI [9], [10];
- метод AIM [11], [12];
- метод, заснований на моделі Бекхарда і Гарріса [13];
- метод, заснований на моделі Вільяма та Бріджа [14];
- метод, заснований на моделі Джона Коттера [15];
- метод, заснований на моделі Коблера Росса [16];
- метод, заснований на моделі Курта Левіна [17].

Метод, заснований на моделі ADKAR, був створений Джефрі Хаяттом наприкінці 1990-х років. Він пропонує глибокий підхід до управління змінами, зосереджений на персоналі. Передбачається, що, в процесі впровадження змін, персоналу необхідно чітко роз'яснити причини змін та їхню важливість, що сприяє залученню персоналу до процесу. Далі важливо навчити кожного співробітника методам виконання змін, що дозволяє показати рівень їхніх професійних знань і навичок у процесі внесення змін. Завершальний етап передбачає закріплення змін, забезпечуючи сталість інновацій у діяльності організації [9].

ADKAR – це скорочення, яке представляє п'ять ключових етапів, необхідних для успішного здійснення змін:

- «Усвідомлення» (Awareness);
- «Бажання» (Desire);
- «Знання» (Knowledge);
- «Здатність» (Ability);
- «Підкріплення» (Reinforcement).

На етапі «Усвідомлення» необхідно розуміти, чому зміна необхідна та як вона впливатиме на організацію або проєкт. Також важливо провести комунікаційну кампанію, щоб проінформувати всі зацікавлені сторони про потребу зміни та її переваги.

На етапі «Бажання» необхідно стимулювати бажання та мотивацію персоналу до зміни. Для цього розробляються та впроваджуються мотиваційні заходи, які будуть підтримувати бажання сторін в зміні.

На етапі «Знання» необхідно забезпечити персонал необхідними знаннями та навичками для успішної реалізації змін. Зокрема, можуть бути проведені тренінги, навчання та розробка матеріалів для передачі необхідної інформації.

На етапі «Здатність» працівники отримують можливість практикувати та використовувати нові знання та навички в реальних умовах. Для цього необхідно забезпечити підтримку та можливість для практичного застосування отриманих навичок.

На етапі «Підкріплення» необхідно забезпечити постійну підтримку та проведення оцінки після внесених змін. Для цього впроваджуються системи постійної оцінки та винагороди за досягнення результатів [10].

При використанні ADKAR важливим є побудова та використання ADKAR-схеми. ADKAR-схема – це схема, або план, що допомагає крок за кроком впровадити зміни. Він складається з конкретних кроків та вказівок, що допомагають зрозуміти, як перейти від початку зміни до її успішного завершення. Адже зміна може бути складною, і ADKAR Blueprint допомагає зрозуміти, що робити на кожному етапі, щоб забезпечити успішний результат. Для побудови ADKAR Blueprint необхідно виконати наступні етапи:

- визначення конкретних моментів, коли зміна буде відбуватися, і ключових віх, які слід досягти;
- визначення чітких цілей для кожної стадії зміни та встановлення реалістичних термінів для досягнення цих цілей;
- визначення можливих перешкод та ризиків, які можуть виникнути під час зміни, та розробка планів їх управління;
- встановлення ефективної системи комунікації з усіма зацікавленими сторонами та залучення їх до процесу зміни.

Основні переваги цього методу полягають у тому, що він надає чіткий керівний механізм для процесу змін, починаючи з розуміння персоналом мотивації та цілей змін. До переваг методу можна віднести наступне:

- розглядає зміну з позиції індивідуальних потреб та перешкод, що дозволяє ефективно керувати внутрішнім опором та мотивацією;
- можна систематично оцінювати прогрес кожної особи чи групи в процесі зміни;
- допомагає зменшити ризики неуспіху зміни та забезпечити більшу ймовірність успішної імплементації.

Метод ADKAR PROSCI є частиною більш широкого підходу до управління змінами, розробленого компанією PROSCI. Він базується на методі з використанням моделі ADKAR, але включає додаткові інструменти, методи та практики, спрямовані на ефективне впровадження змін в організації [10].

Метод прискорення впровадження (Accelerating Implementation Methodology, AIM) є потужним та дисциплінованим методом для управління організаційними змінами, включаючи трансформаційні зміни до повного повернення інвестицій. Це інтегрована система операціоналізованих принципів, стратегій, тактик, аналітики вимірювання та інструментів, що підтримана сертифікаційними та навчальними програмами. AIM можна застосовувати до будь-якого виду ініціативи або проєкту. Але більшість організацій спрямовують основні ресурси та енергію на технічні та бізнес-процесні компоненти [11], [12].

Метод AIM пропонує підхід, що спрямований на визначення можливих проблем у процесі впровадження змін. Особливу увагу приділяють відмінностям установки змін та впровадженню змін, тому що без відповідного управління процесом впровадження змін наслідки можуть бути неочікуваними та незадовільними.

В методі виокремлюють декілька таких ролей:

- чемпіони (люди, що підтримують зміни, але не мають можливості їх робити);
- агенти (люди, які мають змогу вносити зміни);
- спонсори (люди, що управляють змінами);

– цілі (люди, на яких відбувається вплив змін та які мають розробити нові звички).

Одна особа може відігравати декілька ролей одночасно.

Цей метод включає десять етапів, які можна розділити на три основні групи: планування, виконання та моніторинг процесу впровадження.

До етапів методів відносять: «Визначення змін», «Формування потужності агентів», «Оцінка клімату», «Створення спонсорства»; «Визначення стратегії змін», «Розробка готовності до змін», «Розробка плану комунікації», «Розробка стратегії підсилення», «Створення культурної відповідності», «Пріоритизація дій».

Етап «Визначення змін» передбачає визначення потреби в зміні, а також опис того, що саме потрібно змінити і чому. Важливо визначити цілі та області змін для успішної реалізації проєкту. На етапі «Формування потужності агентів» створюються необхідні ресурси та потужності для впровадження змін. Це може включати навчання персоналу, розвиток кадрів або створення команд змін. На етапі «Оцінка клімату» аналізується поточна організаційна атмосфера і стан готовності персоналу до змін. Важливо зрозуміти, які фактори можуть сприяти або перешкоджати успішному впровадженню змін. На етапі «Створення спонсорства» залучаються ключові учасники та підтримка високого рівня для забезпечення успіху проєкту. Це можуть бути керівники, які забезпечують ресурси, підтримують важливі рішення та допомагають вирішувати конфлікти. На етапі «Визначення стратегії змін» обирається підхід до реалізації змін, включаючи методи й інструменти, які будуть використовуватися. Важливо обрати підходящий метод для досягнення поставлених цілей. На етапі «Розробка готовності до змін» готуються необхідні умови та ресурси для успішного впровадження змін. Це включає навчання персоналу, створення необхідних інфраструктурних змін та впровадження підтримки. На етапі «Розробка плану комунікації» розробляється стратегія комунікації для ефективного спілкування з усіма зацікавленими сторонами проєкту. Важливо забезпечити

чітку та своєчасну інформаційну взаємодію. На етапі «Розробка стратегії підсилення» створюється план дій для збереження та підтримки змін після їх впровадження. Це може включати мотиваційні заходи, навчання та систему винагород. На етапі «Створення культурної відповідності» працюють над забезпеченням того, щоб запропоновані зміни були відповідні організаційній культурі. Це може включати адаптацію стратегій та практик до цінностей та норм організації. На етапі «Пріоритезація дій» визначаються пріоритетні завдання та дії для впровадження змін. Важливо визначити найбільш важливі та критичні кроки для успішного завершення проєкту.

Однією з переваг цього методу є можливість систематичного аналізу та передбачення можливих труднощів у процесі змін, що дозволяє організаціям ефективніше підготуватися та відповісти на них. Однак цей підхід може вимагати значних ресурсів та часу на його виконання, а також може стати складним у випадку несприятливих обставин чи непередбачуваних подій у процесі впровадження змін [12].

Метод, заснований на моделі Бекхарда і Гарріса представляє п'ять етапів управління змінами, спрямованих на визначення потреби в змінах, розробку стратегії їх реалізації, формування плану дій та визначення відповідальних виконавців. Етапами управління змінами відповідно до методу, заснованому на моделі Бекхарда і Гарріса, є:

- «Внутрішній організаційний аналіз»;
- «Визначення потреби в змінах»;
- «Аналіз відмінностей між поточним станом та бажаним»;
- «Планування дій»;
- «Керування переходом».

Метою етапу «Внутрішній організаційний аналіз» є визначення загального ставлення до змін в організації.

На цьому етапі необхідно ідентифікувати співробітників, які можуть мати значний опір до змін. Зокрема, необхідно визначити будь-які зовнішні чинники, які можуть перешкоджати процесу змін.

Етап «Визначення потреби в змінах» необхідний для того, щоб створити підставу для впровадження змін. Ключові учасники змін повинні погодитися з тим, що зміни є необхідними для успіху організації. Це передбачає, що учасники змін можуть чітко сформулювати, куди вони хочуть привести організацію і чому впровадження змін допоможе наблизити організацію до бажаного стану. Також необхідно мати розуміння наслідків, пов'язаних з відмовою від впровадження змін [13].

Наступним є етап «Аналіз відмінностей між поточним станом та бажаним». Перед проведенням впровадження змін спочатку необхідно визначити, які відхилення існують між поточним станом організації і тим, яким вона повинна бути. Визначення цих відхилень важливо для того, щоб чітко сформулювати розуміння майбутнього організації.

На етапі «Планування дій» реалізується план змін. Необхідно визначити ключових учасників процесу змін та обов'язки кожного, хто вносить зміни.

Наступним є етап «Керування переходом». Після того, як зміна буде здійснена, співробітники, що здійснюють зміни, відповідають за безперервний моніторинг прогресу змін і внесення коригувань.

Основною перевагою методу, заснованому на моделі Бекхарда і Гарріса, є систематичний підхід до процесу змін, що дозволяє структурувати та керувати ними ефективно. Компанія Praxie з використанням методу, заснованому на моделі Бекхарда і Гарріса, створила програмні застосунки для управління змінами та надає можливість проходити тренінги [13], [14].

Метод, заснований на моделі переходу Бриджеса, надає можливість організаціям та особам краще розуміти та ефективно керувати особистою та людською стороною змін. Метод ідентифікує три етапи, які проходить кожна людина під час змін: завершення того, що є зараз; нейтральна зона; новий початок.

Розроблений Вільямом Бриджесом метод, заснований на моделі переходу, використовується лідерами та консультантами з управління більше

ніж тридцять років. Перехідна модель від William Bridge фокусується на внутрішньому переході організації, який необхідно здійснити, а не на самій зміні. Цей метод передбачає, що зміни відбуваються в принципах роботи організації, а не в особистісних характеристиках її членів.

Основною перевагою цього методу є його спрямованість на внутрішній перехід, що може забезпечити більш глибоку та стійку зміну організаційного середовища. Однак, він може виявитися менш ефективним у випадку, коли потрібно швидко реагувати на зовнішні та непередбачувані зміни, оскільки фокусується на внутрішньому аспекті переходу [13].

Управління переходом в організаціях стосується внутрішнього психологічного процесу, який переживають люди під час змін.

Успішне управління переходом включає наступні кроки:

- комунікація з організацією щодо причини, чому потрібна зміна;
- збір інформації від тих, хто постраждає від змін, для розуміння їх впливу на них, отримання їхньої інвестиції в результат;
- проведення аудиту готовності організації до переходу;
- навчання лідерів тому, як зміна впливатиме на особистості в організації для ефективного управління переходом;
- моніторинг прогресу осіб, який вони проходять через три етапи переходу;
- допомога особам зрозуміти, як вони можуть позитивно вплинути на зміни та важливість їх ролі в організації [14].

Метод, заснований на моделі Джона Коттера, покликаний підвищити залученість персоналу до управління змінами та забезпечити їхню прийнятність для всіх працівників. Метод включає вісім етапів, пропуск одного з яких може призвести до проблем у внесенні змін та ускладнити процес змін.

Основна перевага методу полягає у його систематичному підході до управління змінами та акценті на залученні персоналу до процесу.

Метод, заснований на моделі Джона Коттера, включає в себе наступні етапи:

- «Створення емоційної необхідності для зміни»;
- «Створення коаліції»;
- «Складання бачення»;
- «Розповсюдження бачення»;
- «Впровадження дій»;
- «Формування короткострокових досягнень»;
- «Закріплення досягнень»;
- «Впровадження змін у культуру» [15].

На етапі «Створення емоційної необхідності для зміни» відбувається фокусування на створенні у свідомості співробітників розуміння, що зміни необхідні для існування та процвітання організації. На етапі «Створення коаліції» формується група ключових учасників, які підтримують зміни і можуть стати лідерами впровадження змін. На етапі «Складання бачення» лідер організації розробляє чітке бачення майбутнього стану організації, що дозволяє співробітникам розуміти, куди вони рухаються та які цілі слід досягти. На етапі «Розповсюдження бачення» висвітлюється та поширюється серед всіх членів організації бачення, щоб забезпечити загальне розуміння та підтримку. На етапі «Впровадження дій» виконуються конкретні кроки для реалізації змін, зокрема, зміна процесів, структур та практик організації відповідно до визначеної візії. На етапі «Формування короткострокових досягнень» швидкі досягнення результатів демонструють успіхи впровадження змін та підтримують мотивацію працівників. На етапі «Закріплення досягнень» успішні результати заохочують до подальших зусиль та зміцнюють впевненість у здійсненні змін.

На етапі «Впровадження змін у культуру» зміни закріплюються в культурі організації, роблячи їх сталою та невід'ємною частиною способу життя в організації [15].

До недоліків методу, заснованому на моделі Коттера, можна віднести наступне:

- реалізація всіх восьми етапів методу може вимагати значних зусиль та часу, особливо у великих організаціях або при великому масштабі змін;
- метод фокусується переважно на організаційних аспектах змін, залишаючи осторонь індивідуальні емоції та потреби працівників;
- метод не завжди враховує зовнішні впливи, такі як економічні чи політичні фактори, які можуть впливати на успішність змін.

Метод, заснований на моделі Кюблера-Росса, описує етапи зміни поведінки персоналу, включаючи: опір персоналу до змін, неусвідомлення наслідків змін, адаптацію персоналу до нових умов праці, позитивне ставлення працівників до змін та їх прийняття. Цей метод важливий для розуміння та прогнозування реакцій персоналу на зміни в організації. Однак, варто враховувати, що реакція на зміни може бути індивідуальною та не завжди відповідає цій послідовності етапів. В даному методі розглядається крива змін, яка відображає емоційний стан співробітників. Корисність цього методу також розповсюджується в корпоративному світі для кращого розуміння емоційних турбот, з якими стикається співробітник через будь-яку ініціативу змін на робочому місці, таку як впровадження нового програмного забезпечення чи вдосконалення бізнес-процесів. З практичної сторони метод може бути використаний для виявлення будь-яких перешкод у ранніх етапах змінних проєктів та розробки відповідних стратегій.

До основних переваг методу можна віднести те, що він є простим, але ефективним для управління організаційними змінами.

Недоліком методу є те, що всі співробітники реагують з різною швидкістю, що призводить до того, що співробітники перебувають на різних етапах в кривій змін. Це часто призводить до проблем планування змін [16].

На рисунку 1.1 зображена крива змін Кюблера-Росса.

Метод, заснований на моделі Курта Левіна, що розроблений в середині 20 століття, залишається одним з найбільш популярних на сучасному етапі

розвитку управління змінами. Цей метод включає три основні етапи: «Розморожування», «Впровадження змін», «Замороження».

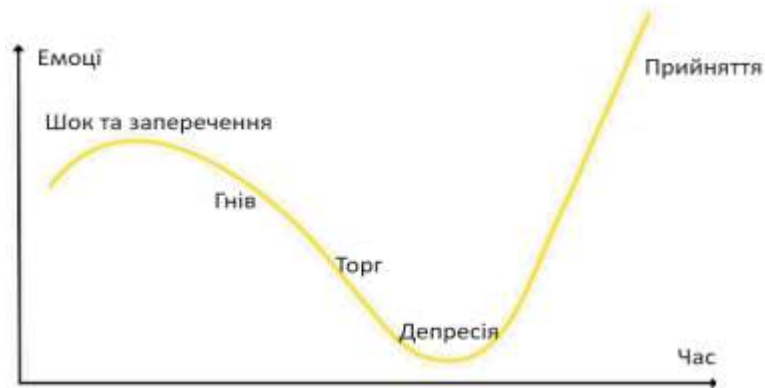


Рисунок 1.1 – Крива змін Кюблера-Росса

Етап «Розморожування» передбачає виявлення та розкриття шляхів, які сприяють зниженню опору серед персоналу організації щодо змін. Важливою складовою цього етапу є створення сприятливого середовища, де співробітники відчувають відкритість до нововведень та готовність до співпраці у впровадженні змін. На етапі «Впровадження змін» зміни впроваджуються. На цій стадії важливо використовувати гнучкий та ітеративний підхід, який враховує відгуки співробітників, щоб згладити перехід. На етапі «Замороження» співробітники відходять від перехідної фази до стабілізації або прийняття. Однак, якщо лідерам змін не вдасться посилити зміни в культурі організації, то співробітники можуть повернутися до попередньої поведінки.

Метод, за оснований на моделі Курта Левіна, відзначається своєю системністю та практичністю, що робить його ефективним для впровадження змін у різних типах організацій. Однак, він може вимагати значних ресурсів та часу на виконання, а також виявитися менш ефективним у випадку складних організаційних структур чи невивіреного застосування методів управління змінами [17].

Для порівняння розглянутих методів управління змінами у проектах запропоновані основні критерії (показники). Результати порівняння методів управління змінами за критеріями наведено в таблиці 1.3.

Таблиця 1.3 – Порівняння існуючих методів управління змінами у проєктах

Показник	Метод, заснований на моделі ADKAR PROSCI та ADKAR PROSCI	Метод AIM	Метод, заснований на моделі Бекхарда і Гарріса	Метод, заснований на моделі Вільяма та Бріджа	Метод, заснований на моделі Джона Коттера	Метод, заснований на моделі Коблера Росса	Метод, заснований на моделі Курта Левіна
Простота використання	–	–	+	–	–	+	+
Складність впровадження	–	–	+	–	+	+	+
Час, необхідний на використання	+	+	–	+	+	+	–
Наявність кількісної оцінки змін	–	–	–	–	–	–	–
Гнучкість методу	–	–	+	+	–	+	–
Використання при швидких змінах	–	+	+	–	–	–	–
Можливість проходження сертифікації	+	+	–	+	+	–	–
Можливості проходження тренінгів	+	+	+	+	+	+	+
Наявність інформативного сайту	+	+	+	–	+	–	–
Наявність безкоштовних тренінгів	–	–	+	–	–	+	+
Можливість використання в складних проєктах	+	+	+	–	+	–	–
Вартість тренінгів, у.о.	2000	1790	–	–	6000 (850)	–	–

1.4 Постановка задачі дослідження

Результати досліджень, описаних в даному розділі кваліфікаційної роботи щодо управління змінами в довгострокових проєктах, дозволяють зробити наступні висновки.

По-перше, при реалізації сучасних довгострокових ІТ-проєктів характерним є значна кількість змін, на різних рівнях, що є наслідком особливостей реалізації та управління довгостроковими та сучасними ІТ-проєктами. Значна кількість змін при реалізації таких проєктів відбувається на операційному рівні задач, що робить необхідним якісне управління змінами на цьому рівні.

По-друге, в процесі досліджених методів управління змінами, було виявлено, що в них відсутні підходи для кількісного оцінювання змін на рівні задач. Отже, актуальним є реалізація підходу з можливістю кількісного оцінювання змін.

В роботі були розглянуті актуальні методи засновані на моделях управління змінами.

Методи управління змінами – способи впливу суб'єктів управління змінами на об'єкти змін;

Моделі управління змінами – узагальнені імітаційні схеми процесів та алгоритмів управління змінами, котрі мають усталені характеристики та елементи.

Процес управління змінами складається з двох основних функцій:

- прийняття рішення про те, що необхідно зробити;
- реалізація прийнятих рішень.

Відповідно до проведеного порівняльного аналізу методів управління змінами було обрано метод, заснований на моделі Бекхарда і Гарріса, з метою його адаптації для використання на рівні задач з можливістю кількісної оцінки змін часу.

Об'єктом дослідження в роботі є процес оцінювання змін часу виконання задач при реалізації довгострокових ІТ-проектів.

Метою даної кваліфікаційної роботи є розробка комбінованого методу оцінювання змін часу виконання задач при реалізації довгострокових ІТ-проектів.

Для досягнення поставленої мети у магістерській кваліфікаційній роботі пропонується виконати наступні задачі дослідження:

- проаналізувати особливості реалізації довгострокових ІТ-проектів;
- проаналізувати та узагальнити існуючі методи управління змінами на підприємствах;
- визначити критерії вибору методу управління змінами на підприємствах відповідно до мети його адаптації для використання на рівні задач з можливістю оцінювання змін;
- розробити підхід оцінювання змін часу при реалізації довгострокових проектів.
- розробити комбінований метод оцінювання змін з використанням існуючого методу управління змінами при реалізації ІТ-проектів;
- практично реалізувати запропонований метод на прикладі ІТ-проекту розробки системи створення сайтів;
- експериментально перевірити ефективність розробленого комбінованого методу управління змінами часу на рівні задач.

2 РОЗРОБКА КОМБІНОВАНОГО МЕТОДУ ОЦІНЮВАННЯ ЗМІН ЧАСУ ВИКОНАННЯ ЗАДАЧ ПРИ РЕАЛІЗАЦІЇ ІТ-ПРОЄКТУ

2.1 Опис дескрипторного підходу для оцінювання змін на рівні задач ІТ-проєкту

В результаті аналізу методів, що можуть бути використані з метою управління змінами в довгострокових ІТ-проєктах, було запропоновано розробити дескрипторний підхід з метою оцінювання змін на рівні задач.

Введемо поняття, пов'язані з використанням дескрипторного підходу в ІТ-проєктах: SCRUM, спринт, задача, дескриптор, показник змін.

SCRUM – методологія управління проєктами, яка зазвичай використовується в розробці програмного забезпечення. Вона базується на ітераційному та інкрементальному підходах до розробки продукту, де команда працює у коротких циклах (спринти) [18].

Спринт – це короткий період часу (зазвичай, від двох до чотирьох тижнів), в рамках якого проєктна команда працює над конкретним набором завдань (задач).

Така практика реалізується при використанні Agile-методології розробки програмного забезпечення для підвищення ефективності та прозорості процесу розробки [18].

Задача – окреме завдання з унікальним ідентифікаційним номером, яке виконується в рамках спринту певною проєктною командою для досягнення поставленої мети або результату.

Задача найчастіше оцінюється за складністю та часом виконання. Вона має наступні характеристики: опис, дату створення, пріоритет, оцінку часу виконання, відповідального виконавця, статус виконання, критерії прийняття, дедлайн, коментарі та додаткові матеріали.

Дескриптор – кількісна характеристика задачі, яка надає описову інформацію про задачу в числовому форматі. Дескриптор може описувати

певні аспекти задач, такі як: інформацію про використання певної технології, інформацію про команду, що буде виконувати задачу, та інше.

Показник змін – числовий показник, з використанням якого можна зробити оцінку змін при їх внесенні. Цей показник виступає числовим показником оцінювання змін. Показником змін може виступати час, необхідний для виконання задачі, або різниця запланованого часу та витраченого часу на виконання задачі.

Дескрипторний підхід, який пропонується для оцінювання змін на рівні задач ІТ-проєкту, включає виконання наступних процесів:

- збір та зберігання в базі даних інформації, яка буде описувати характеристики виконуваних задач у вигляді дескрипторів;
- побудова статистичних моделей для пошуку залежності показника змін від дескриптора з метою кількісного оцінювання змін.

Умовою для використання дескрипторного підходу є наявність достатньої кількості інформації про виконувані задачі.

Для довгострокових ІТ-проєктів характерне накопичування інформації щодо виконаних задач в значній кількості, що дає змогу цю інформацію аналізувати з використанням статистичних методів та реалізувати дескрипторний підхід.

При використанні дескрипторного підходу в рамках реалізації довгострокового ІТ-проєкту рекомендується проводити наступну класифікацію дескрипторів за такими показниками:

- технічні дескриптори;
- дескриптори особливостей задачі;
- дескриптори опису команди;
- дескриптори процесу тестування;
- дескриптори доступності веб-сторінки;
- інші.

Кількість показників, за якими проводиться класифікація, може коливатись в залежності від вимог ІТ-проєкту.

Дескриптор може мати бінарне значення: «Так» або «1», або «Істина» чи «Ні» або «0», або «Хибність». Наприклад, дескриптор зміни бібліотеки для тестування модулів для певного програмного компоненту може мати наступні бінарні значення. Якщо в завданні необхідно змінити бібліотеку для тестування компоненту, то значення дескриптору буде 1, якщо ні – 0.

Прикладом дескриптору з простим числовим значенням може бути дескриптор кількості рядків коду, для якого необхідно реалізувати функціональне тестування, або дескриптор кількості помилок в коді, які необхідно виправити під час виконання задачі.

При використанні дескрипторного підходу рекомендується надавати можливість внесення дескрипторів до бази даних всім членам проєктної команди.

Пропонується процес оновлення бази даних дескрипторів проводити в рамках кожного спринту та інформацію про наявність цього процесу додати до вимог готовності задач, які існують у проєкті (DoD, definition of done) [19].

Дескрипторний підхід дає можливість описати велику кількість дескрипторів, які можуть бути використані на різних етапах вирішення задач.

Аналіз показав, що може бути використано попереднє експертне оцінювання впливу деяких дескрипторів на час, необхідний для виконання задачі. Проте використання статистичного моделювання для оцінювання такого впливу може виявити нові взаємозв'язки та надати більш точні результати.

При наявності бази даних дескрипторів та показників змін можна будувати статистичні моделі залежності показників змін від дескрипторів.

Пропонується будувати одномірну модель з урахуванням наступних причин:

- простота розрахунків;
- необхідність меншої кількості даних для розробки моделі;
- відсутність проблеми, пов'язаної з мультиколінеарністю дескрипторів, які пропонуються [20].

Якщо використовується більше однієї незалежної змінної, то існує ризик мультиколінеарності змінних. Ця проблема виникає, якщо змінні сильно корелюють між собою, і може призвести до нестабільності оцінок коефіцієнтів моделі. Одномірна модель є більш стійкою до цих проблем, що покращує надійність і точність оцінок.

З однією змінною (дескриптором) навіть невеликі набори даних можуть бути достатніми для побудови адекватної моделі.

Додаткові змінні збільшують складність моделі та можуть призвести до перенавчання, коли модель добре підходить до навчальних даних, але погано прогнозує нові дані. Одномірна модель є простішою і менш схильною до перенавчання.

Моделі з однією змінною потребують менше обчислень для оцінки коефіцієнтів. Це важливо, якщо обчислювальні ресурси обмежені або потрібні швидкі розрахунки. Розробка та підтримка одномірних моделей простіша і менш схильна до помилок.

Одномірна модель може бути достатньою для виявлення основних тенденцій і залежностей в даних, особливо на початкових етапах дослідження.

2.2 Опис комбінованого методу оцінювання змін часу виконання задач з використанням дескрипторного підходу

Виходячи з аналізу методів управління змінами запропоновано розробити і використовувати комбінований метод, який буде давати можливість здійснювати кількісне оцінювання змін на рівні задач при реалізації довгострокового ІТ-проєкту.

З метою розробки методу оцінювання змін на рівні задач в довгострокових ІТ-проєктах пропонується, як базовий, використати метод,

заснований на моделі Бекхарда і Гарріса [13]. Цей метод найкраще підходить для управління змінами, виходячи з аналізу, проведеному в розділі 2. Зокрема метод, заснований на моделі Бекхарда і Гарріса, пропонується адаптувати до управління змінами на рівні задач проєкту та додати можливість використання в ньому дескрипторного підходу з метою якісного оцінювання змін.

Етапи комбінованого методу оцінювання змін на рівні задач при реалізації довгострокового ІТ-проєкту з використанням дескрипторного підходу на основі методу, заснованого на моделі Бекхарда і Гарріса, зображені на рисунку 2.1.

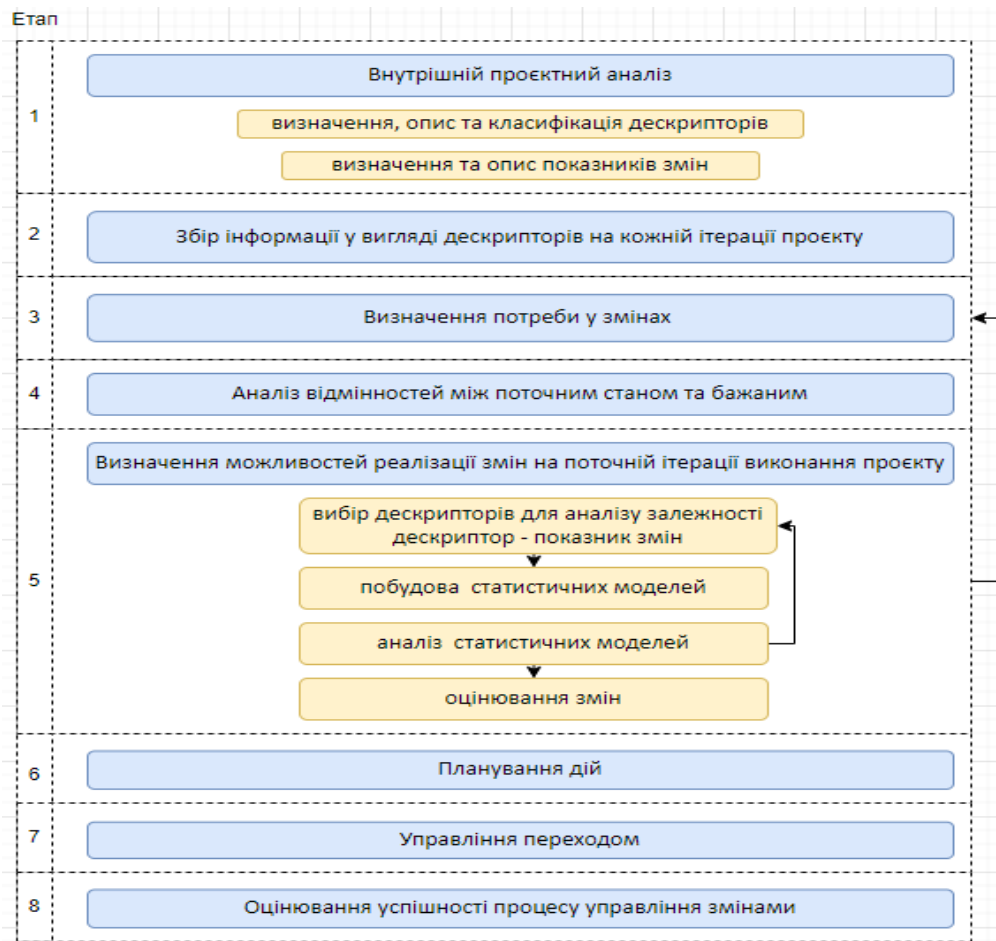


Рисунок 2.1 – Етапи комбінованого методу оцінювання змін на рівні задач при реалізації довгострокового ІТ-проєкту з використанням дескрипторного підходу

На етапі «Внутрішній проєктний аналіз» (етап 1) має бути забезпечене наступне:

- оцінювання поточних процесів та систем управління змінами на рівні задач;
- визначення співробітників, які можуть мати значний опір до змін;
- визначення зовнішніх чинників, які можуть перешкоджати процесу змін;
- аналіз ресурсів, доступних для впровадження змін;
- опис дескрипторів та проведення їх класифікації;
- визначення місця для збереження дескрипторів (бази даних);
- визначення та опис показників змін та місця їх збереження;
- призначення відповідальних за формування бази даних дескрипторів та її оновлення.

Етап «Збір інформації у вигляді дескрипторів на кожній ітерації проєкту» (етап 2) необхідний для формування бази даних дескрипторів. Додавання дескрипторів для поточних задач варто виконувати на кожній ітерації проєкту.

Етап «Визначення потреби у змінах» (етап 3) необхідний для створення підстави для впровадження змін. На даному етапі більшість робіт може бути виконана бізнес-аналітиком. На етапі «Визначення потреби у змінах» має бути забезпечене наступне:

- виявлення конкретних областей, де необхідні зміни на рівні задач;
- проведення оцінювання потреб користувачів та замовників у внесенні змін до задач;
- встановлення цілей та очікуваних результати від впровадження змін.

Наступним є етап «Аналіз відмінностей між поточним станом та бажаним» (етап 4). На даному етапі повинно бути виконано наступне:

- виявлено, які значні відхилення існують між поточним станом аналізованої задачі та її станом після внесення змін;
- проведено оцінювання можливих ризиків та впливу змін в проєкт;
- порівняно технічні характеристики поточного та бажаного стану

проєкту.

Наступним є етап «Визначення можливостей реалізації змін на поточній ітерації виконання проєкту» (етап 5).

На даному етапі має бути проведено наступне:

- вибір дескрипторів для аналізу залежності показника змін від дескриптора;
- побудова статистичних моделей;
- аналіз статистичних моделей;
- оцінювання змін на рівні задач.

Набір дескрипторів можна проаналізувати за середнім значенням дескрипторів, проте даний аналіз може мати значні недоліки. Аналіз даних за допомогою середнього значення (середньої величини) є простим способом описувати центральну тенденцію даних. Проте цей підхід може бути не прогностичним у випадках, коли дані мають велику варіабельність або коли вони розподілені нерівномірно. Особливо у випадках, коли дані мають велику кількість викидів або асиметрію, використання середнього може призвести до неправильного або неповного розуміння даних. При реалізації дескрипторного підходу досліджується залежність між показниками змін (шуканими показниками) та значеннями дескрипторів.

Дану залежність можна представити наступною формулою:

$$y_i = \beta_0 + \beta_1 * x_i, \quad (2.1)$$

де y_i – залежна змінна (показник змін);

x_i – незалежна змінна (дескриптор);

β_0, β_1 – параметри регресії.

Після цього вирішується задача знаходження параметрів регресії.

Існує багато методів для побудови регресійних моделей. Проте, найчастіше використовується метод найменших квадратів (Ordinary Least Squares, OLS).

Перед використанням методу OLS необхідно перевірити виконання передумов використання регресійного аналізу. В результаті такого аналізу можуть бути виявлені викиди, які треба буде опрацювати.

У випадку наявності викидів в даних, які збираються при виконанні задач проєкту, в комбінованому методі, який розробляється, пропонується використовувати регресію Хьюбера, тому що вона має властивості, пов'язані з робастністю [21].

Після побудови статистичних моделей пропонується виконувати перевірку їх працездатності з використанням наступних показників [21]:

- коефіцієнт детермінації (R^2);
- коефіцієнт прогнозування (Q^2);
- коефіцієнт прогнозування, розрахований за процедурою Leave-one-out cross-validation (Q_{LOOCV}^2);
- стандартне відхилення (σ);
- оцінювання значущості статистичної моделі (F – критерій).

Якщо аналіз статистичних моделей надає незадовільні показники, відбувається вибір іншого дескриптору для аналізу та перерахунок статистичних моделей.

В випадку задовільних показників аналізу статистичних моделей відбувається оцінювання змін з використанням побудованої моделі.

При використанні отриманої оцінки робиться припущення про можливість реалізації змін на рівні задачі на поточній ітерації виконання ІТ-проєкту. Якщо оцінка необхідного часу виконання задачі відповідає можливості реалізації змін на поточній ітерації виконання проєкту (спринті), то зміни затверджують. Якщо оцінка не відповідає можливості реалізації змін, то аналізуються результати етапу «Визначення потреби в змінах» (етап 3) та робиться аналіз можливості залучення додаткових людських ресурсів з інших команд або можливість понаднормової роботи учасників поточної команди. У випадку, коли робиться висновок про те, що зміни не можливо реалізувати в поточній ітерації проєкту, створюється нова задача та

планується на наступні ітерації виконання проєкту.

На етапі «Планування дій» (етап 6) реалізується план змін. На даному етапі має бути виконане наступне:

- визначено ключових учасників процесу змін та обов'язки кожного, хто вносить зміни;
- розроблено конкретний план впровадження змін, включаючи ресурси та терміни;
- визначено послідовність кроків для впровадження змін;
- підготовлено комунікаційний план для інформування замовників про зміни та їх вплив на проєкт.

Наступним є етап «Управління переходом» (етап 7). На даному етапі має бути виконане наступне:

- реалізація плану змін для задачі;
- моніторинг впливу змін на проєкт та вчасне виявлення проблем;
- забезпечення підтримки для користувачів під час переходу до нового стану проєкту.

Останнім етапом є «Оцінювання успішності процесу управління змінами» (етап 8). Його рекомендовано проводити раз на декілька ітерацій виконання проєкту або на запит.

З метою оцінювання успішності процесу управління змінами в комбінованому методі пропонується використовувати наступні показники:

- відсоток скасованих змін часу Ch_{Cp} , у зв'язку з неможливістю їх реалізації, за виключенням змін, що стали непотрібними за певний період;
- відсоток змін часу, що були прийняті замовником та затверджені як успішно реалізовані за певний період часу, тобто Ch_{Sp} ;
- відхилення між часом, який був витрачений на виконання робіт, і оціненим часом за певний період часу, тобто δCh_T ;
- відсоток змін, які були завершені вчасно за певний період часу Ch_{ITp} .

Відсоток скасованих змін за певний період часу Ch_{Cp} у зв'язку з неможливістю їх реалізації, за виключенням змін, що стали неактуальними, можна розрахувати за формулою:

$$Ch_{Cp} = \frac{Ch_C}{Ch} * 100\%, \quad (2.2)$$

де Ch – загальна кількість змін за певний період часу;

Ch_C – кількість скасованих змін за певний період часу у зв'язку з неможливістю їх реалізації, за виключенням змін, що стали неактуальними.

Більше значення Ch_{Cp} свідчить про гірше сплановані зміни.

Відсоток змін, що були прийняті замовником та затверджені як успішно реалізовані, за певний період часу Ch_{Sp} можна розрахувати за формулою:

$$Ch_{Sp} = \frac{Ch_S}{Ch} * 100\%, \quad (2.3)$$

де Ch_S – кількість змін, що були прийняті замовником та затверджені як успішно реалізовані за певний період часу.

Більше значення Ch_{Sp} свідчить про кращий процес управління змінами.

Відхилення між часом, який був витрачений на виконання робіт (задач) довгострокового ІТ-проєкту, і оціненим часом за певний період часу δCh_T , можна розрахувати за формулою:

$$\delta Ch_T = \frac{\sum_{i=0} (t_i^{(p)} - t_i^{(a)})}{i * t}, \quad (2.4)$$

де $t_i^{(p)}$ – час, запланований на виконання зміни, дні;

$t_i^{(a)}$ – час, витрачений на виконання зміни, дні;

i – кількість змін за період часу t ;

t – період часу для оцінки, дні.

Показник δCh_T вказує на те, чи виконуються зміни вчасно і дотримуються плану змін. Чим менший показник, тим краще організоване управління змінами.

Відсоток змін, які були завершені вчасно за певний період часу Ch_{iTp} , можна розрахувати за формулою:

$$Ch_{iTp} = \frac{Ch_{iT}}{Ch} * 100\%, \quad (2.5)$$

де Ch_{iT} – кількість змін, які були завершені вчасно за певний період часу.

Більше значення Ch_{iTp} свідчить про кращий процес управління змінами та слідування запланованому графіку.

Розроблений комбінований метод управління змінами має наступні обмеження:

- тривалість проєкту має давати змогу збирати необхідну кількість інформації для аналізу;
- необхідність зберігання інформації для аналізу;
- моделі, побудовані з використанням даного підходу, мають значення тільки для конкретних проєктів, на основі яких вони були зібрані дані для побудови моделей (проте можливі виключення).

Комбінований метод має наступні переваги:

- можливість кількісної оцінки змін;
- можливість виявлення неочевидних взаємозв'язків між дескрипторами та показниками змін проєкту.

2.3 Висновки з розробки комбінованого методу оцінювання змін часу виконання задач при реалізації довгострокового ІТ-проєкту

Виходячи з аналізу існуючих методів управління змінами, було запропоновано розробити комбінований метод управління змінами на рівні задач з використанням дескрипторного підходу на основі методу, заснованому на моделі Бекхарда і Гарріса.

В результаті розробки комбінованого методу можна зробити наступні висновки. По-перше, метод заснований на моделі Бекхарда і Гарріса, було адаптовано для використання на рівні задач, так як стандартний метод використовується для змін на рівні організацій. По-друге, було запропоновано дескрипторний підхід з метою оцінювання змін на рівні задач. По-третє, з використанням дескрипторного підходу було розроблено комбінований метод управління змінами на рівні задач з можливістю числової оцінки змін та описано етапи його виконання.

З метою оцінки змін часу виконання задач на етапі «Визначення можливостей реалізації змін на поточній ітерації виконання проекту» пропонується використовувати статистичні моделі, побудовані з використанням дескрипторного підходу. В розробленому методі з метою побудови регресійних моделей пропонується використовувати метод найменших квадратів (OLS) та робастний метод (регресію Хьюбера).

Отже, в розділі було розроблено комбінований метод управління змінами на рівні задач при реалізації довгострокового ІТ-проекту з використанням дескрипторного підходу на основі методу, заснованому на моделі Бекхарда і Гарріса.

В даній роботі розглянуто зміни часу, проте комбінований метод надає змогу працювати також з іншими показниками змін, актуальними для певного проекту в певних умовах.

3 ДОСЛІДЖЕННЯ ОСОБЛИВОСТЕЙ РЕАЛІЗАЦІЇ КОМБІНОВАНОГО МЕТОДУ ОЦІНЮВАННЯ ЗМІН ЧАСУ ВИКОНАННЯ ЗАДАЧ ПРИ РЕАЛІЗАЦІЇ ІТ-ПРОЄКТУ

3.1 Сценарії використання комбінованого методу оцінювання змін

3.1.1 Основні етапи реалізації комбінованого методу

В розділі 2 кваліфікаційної роботи запропонований та описаний комбінований метод оцінювання змін часу, необхідного для виконання задач за результатами реалізації довгострокового ІТ-проєкту.

В даному розділі розглядаються на аналізуються особливості реалізації кожного етапу розробленого комбінованого методу.

Відповідно до проведених розробок етапами комбінованого методу оцінювання змін часу на рівні задач, виконуваних в рамках ІТ-проєкту, є наступні:

- «Внутрішній проєктний аналіз» (етап 1);
- «Збір інформації у вигляді дескрипторів на кожній ітерації проєкту» (етап 2);
- «Визначення потреби в змінах» (етап 3);
- «Аналіз відмінностей між поточним станом та бажаним» (етап 4);
- «Визначення можливостей реалізації змін на поточній ітерації виконання проєкту» (етап 5);
- «Планування дій» (етап 6);
- «Управління переходом» (етап 7);
- «Оцінювання успішності процесу управління змінами» (етап 8).

Відповідальним за виконання етапів комбінованого методу управління змінами пропонується призначити менеджера проєкту.

В деяких випадках відповідальним може бути майстер-SCRUM або технічний лідер команди проєкту.

У реалізації етапів розробленого комбінованого методу оцінювання змін часу, необхідного для виконання задач за результатами реалізації довгострокового ІТ-проєкту беруть участь наступні члени проєктної команди: бізнес-аналітик, технічний лідер команди, архітектор системи та члени проєктної команди.

3.1.2 Реалізація етапу «Внутрішній проєктний аналіз»

Етап «Внутрішній проєктний аналіз», який є першим етапом комбінованого методу, проводиться з метою ініціалізації необхідних процесів для управління змінами. Відповідно до технології використання розробленого комбінованого методу оцінювання змін часу виконання задач проєкту перший етап відбувається один раз. Проте, повторення першого етапу можливе тільки за запитом проєктної команди або менеджера проєкту.

Спочатку відповідальний за виконання етапів комбінованого методу має проаналізувати наявність процесів управління змінами часу на рівні задач при реалізації проєкту. Після цього проводиться їх оцінювання.

Відповідно до технології використання дескрипторного методу проводиться опитування членів проєктної команди щодо їх сприйняття поточних процесів управління змінами. Зокрема, проводиться опитування щодо проблем, які мають місце на поточний момент в управлінні змінами на рівні задач.

Відповідальний за виконання етапів методу аналізує доступні ресурси для впровадження змін.

На цьому етапі проводиться аналіз можливих дескрипторів при реалізації ІТ-проєкту. Подібний аналіз необхідно проводити за участі всіх учасників проєктної команди: архітектора системи, бізнес-аналітика, розробника серверної частини, розробника користувацької частини, менеджера з тестування, автоматизатора, дизайнера та інших. Для цього можна провести окремо з кожною частиною команди зустріч та визначити можливі дескриптори. На зустрічі проводяться необхідні для виконання комбінованого методу дії:

- визначаються актуальні дескриптори на рівні задач при реалізації

проєкту;

- виконується класифікація дескрипторів;
- визначаються показники змін;
- визначається місце для збереження дескрипторів та показників змін (база даних дескрипторів);
- назначаються відповідальні за формування бази даних дескрипторів та її заповнення.

З метою визначення дескрипторів розглядаються актуальні проблеми для поточного проєкту.

Якщо для певного проєкту для частини задач є актуальним додавання користувацької аналітики, тоді можливо виокремити підклас дескрипторів, пов'язаних з користувацькою аналітикою. Під час обговорення з командою визначаються дескриптори, які можна віднести до цього підкласу.

Відповідальним за формування бази дескрипторів потрібно обирати співробітника, який змінює статус на «Готово» в системах для керування проєктами та відстеження задач, наприклад Jira [22]. Зазвичай при реалізації IT-проєктів цю дію виконують менеджери з тестування після виконання всіх вимог, наведених в списку критеріїв готовності задач (DoD), який існує при виконанні проєкті. Тому, відповідно до технології виконання комбінованого методу, менеджерів з тестування назначають відповідальними за маніпуляції з базою даних дескрипторів. У випадку їх відсутності в проєктній команді дану роботу виконує інший співробітник зі схожими обов'язками.

Для формування бази даних дескрипторів можна використовувати можливості програмного продукту Microsoft Excel або створити власний застосунок. Наприклад, також можна безкоштовно використати наступні системи управління базами даних: Firebase [23], MongoDB [24] або PostgreSQL [25]. Безкоштовні тарифні плани мають наступні обмеження: на обсяг збережених даних, на пропускну здатність, на кількість запитів на сервер та на швидкість відповіді сервера. Проте цих тарифів має бути достатньо для використання технологій реалізації комплексного методу

оцінювання змін часу виконання задач.

Приклад визначених дескрипторів при виконанні ІТ-проекту та їх значення наведено на рисунку 3.1.

	1	2	3	4	5	6	7	8
1	Задача номер	1	2	3	4	5	6	7
2	AC	80	80	20	35	42	76	12
3	time, days actual	20	34	10	8	15	14	5
4	sprints	3	4	1	1	1	2	1
5	team, ppl	7	7	7	7	7	7	7
6	vacation days, holidays	4	3	0	7	7	2	0
7	UX AC	30	65	10	15	12	26	12
8	BE AC	50	15	10	20	30	50	0
9	time, days planned	15	23	8	7	12	9	5
10	vacation days, holidays (PPL who worked on task)	0	0	0	0	0	0	0
11	ACActual	91	96	25	37	47	85	12
12	ACDelta	11	16	5	2	5	9	0
13	ppl	3	3	3	3	3	3	3
14								
15	ppl reserv							
16	reserv days w							
17	days delta	5	11	2	1	3	5	0

Рисунок 3.1 – Приклад визначених дескрипторів при виконанні ІТ-проекту

3.1.3 Реалізація етапу «Збір інформації у вигляді дескрипторів на кожній ітерації проекту»

Другим етапом комбінованого методу виступає «Збір інформації у вигляді дескрипторів на кожній ітерації проекту», який є циклічним процесом та має виконуватись кожну ітерацію проекту (спринт).

Другий етап включає в себе внесення інформації в базу дескрипторів, які є частиною критеріїв готовності задач (DoD задач).

Відповідно до технології виконання розробленого комбінованого методу відповідальним за виконання даного етапу призначають менеджерів з тестування.

При зміні статусу задачі необхідно додавати коментар на сторінці

задачі із значеннями вже описаних дескрипторів для поточної задачі або пропозицією нових дескрипторів. Наприклад, над задачею працював розробник клієнтської частини та виявив нову проблему з певною бібліотекою, яка вирішувалась під час виконання задачі. Розробник може запропонувати дескриптор з інформацією про дану проблему у вигляді коментарю при зміні статусу в системах для керування проєктами та відстеження задач з «В розробці» на «В тестуванні».

Варто відмітити, що під час виконання даного етапу можуть бути виявлені нові дескриптори для задач. В цьому випадку відповідальний за виконання етапу комунікує з відповідальним за виконання методу щодо розширення бази дескрипторів. Якщо новий дескриптор узгоджують, його додають до бази даних.

3.1.4 Реалізація етапу «Визначення потреби в змінах»

Третім етапом комбінованого методу виступає «Визначення потреби в змінах». Даний етап має місце, коли після початку роботи команди над задачею з'являється необхідність зробити в ній зміни. Зміни можуть бути пов'язані з різними причинами, наприклад: оновлення інтерфейсу у зв'язку з оновленням у конкурентів, виявлення певних взаємозв'язків з технічними аспектами, які змінюють складність задачі, та інше.

Більшість робіт на даному етапі можуть бути реалізовані бізнес-аналітиком. Він має провести оцінку потреб, виявити конкретні вимоги до задач та встановити очікувані цілі.

На цьому етапі необхідно встановити пріоритет внесення змін. Якщо пріоритет низький, зміна може бути оформлена як задача для наступних спринтів. Якщо вирішено вносити зміни, то відбувається перехід до четвертого етапу.

3.1.5 Реалізація етапу «Аналіз відмінностей між поточним станом та бажаним»

Четвертим етапом комбінованого методу виступає «Аналіз відмінностей між поточним станом та бажаним». Роботи на даному етапі можуть виконуватись наступними членами команди: бізнес-аналітиком, архітектором системи, технічним лідером команди. Зокрема, за потребою, до робіт на четвертому етапі можуть бути залучені всі учасники проєктної команди. На даному етапі бізнес-аналітик має визначити відмінності між минулим та бажаним станами задачі та детально їх описати.

Зокрема, необхідно переконатись в тому, що кожний учасник проєктної команди має повноцінне розуміння того, що має бути змінено.

На даному етапі необхідно провести зустріч для бізнес-аналітика та проєктної команди. На даній зустрічі потрібно зробити наступне:

- презентувати новий стан задачі;
- зробити попередню оцінку можливих ризиків на виконання проєкту;
- порівняти нові технологічні характеристики задачі та минулі;
- зробити попереднє оцінювання часу, необхідного на реалізацію задачі, з використанням технології оцінювання методом покеру [26] або інших підходів.

Якщо команда не впевнена в тому, що зміни можливо реалізувати на поточній ітерації виконання проєкту повноцінно, вона може запросити проведення п'ятого етапу. Якщо команда впевнена в тому, що зміни можливо реалізувати в поточному спринті, то відбувається перехід на шостий етап.

У випадку, якщо після аналізу наявних ресурсів робиться висновок про неможливість реалізації змін в поточному спринті, створюється задача для виконання у майбутньому спринті.

За умови високого пріоритету змін зі сторони замовника робиться аналіз можливості залучення до виконання змін додаткових ресурсів.

3.1.6 Реалізація етапу «Визначення можливостей реалізації змін на поточній ітерації виконання проєкту»

П'ятим етапом комбінованого методу виступає етап «Визначення можливостей реалізації змін на поточній ітерації виконання проєкту».

Цей етап дозволяє надати команді кількісну оцінку потенційних змін, що допомагає робити обґрунтоване рішення щодо їх впровадження на поточній ітерації проєкту.

Використання числових значень дозволяє об'єктивно порівняти різні альтернативи та визначити їх вплив на поточний спринт. Однак, ці оцінки є лише рекомендаціями, і кінцеве рішення щодо внесення змін повинно базуватися на всебічному аналізі та розумінні потреб проєкту.

Даний етап містить наступні підетапи:

- «Вибір дескрипторів для аналізу залежності показника змін від дескриптора» (підетап 1);
- «Побудова статистичних моделей» (підетап 2);
- «Аналіз статистичних моделей» (підетап 3);
- «Оцінювання змін» (підетап 4).

В рамках даного етапу відповідальний за реалізацію комбінованого методу може залучати до роботи лідера проєктної команди, архітектора системи та інших учасників проєктної команди для консультацій та отримання експертної оцінки. Це сприяє більш об'єктивному аналізу та прийняттю рішень щодо реалізації змін на даному етапі виконання проєкту.

Схематичне зображення реалізації п'ятого етапу «Визначення можливостей реалізації змін на поточній ітерації виконання проєкту» наведено на рисунку 3.2.

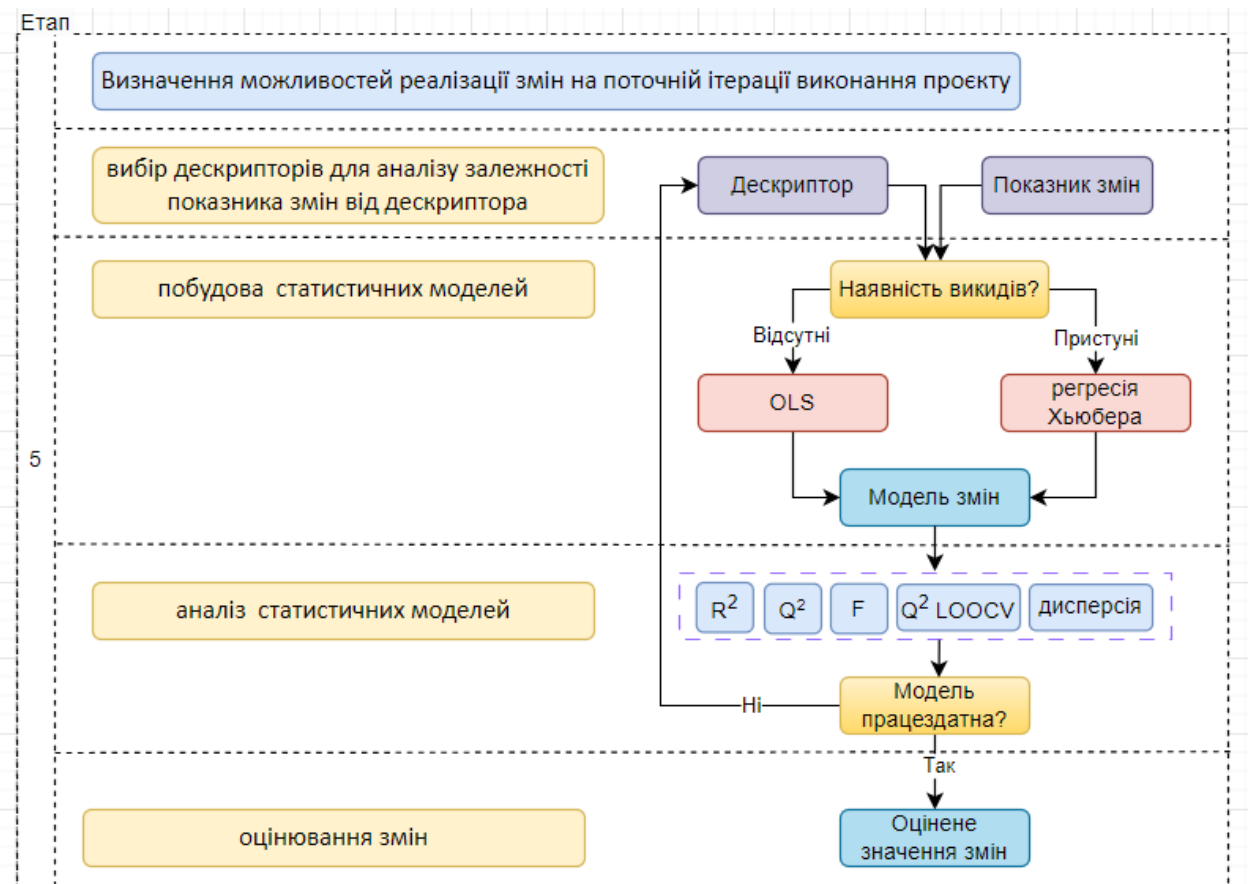


Рисунок 3.2 – Схема реалізації етапу «Визначення можливостей реалізації змін на поточній ітерації виконання проєкту»

На підетапі «Вибір дескрипторів для аналізу залежності показника змін від дескриптора» визначається дескриптор та показник змін для розрахунків.

Показником змін пропонується обирати час або різницю між запланованим часом виконання задачі та реально витраченим часом на виконання задачі.

Проте комбінований метод не ставить обмежень та надає можливість обирати інші показники змін, які будуть актуальними для певних умов.

З метою вибору дескрипторів для розрахунку пропонуються наступні підходи:

- консультація з командою розробки;

– розрахунок методом OLS залежності показника змін від дескриптора та розрахунок коефіцієнта детермінації з метою визначення дескрипторів, з якими коефіцієнт детермінації буде вищим.

Наприклад, команда розглядає задачу під час реалізації якої необхідно реалізувати можливість використання компонента верхньої частини сайту з головним меню (header) [27] з метою зручної навігації сторінкою та можливості налаштування посилань в верхній частині сайту з головним меню сторінки.

На рисунках 3.3–3.8 наведений дизайн інтерфейсу, що має бути створений.



Рисунок 3.3 – Дизайн інтерфейсу для великих екранів



Рисунок 3.4 – Дизайн інтерфейсу для малих екранів (мобільні пристрої)



Рисунок 3.5 – Дизайн інтерфейсу на малих екранах (мобільні пристрої)
(відкрите меню)

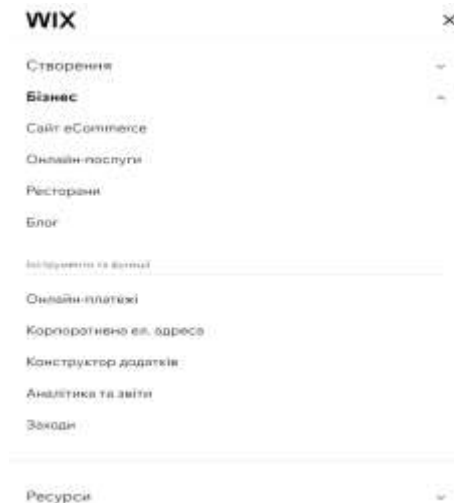


Рисунок 3.6 – Дизайн інтерфейсу на малих екранах (мобільні пристрої)
(відкрите підменю)

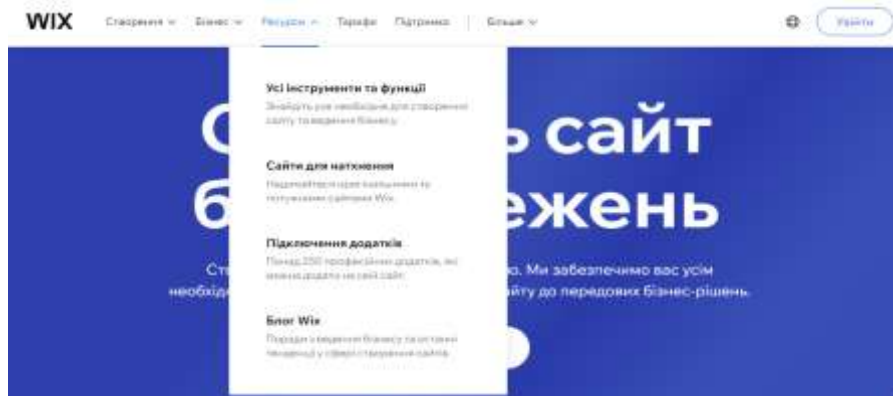


Рисунок 3.7 – Дизайн інтерфейсу на великих екранах, відкрите підменю



Рисунок 3.8 – Дизайн інтерфейсу на великих екранах, відкрите підменю вибору мови

Окрім дизайнів, наведених на рисунках 3.3 – 3.8, в описі задачі містяться посилання на дизайни загальних компонентів, таких, як посилання на кнопки, іконки. За даними посиланнями вказані дизайни різних станів цих компонентів: стан наведення мишки, стан фокусу, стан натискання та інші.

З аналізованих дизайнів розробники мають змогу відокремити необхідні параметри та використати їх.

Також для кожної задачі додається посилання на характеристики задачі, які представлені в таблиці 3.1.

Таблиця 3.1 – Характеристики задачі

Номер	Характеристика задачі
1	2
1	Має бути створений компонент на серверній стороні та надана можливість додавати його на сторінки
2	На всіх сторінках має бути можливість додавати компонент верхньої частини сайту з головним меню
3	Налаштування посилань мають бути на рівні кожної сторінки
4	Має бути реалізована вимога повторного використання шаблонів

	посилань
5	Має бути реалізована можливість використання текстів відповідно до мови сторінки, на якій доданий компонент верхньої частини сайту. У випадку, якщо певного перекладу не існує, має використовуватись переклад англійською мовою
6	Доступ до налаштування шалонів посилань та зміни поточних посилань мають бути у користувачів з відповідним рівнем доступу
7	Має бути додана перевірка на актуальність введених посилань. У випадку, коли посилання веде на неіснуючу сторінку, воно має бути виділеним червоним кольором на момент введення контенту та має з'являтися повідомлення про помилку. Має бути реалізована можливість додавати типи посилань відповідно до існуючої системи
8	Має бути реалізована вимога створення розділів посилань та підрозділів
9	Має бути реалізована можливість завантаження зображення для логотипу сторінки
10	На великих екранах компонент верхньої частини сайту має містити логотип, посилання, кнопку вибору мови та кнопку «Увійти» (див. рисунок 3.3)
11	Кнопка «Увійти» має бути прихована, якщо користувач авторизований
12	Глобальні компоненти (посилання, кнопки, іконки) повинні мати стани відповідні глобальним вимогам
13	При натисканні на підменю на великих екранах має відкриватись меню нижчого рівня (див. рисунок 3.6). При повторному натисненні або при натисненні за межами меню нижчого рівня воно має закриватися. Підменю має містити в такому випадку іконку «стрілочка» і вона має змінювати вид відповідно дизайнам графічного інтерфейсу

Кінець таблиці 3.1

1	2
14	При натисненні на кнопку «Увійти» має відбуватися перехід на сторінку авторизації (див. рисунок 3.6)
15	Якщо в посилань є тип «більше», вони відокремлюються від основних посилань роздільною лінією (відповідно до дизайнів мобільних та не мобільних пристроїв)
16	Якщо в посилань немає підменю, в нього має бути відсутня іконка «стрілочка». При натисканні на таке посилання воно має одразу переводити користувача на відповідну сторінку.
17	При натисканні на кнопку вибору мов має відкритись меню вибору мов (див. рисунок 3.7). Поточна мова повинна мати активний стан та бути синього кольору (відповідно до дизайну). Натиск на кнопку «закрити» повинно закривати це підменю
18	При натисканні на кнопку будь-якої мови, відмінної від поточної, треба переадресувати користувача на поточну сторінку обраної мови
19	На маленьких екранах мобільних пристроїв вид веб-сторінки має відповідати дизайнам та містити іконку меню (див. рисунок 3.4)
20	На маленьких екранах натиск на іконку меню, має відкривати підменю відповідно до дизайнів (див. рисунок 3.5). Пункти меню, які мають підпункти, повинні містити іконку «стрілочка»
21	На маленьких екранах натиск на кнопку іконки меню, повинно відкривати підменю відповідно до дизайнів (див. рисунок 3.5)
22	Дизайн інтерфейсу відкритого підменю вибору мови має відповідати графічним дизайнам (див. рисунок 3.8)
23	При натисканні на кнопку «мова» в меню для мобільних застосунків список мов повинен відкриватись як підменю
24	Відкривання підпунктів має супроводжуватись прокручуванням сторінки (скролінг) таким чином, щоб кнопка відкривання підменю була зверху сторінки
25	Кнопка «Увійти» на мобільних пристроях має бути завжди прикріплена знизу меню, коли воно відкрите (див. рисунок 3.5)
26	При переході за посиланням має збиратись аналітика натиснення на посилання відповідно до загальних вимог системи. Аналітика повинна містити інформацію про текст послання, саме посилання та сторінку, на якій натиск відбувався
27	Якщо компонент знаходиться на сторінці з мовою «Іврит» або іншими мовами, де текст має бути прочитаний з права наліво, контент має бути віддзеркаленим
28	Компонент не повинен мати помилки з доступністю рівнів A та AA [28], [29]
29	Компонент повинен створюватись на стороні сервера (SSR, Server-Side Rendering) [30]

Ці характеристики мають бути реалізовані під час виконання задачі, вони описуються бізнес-аналітиком після обговорень з дизайнерами, замовниками та архітекторами застосунку.

В ході створення множини характеристик бізнес-аналітик надає кожній характеристиці ідентифікатор та поділяє її на ті, що мають бути реалізовані на серверній частині системи, та ті, що мають бути реалізовані на клієнтській частині системи.

На підетапі «Побудова статистичних моделей» необхідно побудувати статистичні моделі залежності показника змін від дескриптора.

Спочатку необхідно провести перевірку на виконання передумов використання регресійного аналізу.

Перша умова передбачає, що середнє значення помилок моделі повинно бути нульовим. Це означає, що модель правильно передбачає вплив незалежних факторів на змінну, яка аналізується. Якщо ця умова не виконується, це може означати, що є помилка у виборі моделі.

Друга умова стверджує, що різноманітність помилок моделі повинна залишатися сталою. Це означає, що помилки рівномірно розподілені навколо нуля. Якщо різноманітність помилок змінюється, це може бути знаком того, що існують інші фактори, які впливають на результат, але не були враховані в моделі.

Третя умова стверджує, що помилки моделі повинні бути незалежними одна від одної. Це означає, що помилки не пов'язані між собою у часі або за якимось іншим критерієм. Якщо ця умова порушується, це може вказувати на наявність складніших взаємозв'язків у даних, які не враховані в моделі.

Четверта умова стверджує, що всі використані фактори у моделі повинні бути незалежними один від одного.

Це означає, що жоден фактор не повинен бути пов'язаний з іншими, що допомагає уникнути проблеми мультиколінеарності, коли фактори взаємозалежні і можуть спотворити оцінку параметрів моделі.

Під час перевірки можливості використання методу найменших квадратів важливо також робити перевірку на наявність викидів.

Викиди (outliers) – це аномальні або несподівані значення в наборі даних, які суттєво відрізняються від інших спостережень. Вони можуть виникати з різних причин, таких як помилки вимірювання, випадкові аномалії, чи навіть вказувати на особливості в даних.

Відповідно до технології використання розробленого комбінованого методу рекомендується робити аналіз дескрипторів на присутність викидів шляхом побудови розподілу Гауса та перевірки методом трьох сигм [31].

Згідно з цим методом, якщо значення відхиляється від середнього більше ніж на 3 стандартних відхилення, воно вважається викидом. Отже, спочатку будується нормальний розподіл даних та визначається середнє значення та стандартне відхилення. Потім, якщо будь-яке значення виявляється більшим або меншим, ніж середнє значення (плюс або мінус три стандартні відхилення) воно вважаються викидом.

У випадку відсутності викидів даних для розрахунку залежності пропонується використовувати метод найменших квадратів.

Метод найменших квадратів (Ordinary Least Squares, OLS) є одним з основних методів в аналізі даних, що використовується для апроксимації функцій. Основна ідея полягає в тому, щоб знайти функцію, яка найкращим чином підходить до даних, шляхом мінімізації квадратів відстаней між функцією і точками даних [21].

При використанні OLS можуть виникати розрахункові проблеми за наступних умов:

- надлишковість дескрипторного набору;
- наявність грубих промахів (викидів).

Надлишковість дескрипторного набору виникає, коли є багато різних описових ознак, і потрібно вибрати лише декілька з них, які найкраще підходять для побудови моделі.

Недолік методу найменших квадратів, пов'язаний із грубими промахами (відсутність робастності), можна умовно зобразити на прикладі, наведеному на рисунку 3.9.

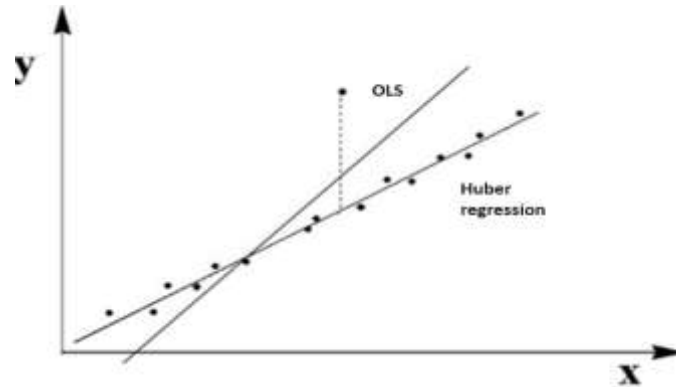


Рисунок 3.9 – Приклад неробастної поведінки методу OLS

На даному зображенні (див. рисунок 3.9) є певний набір точок, які майже проходять по лінії, та один грубий промах. Лінія OLS буде відхилитися в бік точки, що випадає. Дана поведінка лінії називається неробастною.

Найпростіший підхід реагування на викиди – це вилучення викидів та їх ігнорування. Вилучення викидів – це простий спосіб, коли необхідно вилучити аномальні значення з набору даних перед аналізом. Це може призвести до поліпшення точності аналізу для тих випадків, коли викиди є помилковими. Проте, іноді викиди можуть бути інформативними та важливими значеннями. В цьому випадку просте ігнорування їх може призвести до втрати важливої інформації [32].

До ризиків ігнорування викидів можна віднести: спотворення результатів, втрату інформації та отримання некоректних моделей.

Робота з викидами при вирішенні задачі оцінювання змін є важливою складовою для забезпечення точності та адекватності результатів.

Аналіз роботи з викидами показав, що вони можуть бути виявлені та оцінені за допомогою наступних методів [33]:

- методу на основі стандартного відхилення;

- методу на основі розподілу даних;
- методу на основі відстані до середнього;
- методу на основі моделей машинного навчання;
- методу з використанням валідації моделі;
- методу з врівноваженням вагів;
- методу ручного виявлення викидів в ході експертного аналізу.

Відповідно до технології використання комбінованого методу, що розробляється в роботі, у випадку присутності викидів для розрахунків моделі пропонується використовувати регресію Хьюбера.

Регресія Хьюбера (Huber Regression) – це метод регресії, призначений для покращення стійкості моделі до викидів даних [33].

Метод використовує оптимізацію методом зважених найменших квадратів. Його можна віднести до методів на основі відстані до середнього. Замість мінімізації суми квадратів втрат, як у OLS, регресія Хьюбера мінімізує зважену суму втрат, де ваги залежать від величини втрат. Це дозволяє знизити вплив викидів на оцінку параметрів моделі. Функція втрат Хьюбера розраховується за формулою:

$$L_{\delta}(r) = \begin{cases} \frac{1}{2}r^2 \\ \delta \left(\left| r - \frac{1}{2}\delta \right| \right) \end{cases}, \quad (3.1)$$

де r – відхилення між спостереженим значенням та передбаченим значенням (залишкова помилка);

δ – параметр, який контролює «поріг» між квадратичною та лінійною частинами функції втрат.

Функція втрат Хьюбера має параметр δ (delta), який визначає, наскільки великою буде залишкова помилка, щоб її вважати «великою». Цей параметр також вказує на те, на якій відстані від середнього починати застосовувати лінійну функцію втрат замість квадратичної. У межах цього

параметра функція втрат зберігає квадратичну форму, що допомагає зберегти чутливість до великих відхилень. Чим більше значення δ , тим менше вплив великих відхилень на результат, що робить метод регресії більш стійким до викидів у даних [34].

Регресія Хьюбера виявляється кращою за метод найменших квадратів (OLS) у випадках, коли є викиди в даних, які можуть впливати на точність моделі. Вона надає більш стійкі оцінки параметрів моделі, оскільки зменшує вплив великих відхилень. Проте, обробка регресії Хьюбера може бути більш складною через необхідність налаштування параметра δ та додаткові обчислення, які вимагають ітераційного підходу.

На підетапі «Аналіз статистичних моделей» необхідно перевірити працездатність моделей змін, отриманих на попередньому підетапі.

Для перевірки працездатності моделей оцінювання змін використовуються наступні показники: R^2 , Q^2 , Q_{LOOCV}^2 , σ , F – критерій.

Коефіцієнт детермінації R^2 є статистичним показником, який використовується для оцінювання того, наскільки добре модель регресії описує дані. Коефіцієнт детермінації визначає, який відсоток варіації у відгуку може бути пояснений моделлю. В ідеальному випадку значення R^2 дорівнює 1.

Коефіцієнт детермінації розраховується з використанням формул:

$$R^2 = 1 - \frac{SS_{res}^{(calc)}}{SS_{tot}}, \quad (3.2)$$

$$SS_{res}^{(calc)} = \sum_{i=1}^n (y_i - \hat{y}_i^{(calc)})^2, \quad (3.3)$$

$$SS_{tot} = \sum (y_i - \bar{y})^2, \quad (3.4)$$

де $SS_{res}^{(calc)}$ – сума квадратів залишкових (регресійних) помилок;

$\hat{y}_i^{(calc)}$ – значення, знайдені за рівнянням регресії;

y_i – експериментальні значення;

\bar{y} – середнє значення експериментальних даних;

SS_{tot} – загальна сума квадратів відхилень від середнього значення залежної змінної [35].

Коефіцієнт прогнозування Q^2 використовується для оцінки якості прогностичної здатності моделі. Ідеальне значення Q^2 дорівнює 1 та вказує на те, що модель дуже точно передбачає спостережені дані.

Коефіцієнт прогнозування розраховується з використанням формул:

$$Q^2 = 1 - \frac{SS_{\text{res}}^{(\text{pred})}}{SS_{\text{tot}}}, \quad (3.5)$$

$$SS_{\text{pred}}^{(\text{calc})} = \sum_{i=1}^n (y_i - \hat{y}_i^{(\text{pred})})^2, \quad (3.6)$$

де $SS_{\text{res}}^{(\text{pred})}$ – сума квадратів відхилень між прогнозованими та спостереженими значеннями залежної змінної;

$\hat{y}_i^{(\text{pred})}$ – значення, знайдені за рівнянням регресії;

y_i – експериментальні значення;

\bar{y} – середнє значення експериментальних значень [35].

Відмінність між R^2 та Q^2 полягає у тому, що R^2 використовується для оцінки якості адаптації моделі до даних, тоді як Q^2 використовується для оцінки її передбачувальної здатності.

Коефіцієнт прогнозування Q_{LOOCV}^2 , розрахований за процедурою Leave-one-out cross-validation (LOOCV), дозволяє оцінити, наскільки добре модель узгоджується з новими даними, які не використовувалися під час навчання моделі. Процедура LOOCV заснована на викиді однієї точки з вибірки, знаходження рівняння регресії на основі інших (N-1) точок та знаходженні теоретичного значення залежної змінної для цієї «вилученої» точки. Процес розрахунку Q_{LOOCV}^2 складається з наступних двох кроків:

- обчислення прогнозів моделі для кожного спостереження;
- розрахунок Q^2 на основі прогнозів.

Обчислення прогнозів моделі для кожного спостереження складається з наступних фаз:

- видалення по одному спостереженню з даних;
- побудова моделі на основі залишкових даних;
- прогнозування значення для вилученого спостереження.

Розрахунок Q^2 на основі прогнозів складається з наступних фаз:

- обчислення суми квадратів різниць між спостереженнями і прогнозами;
- обчислення загальної суми квадратів різниць між спостереженнями і їх середнім значенням [21].

Процедура повторюється для всіх точок навчальної вибірки.

Стандартне відхилення σ є мірою розсіювання або розбіжності даних від їх середнього значення. В регресійному аналізі стандартне відхилення використовується для вимірювання точності або стійкості прогнозних моделей. У разі ідеального прогнозу показник буде дорівнювати 0, що вказуватиме на те, що модель ідеально прогнозує або адекватно відтворює фактичні дані. Для розрахунку стандартного відхилення використовується формула:

$$\sigma = \frac{\sqrt{\sum_i (y_i^{(\text{exp})} - y_i^{(\text{calc})})^2}}{N - k - 1}, \quad (3.7)$$

де N – загальна кількість спостережень;

$y_i^{(\text{calc})}$ – розраховане значення змінної y для i -го спостереження;

$y_i^{(\text{exp})}$ – фактичне значення змінної y для i -го спостереження;

k – кількість параметрів моделі (включаючи зсув).

Знаходження параметру Q_{test}^2 відбувається шляхом перевірки за тестовою вибіркою. Для цього за навчальною вибіркою знаходиться рівняння регресії, після чого з його використанням знаходяться теоретичні значення

для спостережень, які не ввійшли до навчальної вибірки. Після чого знаходиться коефіцієнт детермінації між отриманими значеннями та експериментальними даними.

Для оцінювання значущості статистичної моделі (F-критерій) використовується F-критерій. Він показує, чи є регресійна модель статистично значущою. Велике значення F-критерію вказує на те, що модель має статистичну значущість і пояснює змінність даних краще, ніж проста модель. F – критерій розраховується з використанням формули:

$$F = \frac{R^2}{(1-R^2)} \frac{N-k-1}{k}, \quad (3.9)$$

де N – розмір навчальної вибірки (число задач);

R^2 – коефіцієнт детермінації;

k – число дескрипторів в регресійній моделі.

Якщо з використанням розрахованих показників можна стверджувати, що модель змін є працездатною, то відбувається перехід на наступний підетап. Якщо визначається, що модель не є працездатною, то відбувається вибір іншого дескриптору та повторюються розрахунки, що описані в п'ятому етапі. На підетапі «Оцінювання змін» з використанням побудованих моделей, відбувається розрахунок показника змін та робиться оцінювання змін. З використанням розрахованої величини аналізується можливий результат внесення змін, їх відтермінування або відмова від їх внесення. Знайдене значення є рекомендаційним, рішення про внесення змін приймається командою. Якщо робиться висновок про неможливість виконання змін в поточному спринті, то відбувається перехід на третій етап комбінованого методу.

В іншому випадку здійснюється перехід на наступний етап.

3.1.7 Реалізація етапу «Планування дій»

Шостим етапом комбінованого методу виступає «Планування дій».

На даному етапі відповідальний за виконання етапів комбінованого методу визначає, хто з проєктної команди буде реалізовувати зміни та перевіряти їх виконання.

У випадку, якщо ресурсів недостатньо, але завдання є пріоритетним, то відбувається аналіз можливих рішень проблеми. До можливих рішень можна віднести:

- вищеоплачувану понаднормовану роботу членів команди, призначених для виконання змін, у випадку їх згоди на цю роботу;
- залучення працівників інших команд до внесення змін;
- перегляд поточних задач в спринті та відмова від менш пріоритетних, з комунікацією із замовником.

Відповідальний за зміни, при необхідності, має розробити план внесення змін та оновити існуючі плани роботи.

Зокрема на даному етапі необхідно підготувати план комунікацій з метою інформування замовників про зміни.

3.1.8 Реалізація етапу «Управління переходом»

Сьомим етапом комбінованого методу виступає етап «Управління переходом». На даному етапі відбувається реалізація змін, якою має управляти відповідальний за реалізацію етапів. За умови використання моделей змін після їх внесення та отримання експериментального значення показника змін можна розрахувати коефіцієнт прогнозування на основі експериментально отриманих даних Q_{test}^2 за формулою:

$$Q_{\text{test}}^2 = 1 - \frac{y^{(\text{calc})}}{y^{(\text{pred})}}, \quad (3.10)$$

де $y^{(\text{calc})}$ – значення показника змін, розраховане з використанням моделі змін;

$y^{(\text{pred})}$ – значення показника змін після впровадження змін.

В ідеальному випадку значення Q_{test}^2 має бути близьким до 1.

3.1.9 Реалізація етапу «Оцінювання успішності процесу управління змінами»

Восьмим етапом комбінованого методу виступає «Оцінювання успішності процесу управління змінами». Даний етап відбувається за запитом або раз на декілька спринтів. Його метою є аналіз успішності процесу управління змінами за рахунок визначення параметрів, описаних в другому розділі пояснювальної записки, тобто Ch_{Cp} , Ch_{Sp} , δCh_{T} , Ch_{ITp} .

З використанням розрахованих показників робиться висновок про процес управління змінами на рівні задач. У випадку незадовільних показників рекомендується проводити аналіз можливих проблем їх вирішення.

На даному етапі проводиться опитування співробітників щодо їх ставлення до процесу управління змінами. Дане опитування можна проводити в тестовому форматі.

Для цього можна створити онлайн форму з питаннями та надіслати її працівникам. Питання, які будуть представлені в формі, визначаються відповідальним за внесення змін.

3.2 Класифікація та опис дескрипторів, які використовуються при реалізації довгострокового ІТ-проекту

В рамках класифікації дескрипторів, яка наведена в розділі 2, проведемо опис та аналіз класів, на які поділена множина визначених дескрипторів. До технічних дескрипторів відноситься інформація, яка відображає технологічні характеристики проєкту. На рівні однієї команди дані дескриптори можуть бути незначними, проте на рівні декількох команд або компанії вони можуть бути цінними для аналізу.

Приклади технічних дескрипторів наведені в таблиці 3.2.

Таблиця 3.2 – Технічні дескриптори

Назва дескриптору	Умовний показник
Бінарний дескриптор з інформацією про те, чи використовується бібліотека для створення інтерфейсів користувача <code>redux-toolkit</code> [36]	<code>isReduxToolkitUsed</code>
Бінарний дескриптор використання розширеного синтаксису мови програмування JavaScript, який додає типізацію TypeScript [37]	<code>isTSUsed</code>
Бінарний дескриптор використання мови стилів, що розширює звичайний CSS – LESS [38]	<code>isLESSUsed</code>
Бінарний дескриптор з інформацією про те, чи використовується мова програмування JAVA [39]	<code>isJAVAUsed</code>
Бінарний дескриптор з інформацією про те, чи використовується хмарна платформа AWS [40]	<code>isAWSUsed</code>
Дескриптор з інформацією про кількість мікросайтів	<code>micrositesAmount-</code>

На рівні проєктної команди можуть бути цінними дескриптори, які будуть відрізнятися від задачі до задачі. Наприклад, необхідність написання тестів або необхідність проведення тестування в специфічному середовищі.

Інформація про задачу може мати різне значення. Наприклад, це може бути загальна характеристика вимог, технічні особливості тощо. Приклади дескрипторів особливостей задачі наведені в таблиці 3.3.

Таблиця 3.3 – Дескриптори особливостей задачі

Назва дескриптору	Умовний показник
Дескриптор з інформацією про загальну кількість характеристик задачі	ACCount
Дескриптор з інформацією про кількість характеристик задачі, які мають бути виконані на користувачькій стороні застосунку	ACFECCount
Дескриптор з інформацією про кількість характеристик задачі, які мають бути виконані на серверній стороні застосунку	ACBECCount
Бінарний дескриптор з інформацією про те, чи є метою задачі проведення оновлення графічних дизайнів	isFERedesign
Бінарний дескриптор з інформацією про те, чи є метою задачі зміна технології реалізації певної частини функціоналу	isTechChange
Бінарний дескриптор з інформацією про те, чи метою задачі є створення нового компоненту	isNewCompon
Бінарний дескриптор з інформацією про те, чи необхідно оновити документацію в ході виконання задачі	isDUpdating
Бінарний дескриптор з інформацією про те, чи є в задачі невіршені, критичні питання або ризики, які не дають змогу виконати задачу повноцінно	isRedFlag
Бінарний дескриптор з інформацією про те, чи є необхідною допомога архітектора системи в ході виконання задачі	isSASupport
Бінарний дескриптор з інформацією про те, чи для розробки задачі необхідно тільки робити зміни на користувачькій стороні застосунку	isOnlyFE
Бінарний дескриптор з інформацією про те, чи для розробки задачі необхідно тільки робити зміни на серверній стороні застосунку	isOnlyBE
Бінарний дескриптор з інформацією про те, чи для розробки задачі необхідно тільки робити перевірку виконаного функціоналу	isOnlyQA
Дескриптор з інформацією про кількість вимог, доданих в ході реалізації задачі	isACNewAfter
Дескриптор з інформацією про кількість реквестів на зміну, створених під час виконання задачі	isCRNewAfter
Бінарний дескриптор з інформацією про те, чи готові дизайни для роботи перед виконанням задачі	isDesReady
Бінарний дескриптор з інформацією про те, чи проводилось демонстрування для замовника в ході реалізації задачі	isDemoDuring

Час є важливим показником при реалізації ІТ-проектів з кількох причин.

По-перше, в бізнесі швидкість впровадження може мати прямий вплив на конкурентоспроможність компанії. Швидке впровадження дозволяє отримати перевагу на ринку.

По-друге, витрати на проєкт можуть зростати з плином часу через зміни у вимогах, переклади та інші фактори.

Отже, керування часом є ключовим для забезпечення успішного завершення проєкту в межах бюджету та із задоволенням від сторони клієнта.

Приклади показників часу наведені в таблиці 3.4.

Таблиця 3.4 – Дескриптори часу (показники часу)

Назва дескриптору	Умовний показник
Дескриптор з інформацією про запланований час, необхідний на виконання задачі	timePlanned
Дескриптор з інформацією про реальний час, витрачений на виконання задачі	timeActual
Дескриптор з інформацією про різницю між реальним часом, витраченим на виконання задачі та запланованим часом, необхідним для виконання задачі	deltaTime

Приклади дескрипторів опису команди наведені в таблиці 3.5.

Таблиця 3.5 – Дескриптори опису команди

Назва дескриптору	Умовний показник
1	2
Дескриптор з інформацією про загальну кількість членів команди	teamCount
Дескриптор з інформацією про кількість розробників клієнтської сторони	FECCount
Дескриптор з інформацією про кількість розробників серверної сторони	BECCount
Дескриптор з інформацією про кількість тестувальників	QACount

Кінець таблиці 3.5

1	2
Дескриптор з інформацією про кількість членів команди з початковим рівнем в команді (рівень «Студент»)	traineeCount
Дескриптор з інформацією про кількість членів команди з високим рівнем в команді (рівень «Наставник»)	seniorCount

Дескриптор з інформацією про кількість членів команди з середнім рівнем в команді (рівень «Працівник»)	middleCount
Дескриптор з інформацією про кількість членів команди з відповідною вищою освітою в команді	withDegreeCount
Дескриптор з інформацією про кількість розробників з відповідною вищою освітою в команді	DevWithDegreeCount
Дескриптор з інформацією про кількість розробників користувачької сторони з відповідною вищою освітою в команді	FEWithDegreeCount
Дескриптор з інформацією про кількість розробників серверної сторони з відповідною вищою освітою в команді	BEWithDegreeCount

Приклади дескрипторів, пов'язаних з процесом тестування, наведені в таблиці 3.6.

Таблиця 3.6 – Дескриптори, пов'язані з процесом тестування

Назва дескриптору	Умовний показник
Бінарний дескриптор з інформацією про те, чи реалізоване автоматичне тестування для задачі	isATestingEx
Бінарний дескриптор з інформацією про те, чи потрібно реалізувати автоматичне при виконанні задачі	isATestingAd
Бінарний дескриптор з інформацією про те, чи існують описані кейси ручного тестування для певної задачі	isMTestCaseEx
Бінарний дескриптор з інформацією про те, чи потрібно описати кейси ручного тестування для певної задачі	isMTestCaseAd
Бінарний дескриптор з інформацією про те, чи необхідно проводити ручне тестування для певної задачі	isMTest

Приклади дескрипторів, пов'язаних з доступністю веб-сторінки, наведені в таблиці 3.7.

Таблиця 3.7 – Дескриптори доступу до веб-сторінки

Назва дескриптору	Умовний показник
Бінарний дескриптор з інформацією про те, чи є неприпустимим мати помилки рівня A [41]	isAccA
Бінарний дескриптор з інформацією про те, чи є неприпустимим мати помилки рівня AA [41]	isAccAA

Бінарний дескриптор з інформацією про те, чи є неприпустимим мати помилки рівня AAA [41]	isAccAAA
Бінарний дескриптор з інформацією про те, чи необхідно при виконанні задачі робити перевірку на доступність контенту	isAccC
Бінарний дескриптор з інформацією про те, чи необхідно при виконанні задачі робити перевірку на контрастність кольорів	isAccCC

Приклади дескрипторів, пов'язаних з організаційними процесами при реалізації ІТ-проєкту, наведені в таблиці 3.8.

Таблиця 3.8 – Дескриптори організаційних процесів при реалізації ІТ-проєкту

Назва дескриптору	Умовний показник
Бінарний дескриптор з інформацією про те, чи присутній процес освоєння	isOnboarding
Бінарний дескриптор з інформацією про те, чи ведеться документація	isDocumentation
Бінарний дескриптор, що вказує, чи здійснюється оцінка ризиків	isRiskAssessment
Бінарний дескриптор, що вказує, чи введено систему звітності	isReportingSystem

3.3 Висновки з дослідження особливостей реалізації комбінованого методу оцінювання змін часу виконання задач

За результатами дослідження та опису особливостей реалізації комбінованого методу оцінювання змін часу виконання задач довгострокового ІТ-проєкту було зроблено наступні висновки.

По-перше, під час дослідження сценарію використання комбінованого методу оцінювання змін часу виконання задач довгострокового ІТ-проєкту було розширено опис етапів методу. В розділі були описані та проаналізовані роботи, які здійснюються визначеними виконавцями на кожному етапі реалізації методу.

По-друге, було розглянуто особливості та обмеження використання методу OLS, методу робастного оцінювання (регресії Хьюбера) та підходів для оцінювання працездатності моделей змін.

По-третє, були описані приклади дескрипторів при виконанні довгострокового IT-проєкту.

Отже, в розділі було проведено дослідження особливостей реалізації дескрипторного підходу при виконанні IT-проєкту.

4 АПРОБАЦІЯ КОМБІНОВАНОГО МЕТОДУ ОЦІНЮВАННЯ ЗМІН ЧАСУ ВИКОНАННЯ ЗАДАЧ ПРИ РЕАЛІЗАЦІЇ IT-ПРОЄКТУ

4.1 Аналіз довгострокового проєкту розробки інформаційної системи для створення сайтів

В даному розділі розглядається довгостроковий проєкт розробки інформаційної системи для створення сайтів. Назва проєкту – «Web Constructor». Назва створюваної системи – «Construct».

Дослідження, які проводяться в даному розділі роботи, пов'язані з виконанням задач на рівні однієї команди розробки.

Система «Construct» реалізує адміністративний портал, на якому замовники мають змогу використовувати бібліотеку компонентів з метою створення сайтів. Система підтримує понад 20 мов та надає змогу створювати веб-сторінки для більшості країн світу. Створений інтерфейс вже знаходиться в експлуатації та продовжує розширюватись.

Тривалість реалізації проєкту «Web Constructor» становить 6 років, тому він є довгостроковим. Проєкт поділяється на підпроєкти в залежності від вимог. Наприклад, за час виконання проєкту мали місце підпроєкти з оновлення дизайнів, перехід на більш прогресивні та конкурентоспроможні технології, додавання нового функціоналу та інші.

Проєкт продовжує масштабуватись, тому зараз створюються нові команди та підпроєкти з метою виконання бізнес-завдань.

Деякими з робіт, що реалізуються при виконанні проєкту, є:

- створення нових компонентів системи;
- оновлення існуючих компонентів системи;
- створення нових шаблонів веб-сторінок;
- покращення продуктивності існуючих веб-сторінок;
- збір аналітичної інформації щодо використання системи;
- підтримка\оновлення системи;
- вирішення проблем з доступністю контенту;
- перехід на нові технології;
- реалізація можливості використання системи в нових країнах;
- оновлення дизайну системи.

ІТ-проект «Web Constructor» є підпроектом, основною метою якого є розробка та підтримка створеної системи. Одночасно з проектом «Web Constructor» виконуються інші пов'язані з ним проекти. Наприклад, проект розробки глобальних компонентів для всіх мікросайтів та проект внесення контенту в систему для різних країн та інші [42].

Функціонування проекту «Web Constructor» здійснюється в декількох підрозділах, основними з них є: відділ управління, відділ інклюзивності та доступності, відділ експертів з тестування програмного забезпечення (QA, Quality Assurance) [43], відділ розробників серверної частини (BE, Back End) [44], відділ розробників клієнтської частини (FE, Front End) [45], відділ бізнес-аналізу (BA, Business Analysis) [46], відділ забезпечення ефективності та швидкодії системи, департамент автоматизації та оптимізації розробки і впровадження програмного забезпечення (DevOps) [47], відділ стратегічної архітектури та консультацій (SA, Solution Architects) [48] та відділ дизайну користувацького інтерфейсу.

У виконанні проекту бере участь декілька команд. Робота відбувається з використанням технології SCRUM [18]. Більша частина команд є командами розробки, в кожній такій команді працює 6-10 співробітників. До такої команди зазвичай входить декілька BE розробників, FE розробників та QA. Один з учасників проекту відіграє управляючу роль та звітує іншим відділам про успішність виконання роботи, підтримує з ними комунікацію.

Розробка частин системи поділена на спринти, які складають 3 тижні.

Задачі ІТ-проекту, які виконуються в рамках одного спринту, можуть бути сплановані як такі:

- що мають бути повністю виконані в поточному спринті;
- що мають бути виконані в наступному спринті;
- що будуть виконуватись декілька спринтів.

Якщо завдання береться з урахуванням отримати закінчену версію програми (тобто, як релізна задача), на його виконання команда має витратити не більше ніж 2 тижні, тому що воно має бути перевірене

інженерами тестування та замовниками в ході приймального тестування (UAT) у середовищі програмного продукту [49]. Якщо завдання заплановане без урахування отримання закінченої версії програми (тобто, як нерелізна задача), розробка може відбуватись в рамках усього спринту, та для тестування задача буде віддана на наступний спринт.

Відповідно до класифікації на основні групи проєктів за інтересами PMI (Project Management Institute) даний проєкт відноситься до проєктів розробки інформаційних систем [4].

В таблиці 4.1 наведена додаткова класифікація проєкту «Web Constructor». Дана класифікація проведена за класифікаційними ознаками згідно з матеріалами джерел [2], [3].

Таблиця 4.1 – Класифікація проєкту «Web Constructor»

Класифікаційна ознака	Тип проєкту
За масштабом	Великий
За складністю	Технічно складний
За термінами реалізації	Мегапроєкт
За вимогами до обмеженості ресурсів множини проєктів	Програма
За характером проєкту та рівнем учасників	Міжнародний
За характером цільової задачі проєкту	Маркетинговий
За головною причиною виникнення проєкту	Необхідність структурно-функціональних перетворень
За знаходженням замовника	Зовнішній
Ступінь участі замовника в проєкті	Середня

Проєкт є міжнародним та включає співробітників з різних країн світу.

Основними технологіями, що використовуються при виконанні проєкту, є: JS [50], React [51], JAVA [39], SQL [52], HTML [27], CSS [53], Redux [36], TS [37], LESS [38], AWS [40].

Ризики та зміни, що мають місце при виконанні проєкту «Web Constructor», мають тісний зв'язок з його особливостями та класифікацією.

Розглянемо найбільш актуальні проблеми, які мають місце при виконанні проєкту «Web Constructor» на поточний момент, та їх класифікацію. Найбільш актуальною проблемою, яка зустрічається в діяльності команди проєкту, є проблема зміни характеристик задач під час їх виконання. Дуже часто вони змінюються в процесі розробки задачі після проведення оцінювання часу, необхідного для її виконання [54].

Характеристика вимоги – це опис того, що має бути виконане при реалізації задачі. Характеристики описуються та поділяються бізнес-аналітиками на характеристики на стороні серверної частини; та характеристики користувацького інтерфейсу.

При виконанні проєкту «Web Constructor» характеристики задач описуються в форматі таблиці та кожній з них надається універсальний ідентифікатор. Якщо задача має технічний характер, то характеристики можуть бути розтлумачені та описані архітектором системи. На момент обговорення задачі перед взяттям її для виконання в робочий спринт, характеристики вже мають бути описані та команда має провести завчасне ознайомлення з ними. Це надає можливість вирішити непорозуміння щодо задачі до її фінального обговорення (грумінг, етап SCRUM) [18].

Характеристики можуть бути пов'язані з користувацькою частиною (FE) або з серверною частиною (BE). Інколи одна й та ж характеристика може потребувати роботи з двох частин (BE та FE).

В такому випадку для дескрипторів вимог пропонується зараховувати таку вимогу до обох дескрипторів.

В залежності від характеристик задачі поділяються на технічні та бізнес-задачі. Бізнес-задачі виконують завдання замовників, технічні задачі існують з метою покращення технічних показників та якості створюваного продукту.

Розглянемо та проаналізуємо актуальні проблеми, які виникають в процесі реалізації проєкту.

По-перше, до таких проблем відносяться обмеження прямої комунікації між деякими відділами. Комунікація є ускладненою та інколи відбувається шляхом обговорень у формі листування, що в свою чергу може спричиняти повільне внесення редагувань (при необхідності) та призводити до змін, які вносяться вже в момент роботи над певними задачами.

По-друге, мають місце ситуації, коли зі сторони відділу дизайну в дизайнах (зазвичай Figma) відбуваються зміни під час реалізації певного функціоналу, тобто, коли задача вже пройшла стадію оцінювання командою розробки. Приклади найчастіших змін такі: заміна зображень, зміна кольорів, зміна шрифтів, зміна розмірів та внесення нових дизайнів для окремих випадків або розмірів екранів.

По-третє, після внесення змін в графічні дизайни дизайнери не завжди інформують про це бізнес-аналітиків. Пряма комунікація між командою розробки та дизайнерами на даному проєкті відсутня та відбувається через комунікацію з бізнес-аналітиками. Подібні зміни можуть робити неактуальними характеристики, описані бізнес-аналітиками (BA, Business Analytics), та створювати потребу у внесенні змін на рівні задач [22].

Якщо необхідні зміни є незначними та в команді є можливість їх виконати в межах задачі, над якою вона працює, їх виконують одразу. Якщо зміни є критичними та комплексними, то їх виносять як окрему задачу та привласнюють вищий пріоритет, замінюючи іншу задачу, заплановану на поточний спринт, на нову, якщо в команді є можливість її виконати, або спрощують вимоги до задач, що знаходяться в роботі. Задачу такого типу називають запитом на зміни (CR, Change request) [55].

Особливості реалізації серверної частини також призводять до ряду ризиків. У серверній частині використовується мова програмування JAVA та платформа AEM (Adobe Experience Manager), яка, на жаль, у деяких випадках призводить до складнощів та обмежень у розширенні функціоналу [56]. Це інколи призводить до того, що деякі завдання не можуть бути виконані вчасно, як заплановано. В деяких випадках завдання відтермінуються і

складається лист на сторону розробників АЕМ з проханням виправити знайдені проблеми.

Схожі проблеми та обмеження є також на стороні розробки користувацького інтерфейсу. Існує окремий внутрішній проєкт, що надає загальну для використання бібліотеку з компонентами, які мають використовуватись на рівні проєктів «Web Constructor». Це призводить до проблем інтеграції зі створюваною ними бібліотеками. Зокрема, деякі характеристики на паралельно виконуваних проєктах реалізовані по-різному, що також призводить до проблем та поскладнює процес інтеграції з суміжними системами.

Можливі проблеми існують і з використанням зовнішніх бібліотек при (node package manager) [57], [58]. Мали місце ситуації, коли оновлення версії бібліотеки призводило до значних дефектів системи та вимагало заміни використовуваної бібліотеки або прийняття рішення про використання старої версії поточної бібліотеки з прийняттям ризику того, що відмова від оновлень може викликати безпекові проблеми системи. В таких випадках можливо прийняти рішення про очікування оновлень необхідної бібліотеки, тому що про існуючу проблему можуть знати на стороні розробки цієї бібліотеки та планувати її вирішити в майбутніх релізах.

Мають місце проблеми у процесі оцінювання часу, необхідного для виконання певної задачі та реалізації змін на рівні задач [59]. У випадку, коли команда не може оцінити час, необхідний на виконання задачі або реалізації змін, з причини недостатньої інформації щодо технічних особливостей, пропонується виконати завдання з метою більш детального дослідження та оцінювання часу, необхідного на виконання задачі або реалізації зміни. Таку задачу називають перевіркою на можливість виконання (POC, Proof of concept). Проте, не завжди команда може знати про ризики та пропонувати проведення подібного дослідження. В таких випадках під час розробки виявляється необхідність внесення змін до характеристик або до розрахунку кількості часу, необхідного для виконання завдання.

Оцінювання часу, необхідного для виконання завдань, в різних командах досліджуваного проєкту відбувається по-різному. В команді, робота якої досліджується, оцінювання часу проводиться шляхом планування з використанням технології карток покеру (Poker Planning), яка полягає в перспективі оцінювання складності й обсягу задач за допомогою колегіальності [26]. Оцінки виставляються в розмірі від 1 до 13 всіма членами команди. Зустріч, на якій відбувається процес оцінювання, зазвичай проходить через тиждень після проведення зустрічі з метою обговорення задачі з бізнес-аналітиками. Це в свою чергу призводить до того, що оцінки можуть бути не точними та не актуальними, тому що команда може не пам'ятати про всі аспекти оцінюваної задачі.

Деякі наявні проблеми пов'язані з тим, що команда має працювати в стресових умовах. У зв'язку з високою конкурентністю, завдання мають виконуватись швидко та якісно. Ця проблема призводить інколи до необхідності понаднормової роботи працівників або роботи в вихідні дні. Часті випадки такої необхідності можуть призводити до емоційної втоми від роботи працівників та погіршувати ефективність діяльності команди та індивідуальних учасників проєкту. Окрім цього, актуальною є проблема компенсації роботи [60]. У зв'язку з міжнародною кризою в ІТ-ринку спостерігається ситуація, при якій замовники надають перевагу бюджетним рішенням та відмовляються впроваджувати новітні технології у зв'язку з грошовим ресурсом.

Це призводить до підвищення конкуренції на ринку та зниження рівня.

Проблемою, яка має місце на аналізованому проєкті, є, також, складність організаційної структури [60]. Проблема відображається в тому, що члени команди не знайомі особисто з менеджерами вищого рівня, та можуть не знати, до якої ланки в організаційній структурі варто звертатись з різноманітних питань. Це може призводити до проблеми повільної відповіді, коли працівник звертається до свого менеджера, який йде далі по ланкам.

Як проблему можна виокремити низький рівень досвіду певної ланки співробітників в роботі з частинами розроблюваної системи. Це пов'язано з довготривалістю виконання проєкту, тобто значна частина компонентів системи була реалізована співробітниками, які, іноді, вже не працюють над проєктом. Зокрема існує наявна проблема неякісного процесу передавання відомостей щодо проєкту та неякісного процесу адаптації співробітників.

Отже, виходячи з описаних вище проблем та інших проблем, що мають місце при виконанні проєкту, можна надати їх наступну класифікацію для досліджуваної команди: проблеми якості оцінювання часу, необхідного на виконання задач та внесення змін, середньострокове прогнозування, проблеми змін характеристик задач, проблеми емоційного характеру, проблеми рівня компенсації, проблеми необхідності понаднормової роботи, проблеми технічного характеру, проблеми комунікації, проблеми рівня знань працівників, складність організаційної структури, застаріла документація та відсутність якісного процесу адаптації співробітників, які виконують проєкт.

4.2 Апробація комбінованого методу оцінювання змін часу виконання задач ІТ-проєкту розробки ІС для створення сайтів

Комбінований методу оцінювання змін часу виконання задач було впроваджено в рамках проєкту «Web Constructor» для однієї команди з метою аналізу ефективності розробленого методу.

Відповідальним за виконання етапів комбінованого методу оцінювання змін часу виконання задач був призначений менеджер проєкту.

Спочатку було проведено перший етап комбінованого методу «Внутрішній проєктний аналіз».

До впровадження комбінованого методу при реалізації проєкту використовувався метод, заснований на моделі Бекхарда і Гарріса [13].

Менеджером було проведено опитування команди щодо сприйняття процесу управління змінами на рівні задач, який мав місце при реалізації проєкту. Питання були описані в анонімній анкеті (гугл-формі) та надіслані кожному учаснику проєкту. Запитання та відповіді учасників проєкту наведені в таблиці 4.2.

Таблиця 4.2 – Оцінювання ставлення співробітників до впровадження комбінованого методу оцінювання змін часу виконання задач

Запитання для оцінювання за шкалою від 0 до 10	Співробітник								Середня оцінка за запитанням
	1	2	3	4	5	6	7	8	
Швидкість впровадження змін: 10 – швидко	6	5	4	6	7	4	5	3	5
Комфортність процесів управління змінами: 10 – комфортні	5	4	5	6	7	6	8	7	6
Наявність стресових ситуацій під час впровадження змін: 10 – відсутні	7	8	6	7	7	6	8	6	6,875
Необхідність понаднормової роботи: 10 – відсутня	8	9	8	10	9	10	9	9	9
Складність оцінки змін: 10 – не складно	6	7	5	6	7	8	5	4	6
Середня оцінка співробітника	6,4	6,6	5,6	7	7,4	6,8	7	5,8	6,575

Згідно проведеного оцінювання середня оцінка методу управління змінами до впровадження комбінованого методу оцінювання змін часу виконання задач становить 6,575.

За участю всіх членів проєктної команди був проведений аналіз використання дескрипторів.

Кожному дескриптору було обрано умовну назву для використання в багах даних дескрипторів.

Основними дескрипторами на момент обговорення було визначено наступні:

- кількість характеристик задач, вимірених в одиницях (C_Amount);
- кількість характеристик задач, що мають бути реалізовані на клієнтській стороні, вимірених в одиницях (C_FE_Amount);
- кількість характеристик задач, що мають бути реалізовані на серверній стороні, вимірених в одиницях (C_BE_Amount);
- запланована кількість днів на виконання задачі (W_Days_Planned);
- витрачена кількість днів на виконання задачі (W_Days_General);
- кількість характеристик задачі під час завершення роботи над нею, вимірених в одиницях (C_Amount_Final);
- різниця кількості характеристик задачі на момент взяття задачі в роботу та кількості характеристик, яка має місце на момент завершення роботи над задачею в днях (C_Delta);
- різниця кількості запланованих днів на виконання задачі на момент її взяття в роботу та кількості днів, які були витрачені на її реалізацію в момент завершення роботи над нею (W_Days_Delta).

Наступним етапом виступає «Збір інформації у вигляді дескрипторів на кожній ітерації проєкту». На даному етапі проводився збір інформації у вигляді дескрипторів, шляхом їх внесення до бази даних дескрипторів.

Були проаналізовані задачі, які виконувались останні п'ять спринтів. Було описано дескриптори для цих задач в таблиці 4.3.

Таблиця 4.3 – Дескриптори для задач, що виконувались останні п'ять спринтів

Ідентифікатор задачі	C_Amount, штук	C_Amount_Final, штук	C_FE_Amount, штук	C_BE_Amount, штук	W_Days_Planned, днів	W_Days_General, днів
WEBCO_2342	80	92	30	50	15	20
WEBCO_2343	80	94	65	15	23	31
WEBCO_2344	20	25	10	10	8	10
WEBCO_2345	35	38	15	20	7	8

WEBCO_2346	42	47	12	30	12	15
WEBCO_2347	76	85	26	50	9	14
WEBCO_2348	12	12	12	0	5	5
WEBCO_2349	56	61	26	30	16	18
WEBCO_2350	15	20	10	5	5	7
WEBCO_2351	71	75	25	46	14	16
WEBCO_2352	12	12	0	12	3	3
WEBCO_2353	7	10	7	0	3	4
WEBCO_2354	2	2	2	0	1	1
WEBCO_2355	1	8	1	0	2	6
WEBCO_2356	35	40	32	3	18	20
WEBCO_2357	8	10	1	7	4	5
WEBCO_2358	54	65	53	1	50	55
WEBCO_2359	34	53	25	9	74	84
WEBCO_2375	50	43	30	20	15	12
WEBCO_2376	40	33	25	15	12	9
WEBCO_2377	35	28	20	15	10	8
WEBCO_2380	20	15	12	8	7	5
WEBCO_2384	70	62	40	30	20	16
WEBCO_2385	50	42	28	22	15	12
WEBCO_2386	45	32	25	20	13	8
WEBCO_2387	40	30	22	18	12	8
WEBCO_2388	35	32	20	15	11	10
WEBCO_2342	80	92	30	50	15	20

В рамках наступного спринту дескриптори також вносились в базу даних дескрипторів. Дані для шостого спринту відображені в таблиці 4.4.

Таблиця 4.4 – Дескриптори для завдань, що виконувались під час робіт шостого спринту

Ідентифікатор задачі	C_Amount, штуки	C_Amount_Final, штуки	C_FE_Amount, штуки	C_BE_Amount, штуки	W_Days_Planned, дні	W_Days_General, дні
WEBCO_2389	30	26	18	12	10	9
WEBCO_2395	20	20	12	8	8	8
WEBCO_2396	25	21	15	10	9	7

WEBCO_ 2397	30	35	18	12	10	13
WEBCO_ 2398	22	24	14	8	8	9
WEBCO_ 2399	35	35	20	15	11	11
WEBCO_ 2360	65	75	40	25	20	25
WEBCO_ 2361	25	59	15	10	8	23

Наступним етапом виступає «Визначення потреби в змінах».

Під час робіт шостого спринту, при виконанні задачі WEBCO_2360 з'явилась необхідність внесення змін на рівні задачі. В рамках задачі відбувалось оновлення дизайнів компоненту нижнього меню сторінки (футеру). Під час виконання задачі було виявлено необхідність внесення змін до дизайнів графічного інтерфейсу. Дизайнером було оновлено графічний інтерфейс, бізнес-аналітик оновив характеристики компоненту відповідно до оновлень дизайнів.

Наступним етапом виступає «Аналіз відмінностей між поточним станом та бажаним». На даному етапі було детально вивчено різницю станів задачі без внесення змін та з їх внесенням.

Команді було розтлумачено зміни, які необхідні для реалізації.

Команда розробки запросила проведення п'ятого етапу комбінованого методу «Визначення можливостей реалізації змін на поточній ітерації виконання проєкту». Для побудови статистичних моделей змін було обрано дескриптор, який визначає різницю кількості характеристик задачі на момент взяття задачі в роботу та кількості характеристик, яка має місце на момент завершення роботи над задачею (C_Delta). Показником змін було обрано різницю кількості запланованих днів на задачу на момент її взяття в роботу та кількості днів, які були витрачені на її реалізацію в момент завершення роботи над нею (W_Days_Delta). Першим кроком проводиться перевірка виконання передумови використання регресійного аналізу. Вона проводиться для даних, зібраних на основі перших п'яти спринтів.

З використанням реалізованої програми будується розподіл Гаусса, представлений на рисунку 4.1.

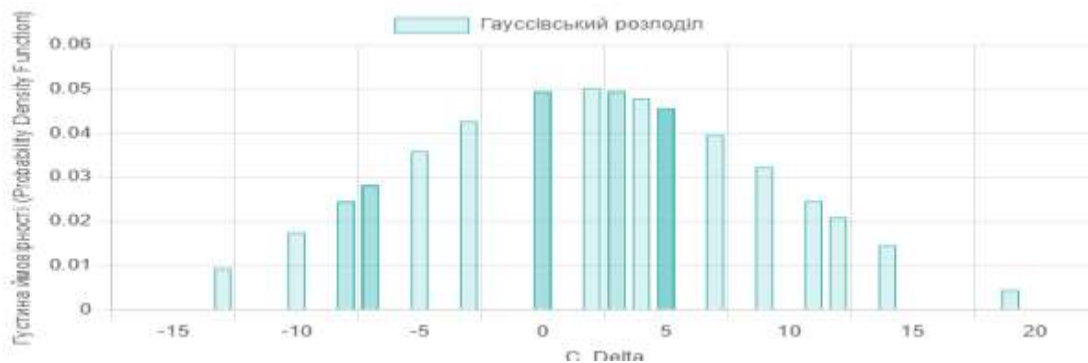


Рисунок 4.1 – Розподіл Гаусса для даних кількості змінених характеристик, які були зібрані для перших п'яти спринтів

З використанням програми розраховується середнє значення, дисперсія та стандартне відхилення значень дескрипторів, зібраних для перших п'яти спринтів. Лістинг коду цих розрахунків наведений в додатку А.

Середнє значення становило 1.52.

Дисперсія (Variance) становила 62.72.

Стандартне відхилення (Standard Deviation) становило 7.92.

Аномальні значення за межами 3σ відсутні.

Перевірка показала відсутність викидів в даних, тому проводиться побудова моделей з використанням методу OLS.

Для розрахунків використовується програма, створена в процесі роботи над кваліфікаційною роботою.

Розрахунки моделей проводяться для дескрипторів, зібраних з перших п'яти спринтів. Лістинг коду цих розрахунків за методом OLS наведений в додатку А.

Модель, що розрахована за методом OLS, наведена як формула:

$$y_i = 0,46 * x_i + 0,26, \quad (4.1)$$

де y_i – значення показника змін W_Days_Delta (властивість системи);

x_i – значення дескрипторів C_Delta .

Модель змін, що побудована за методом OLS для даних перших п'яти спринтів, зображена на графіку (рисунок 4.2).

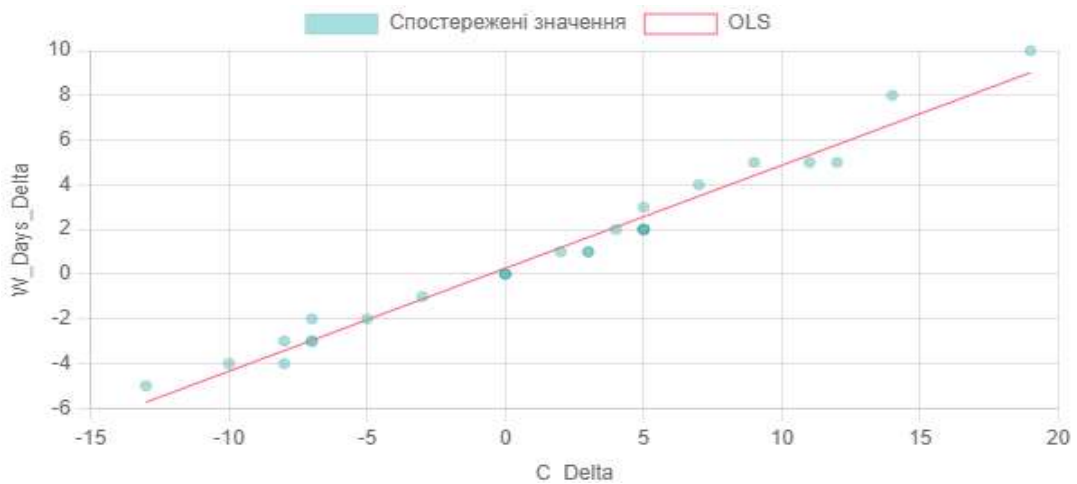


Рисунок 4.2 – Модель змін, що побудована з використанням методу OLS, для даних перших п'яти спринтів

Після побудови моделі була проведена перевірка її працездатності з використанням наступних перевірок за показниками: R^2 , Q^2 , Q_{LOOCV}^2 , F-критерію та стандартного відхилення σ .

Розрахунок цих показників відбувався з використанням програми, створеної під час виконання кваліфікаційної роботи. Лістинг коду при використанні методів розрахунку показників наведений в додатку А. Розраховані показники працездатності для моделі змін, побудованої на основі дескрипторів для задач перших п'яти спринтів, наведені в таблиці 4.5.

Таблиця 4.5 – Показники працездатності для моделі змін, побудованої з використанням методу OLS

Метод	Рівняння регресії	R^2	Q^2	Q_{LOOCV}^2	F	σ
OLS	$y_i = 0,46 * x_i + 0,26$	0.98	0.98	0.97	481.61	0.59

Розраховані показники працездатності моделі показали, що модель є працездатною та може бути використана для оцінювання показника змін. Під час виконання шостого спринту вирішувались питання щодо оновлення списку характеристик задачі WEBCO_2360 після взяття її в роботу. Початкова кількість характеристик складала 65 одиниць.

Перед командою було завдання обрати один з запропонованих варіантів:

- додати 20 нових характеристик до поточної задачі;
- додати 10 пріоритетних нових характеристик до поточної задачі та створення задачі з 10 іншими характеристиками для виконання в наступному спринті.

Розрахунок шуканої величини за умови додавання 10 та 20 нових характеристик наведений в таблиці 4.6.

Таблиця 4.6 – Розрахунок оцінки додаткових днів, необхідних для виконання завдання WEBCO_2360 у зв'язку зі змінами кількості характеристик

Метод	Рівняння регресії	Зміна днів (20 характеристик)	Зміна днів (10 характеристик)
OLS	$y_i=0,46*x_i+0,26$	9,46	4,86

З використанням отриманої оцінки змін було вирішено в поточному спринті додати 10 нових характеристик.

Менеджер проєкту назначив відповідальних за внесення змін та їх перевірку.

На внесення змін було витрачено на 5 днів більше від запланованого, що свідчить про те, що з використанням комбінованого методу оцінювання змін часу виконання задач було отримано близьку оцінку. Розрахунки відношення прогнозованого значення до реального значення наведені в таблиці 4.7.

Таблиця 4.7 – Розрахунок відношення прогнозованого значення часу до реального значення часу для виконання додаткових характеристик задачі WEBSCO_2360

Метод	Реальна зміна, дні	Прогнозована зміна, дні	Відношення прогнозованого та реального результатів
OLS	5	4,86	0,97

Далі розглядається випадок використання дескрипторного підходу при виконанні сьомого спринту.

При виконанні сьомого спринту, вирішувались питання щодо зменшення списку характеристик задачі WEBSCO_2361 після взяття її в роботу. Початкова кількість характеристик складала 55 одиниць.

Після взяття завдання в роботу, було виявлено, що 7 характеристик більш не є актуальними. З використанням комбінованого методу оцінювання змін часу виконання задач було вирішено розрахувати приблизне значення звільнених днів розробки. Під час реалізації комбінованого методу команда розробки запросила проведення п'ятого етапу комбінованого методу «Визначення можливостей реалізації змін на поточній ітерації виконання проєкту».

Для побудови статистичних моделей змін було вирішено використати дескриптор C_Delta та показник змін W_Days_Delta , як для попередніх розрахунків. Проводилась перевірка на викиди даних, зібраних на основі перших шести спринтів.

Розподіл Гаусса для даних кількості змінених характеристик, які були зібрані для попередніх шести спринтів, зображений на рисунку 4.3.

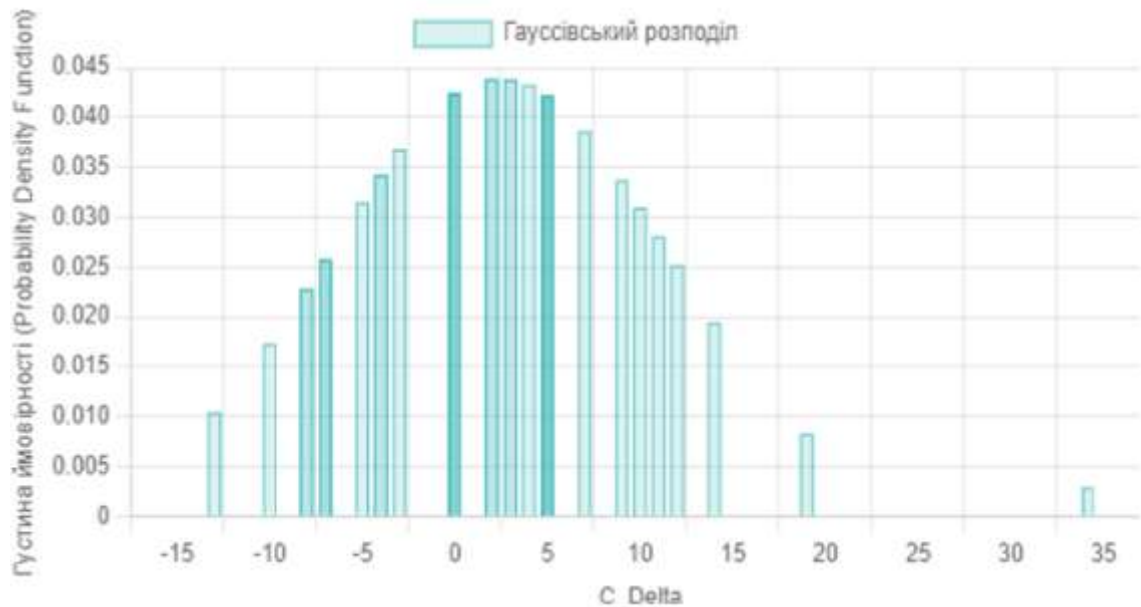


Рисунок 4.3 – Розподіл Гаусса для даних кількості змінених характеристик, які були зібрані для попередніх шести спринтів

Розраховане середнє значення становило 2.4.

Дисперсія (Variance) становила 82.6.

Стандартне відхилення (Standard Deviation) становило 9.09.

Перевірка методом «3 сигми» показала присутність викиду в даних зі значенням 34.

Перевірка показала присутність викидів, тому проводилась побудова моделей з використанням регресії Хьюбера. Лістинг коду для розрахунків моделі з використанням методу регресії Хьюбера наведений в додатку А. Проводились розрахунки моделей для дескрипторів, зібраних з перших шести спринтів, з використанням наступних параметрів δ (дельта): 1.345, 0.8, 0.1, 0,02 та 1.5. Для знайдених моделей змін проводимо перевірку працездатності. Розрахунки наведені в таблиці 4.8.

Таблиця 4.8 – Моделі, розраховані з використанням регресії Хьюбера, для задач перших семи спринтів з їх показниками працездатності

Параметр δ (дельта)	Рівняння регресії	R^2	Q^2	Q_{LOOCV}^2	F	σ
1.345	$y_i = 0,25 * x_i + 0,01$	0.76	0.76	0.78	51	2.09

0.8	$y_i = 0,24 * x_i + 0,01$	0.74	0.74	0.76	46	2.16
0.1	$y_i = 0,37 * x_i + 0,01$	0.94	0.94	0.93	259	1.02
0.02	$y_i = 0,41 * x_i + 0,02$	0.97	0.97	0.97	517	0.74
1.5	$y_i = 0,25 * x_i + 0,01$	0.76	0.76	0.78	51	2.07

Відповідно до розрахованих показників працездатності для подальших розрахунків було обрано значення $\delta = 0.02$.

Модель, побудована з використанням регресії Хьюбера, для даних перших шести спринтів зображена на графіку на рисунку 4.4.

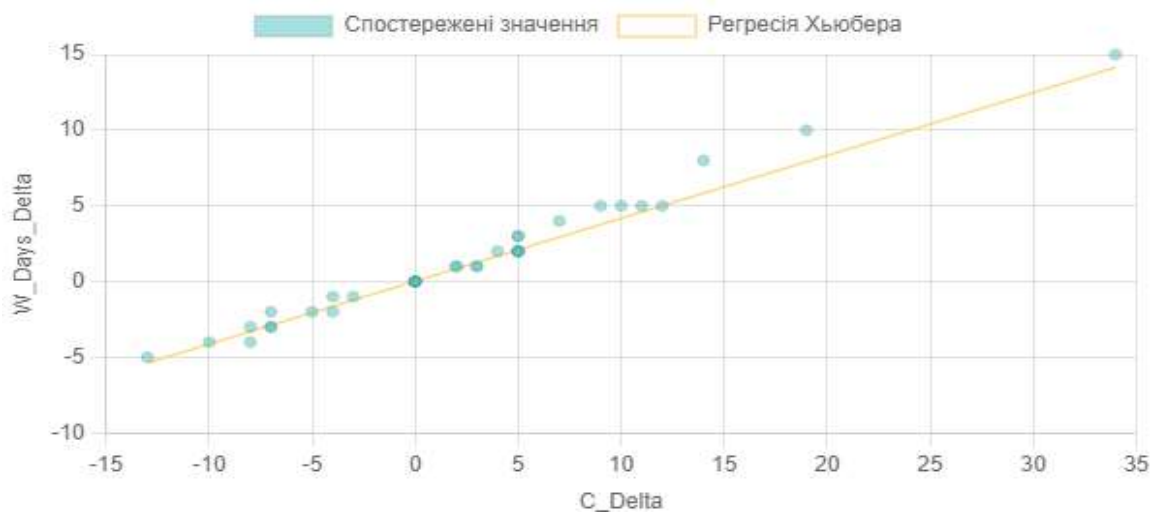


Рисунок 4.4 – Модель, побудована з використанням регресії Хьюбера, для даних перших шести спринтів

Розраховані показники працездатності моделі показали, що модель є працездатною та може бути використана для оцінювання показника змін часу виконання задач. Розрахунок оцінки звільнених днів за умови зміни кількості характеристик задачі WEBCO_2361 наведені в таблиці 4.9.

Таблиця 4.9 – Розрахунок оцінки звільнених днів при умові зміни кількості характеристик задачі WEBCO_2361

Метод	Рівняння регресії	Показник змін
Регресія Хьюбера	$y_i = 0,41 * x_i + 0,02$	2.85

Для задачі WEBCO_2361 було вирішено прибрати 7 характеристик з опису задачі в поточному спринті. На завдання витрачено на 4 дні менше. Розрахунки відношення прогнозованого значення до реального значення наведені в таблиці 4.10.

Таблиця 4.10 – Розрахунок відношення прогнозованого значення до реального для задачі WEBCO_2361

Метод	Реальна зміна, дні	Прогнозована зміна, дні	Відношення прогнозованого та реального результатів
Регресія Хьюбера	3.5	-2.85	0,81

Отже, розраховані оцінки змін виявились близькими до реальних змін. Що підтвердило прогностичних розрахованих моделей змін та можливість їх використання на запит команди.

4.3 Оцінювання успішності процесу управління змінами

В даному підрозділі наведені результати оцінювання успішності процесу управління змінами часу виконання задач.

На першому етапі комбінованого методу «Внутрішній проєктний аналіз» менеджером проводиться анонімне опитування членів проєктної команди щодо процесу управління змінами.

Середня оцінка при проведенні опитування становила – 6,575.

Після семи спринтів використання комбінованого методу опитування було проведене ще раз. Результати опитування наведені в таблиці 4.11.

Таблиця 4.11 – Результати оцінювання ставлення співробітників до процесів управління змінами після впровадження комбінованого методу

Запитання для оцінювання за шкалою від 0 до 10	Співробітники								Середня оцінка за питанням
	1	2	3	4	5	6	7	8	
Швидкість впровадження змін: 10 – швидко	8	7	8	7	8	6	7	8	7,375
Комфортність процесів управління змінами: 10 – комфортні	7	9	7	8	8	9	9	7	8
Наявність стресових ситуацій під час впровадження змін: 10 – відсутні	8	8	7	8	7	8	9	8	7,875
Необхідність понаднормової роботи: 10 – відсутня	9	9	10	10	9	10	9	10	9,5
Складність оцінки змін: 10 – не складно	7	7	9	8	7	8	7	6	7,375
Середня оцінка співробітника	7,8	8	8,2	8,2	7,8	8,2	8,2	7,8	8,025

Середня оцінка методу управління змінами до внесення змін в процес управління змінами на рівні задач становить – 8,025.

З цього можна зробити висновок, що сприйняття команди методу управління змінами на рівні задач зросло приблизно на 7,4%.

Показники оцінювання успішності процесу управління змінами в проєкті були розраховані до впровадження комбінованого методу, коли в роботі проєкту для управління змінами використовувався метод, який заснований на моделі Бекхарда і Гарріса.

Також показники успішності процесу управління змінами були розраховані після впровадження комбінованого методу.

Значення показників до впровадження комбінованого методу та після наведені на таблиці 4.12.

Таблиця 4.12 – Показники оцінювання успішності процесу управління змінами в комбінованому методі до впровадження комбінованого методу та

після його впровадження

Показник	Метод, заснований на моделі Бекхарда і Гарріса	Комбінований метод оцінювання змін часу виконання задач
Відсоток скасованих змін часу у зв'язку з неможливістю їх реалізації Ch_{cp}	7	5
Відсоток змін часу, що були прийняті замовником та затверджені як успішно реалізовані Ch_{sp}	92	95
Відхилення між часом, який був витрачений на виконання робіт, і оціненим часом δCh_T	7	2
Відсоток змін, які були завершені вчасно Ch_{ITp}	79	86

Зокрема, під час використання комбінованого методу оцінювання змін часу виконання задач зроблені наступні висновки.

По-перше, комбінований метод надає можливість кількісно оцінювати зміни на рівні задач, що є зручним інструментом для роботи зі схожими змінами у проєкті;

По-друге, тривалість процесу розрахунку моделей буде значно збільшуватись при збільшенні кількості дескрипторів, взятих для аналізу, тому можна покращувати розрахункові операції.

Також, можливо робити вибірки задач для аналізу, щоб не виконувати аналіз серед всіх задач.

4.4 Висновки з четвертого розділу кваліфікаційної роботи

В процесі апробації комбінованого методу оцінювання змін часу виконання задач ІТ-проєкту з розробки конструктору сайтів було зроблено наступні висновки.

По-перше, було проведено аналіз проєкту розробки інформаційної системи для створення сайтів. Проаналізовано бізнес-процеси, які мають місце в проєкті, та технології, що використовуються. Зокрема, була проведена класифікація проєкту та аналіз ризиків та змін з ними пов'язаними, які найчастіше мають місце при вирішенні задач на рівні команди розробки.

По-друге, комбінований метод оцінювання змін часу виконання задач при реалізації ІТ-проєкту було апробовано на проєкті розробки інформаційної системи для створення сайтів «Web Constructor». Апробація методу відбувалась на рівні однієї проєктної команди розробки веб-сайтів впродовж декількох спринтів реалізації оновлень дизайнів графічного інтерфейсу системи та сайтів, який має місце раз на два роки.

Під час апробації були виконані всі етапи комбінованого методу.

По-третє, були розраховані показники оцінювання успішності процесу управління змінами при використанні комбінованого методу та без його використання.

З метою оцінювання успішності процесу управління змінами було використано наступні показники:

- відсоток скасованих змін часу у зв'язку з неможливістю їх реалізації Ch_{Cp} ;
- відсоток змін часу, що були прийняті замовником та затверджені, як успішно реалізовані Ch_{Sp} ;
- відхилення між часом, який був витрачений на виконання робіт, і оціненим часом δCh_T ;
- відсоток змін, які були завершені вчасно Ch_{ITp} ;
- анонімне опитування щодо ставлення співробітників до процесів управління змінами.

Порівняння значень показників оцінювання успішності процесу управління змінами до впровадження комбінованого методу та після наведені

в таблиці 4.13.

Таблиця 4.13 – Порівняння значень показників оцінювання успішності процесу управління змінами до впровадження комбінованого методу та після

Показник	Метод, заснований на моделі Бекхарда і Гарріса	Комбінований метод оцінювання змін часу виконання задач	Зміна показника, %
Відсоток скасованих змін часу у зв'язку з неможливістю їх реалізації Ch_{CP}	7	5	2
Відсоток змін часу, що були прийняті замовником та затверджені як успішно реалізовані Ch_{SP}	92	95	3
Відхилення між часом, який був витрачений на виконання робіт, і оціненим часом δCh_T	7	2	28
Відсоток змін, які були завершені вчасно Ch_{ITP}	79	86	7
Оцінка ставлення співробітників до процесів управління змінами відповідно до проведеного анонімного опитування	6,575	8,025	7,4

Отже, можна зробити висновок, що після впровадження комбінованого методу спостерігається покращення показників оцінювання успішності процесу управління змінами.

ВИСНОВКИ

Відповідно до завдання магістерської кваліфікаційної роботи було проведено аналіз існуючих методів управління змінами при виконанні ІТ-проектів.

Під час реалізації кваліфікаційної роботи було виконано наступні пункти:

- проаналізовано особливості реалізації довгострокових ІТ-проектів;
- проаналізовано та узагальнено існуючі методи управління змінами в ІТ-проектах;
- визначено критерії вибору методу управління змінами в ІТ-проектах відповідно до мети його адаптації для використання на рівні задач з можливістю оцінювання змін;
- розроблено підхід оцінювання змін часу при реалізації довгострокових проектів.
- розроблено комбінований метод оцінювання змін з використанням існуючого методу управління змінами при реалізації ІТ-проектів;
- практично реалізовано запропонований метод на прикладі ІТ-проекту розробки системи створення сайтів;
- експериментально перевірено ефективність розробленого комбінованого методу управління змінами часу на рівні задач.

Під час виконання кваліфікаційної роботи було описано особливості реалізації та застосування комбінованого методу, описано його етапи та відповідальних за їх виконання.

Для перевірки ефективності комбінованого методу було проведено його апробацію при виконанні довгострокового ІТ-проекту розробки системи створення сайтів. Для цього було описано показники ефективності та проведено опитування членів команди.

Порівняння комбінованого методу з методом заснованим на моделі Бекхарда і Гарріса продемонструвало покращення показників ефективності процесом управління змінами на рівні задач та процесом їх оцінювання.

Розроблений комбінований метод може бути використаний з метою кількісного оцінювання змін на рівні задач при виконанні довгострокових проектів. Показником змін може бути обране значення актуальне для певної задачі.

Графічний матеріал, розроблений при виконанні кваліфікаційної роботи, наведений в додатку Б.

За тематикою кваліфікаційної роботи опубліковано тези доповіді на тему «Аналіз використання дескрипторного підходу при управлінні змінами в ІТ-проектах» на 28-й Міжнародному молодіжному форумі «Радіоелектроніка та молодь у ХХІ столітті» (Харків, ХНУРЕ) [61].

Також, за тематикою кваліфікаційної роботи опубліковано тези доповіді на тему «Оцінювання успішності управління змінами в довгострокових ІТ-проектах» на 26-ту Міжнародну науково-практичну конференцію «Theoretical and Practical Aspects of Modern Research» (Канада, Оттава) [62].

Кваліфікаційна робота виконувалась згідно з вимогами методичних вказівок з організації та виконанню кваліфікаційної роботи [63] та національним стандартом ДСТУ 3008:2015 «Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлення» [64].

Перелік джерел посилання оформлено згідно з національним стандартом ДСТУ 8302:2015 «Бібліографічне посилання. Загальні положення та правила складання» [65].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Добровська Л. М., Коваленко О. С., Аверьянова О. А. Управління ІТ-проєктами : навч. посіб. Київ, 2022. 284 с.
2. Кобилянський Л.С. Управління проєктами : навч. посіб. Київ, 2002. 200 с.
3. Катренко А.В. Управління ІТ-проєктами : Новий світ, 2011. 550 с.
4. Managing Change in Organizations : A Practice Guide, Project Management Institute, 2013. 127 p.
5. Довгань Л. Є., Мохонько Г. А., Малик І. П. Управління проєктами: підручник для студентів – магістрів галузі знань 07 «Управління та адміністрування» спеціальності 073 «Менеджмент». Київ: КПІ ім. Ігоря Сікорського, 2017. 420 с.
6. Yandarbaeva L. A., Kostoeva A. A. Gayrbekova R. S. Features and methodology of change management // The European Proceedings of Social and Behavioral Sciences EpSBS. 2022. P. 1262–1268.
7. Управління проєктами: навч. посіб. / Л. П. Батенко та ін. Київ, 2003. 231 с.
8. Бабаєв В.М. Управління проєктами : навч. посіб. Харків, 2006. 244 с.
9. Adkar model // Сайт «Maxzosim». URL: <https://www.maxzosim.com/adkar-model/> (дата звернення: 05.04.2024).
10. Adkar Prosci // Сайт «Prosci». URL : <https://www.prosci.com/methodology/adkar> (дата звернення: 05.04.2024).
11. Aim change management methodology // Сайт «Imaworldwide». URL: <https://www.imaworldwide.com/aim-change-management-methodology> (дата звернення: 05.04.2024).

12. Aim change management methodology // Сайт «Aim». URL: <https://www.aim.com.au/leadership-strategy/courses/change-management-embrace-evolve-thrive> (дата звернення: 05.04.2024).

13. Beckhard & Harris Change Process // Сайт «Praxie». URL: <https://praxie.com/beckhard-harris-change-process-online-tools-templates-web-software/> (дата звернення: 05.04.2024).

14. Bridges Transition Model // Сайт «Wmbridges». URL: <https://wmbridges.com/about/what-is-transition/> (дата звернення: 07.04.2024).

15. The 8 Steps for Leading Change // Сайт «Kotterinc». URL: <https://www.kotterinc.com/methodology/8-steps/> (дата звернення: 07.04.2024).

16. The Kübler Ross Change Curve in the Workplace (2024) // Сайт «Whatfix». URL: <https://whatfix.com/blog/kubler-ross-change-curve/> (дата звернення: 08.04.2024).

17. Lewin's 3-Stage Model of Change Theory: Overview // Сайт «Whatfix». URL: <https://whatfix.com/blog/lewins-change-model/> (дата звернення: 15.04.2024).

18. Malhotra V. Single Reference Guide for Scrum Certification (Professional Scrum Master I (PSM I) and Professional Scrum Product Owner I (PSPO I) Certification) : Vishal Malhotra, 2020. 169 p.

19. Larman C., Multicoloniarnist B. V. Large-Scale Scrum : Pearson Education, 2016. 368 p.

20. Van Huffel S. Recent Advances in Total Least Squares Techniques and Errors-in-variables Modeling : Society for Industrial and Applied Mathematics, 1997. 377 p.

21. Diego O., Essam H. H., Salvador H.. Metaheuristics in Machine Learning Theory and Applications : CRC Press. 2021. 769 p.

22. Doar M. Practical JIRA Administration : O'Reilly Media, 2011. 71 p.

23. Firebase. Make your app the best it can be // Сайт «Firebase». URL: <https://firebase.google.com/> (дата звернення: 14.05.2024).

24. MongoDB Atlas. Fully managed MongoDB in the cloud // Сайт «Mongodb». URL: <https://www.mongodb.com/lp/cloud/atlas/> (дата звернення: 14.04.2024).

25. PostgreSQL: The World's Most Advanced Open Source Relational Database // Сайт «Postgresql». URL: <https://www.postgresql.org/> (дата звернення: 20.04.2024).

26. Ioannou A., Zaphiris P. Learning and Collaboration Technologies : Springer Nature Switzerland, 2023. 602 p.

27. Coremans C. HTML: A Beginner's Tutorial : Brainy Software Inc. 2015. 110 p.

28. Web Accessibility Initiative. WAI fundamentals main page // Сайт «W3». URL: <https://www.w3.org/WAI/fundamentals/accessibility-intro/> (дата звернення: 18.04.2024).

29. Accessiblyapp. Accessibility Levels: WCAG Compliance A vs. AA vs. AAA // Сайт «Accessiblyapp». URL: <https://accessiblyapp.com/blog/accessibility-levels-wcag-a-aa-aaa/> (дата звернення: 05.05.2024).

30. DEV. Server-side Rendering (SSR) From Scratch with React // Сайт «Dev.to». URL: <https://dev.to/woovi/server-side-rendering-ssr-from-scratch-with-react-19jm> (дата звернення: 05.05.2024).

31. Anand M. J. Statistical Methods for Six Sigma : Wiley, 2003. 344 p.

32. Ranga Suri N. N. R., Narasimha M. M., Athithan G. Outlier Detection Techniques and Applications : Springer International Publishing, 2019. 214 p.

33. Furno M., Vistocco D. Quantile Regression : Wiley, 2018. 312 p.

34. Farebrother R. W. Linear Least Squares Computations : CRC Press, 2018. 320 p.

35. Geisser S. Predictive Inference : CRC Press, 2017. 240 p.

36. Redux-toolkit. Home page // Сайт «Redux-toolkit». URL: <https://redux-toolkit.js.org/> (дата звернення: 15.05.2024).

37. Type Script. Type Script documentation main page // Сайт «Typescriptlang». URL: <https://www.typescriptlang.org/> (дата звернення: 15.05.2024).

38. LESS. LESS documentation main page // Сайт «Lesscss». URL: <https://lesscss.org/features/> (дата звернення: 16.05.2024).

39. Kurniawan B. Java a Beginner's Tutorial (Fifth Edition) : Brainy Software, 2019. 522 p.

40. AWS. AWS documentation main page // Сайт «Docs.aws.amazon». URL: <https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/home.html> (дата звернення: 18.05.2024).

41. Harper S., Yesilada Y. Web Accessibility : Springer London, 2019. 807 p.

42. Mailchimp. What Is a Microsite? // Сайт «Mailchimp». URL: <https://mailchimp.com/resources/what-is-a-microsite/> (дата звернення: 20.05.2024).

43. Stapp L., Roman A., Pilaeten M. ISTQB Certified Tester Foundation Level : Springer Nature Switzerland, 2023. 406 p.

44. Marquez-Soto P. Backend Developer in 30 Days : Vpb Publications, 2022. 464 p.

45. Strazzullo F. Frameworkless Front-End Development : Apress, 2019. 248 p.

46. Milani F. Digital Business Analysis : Springer International Publishing, 2019. 429 p.

47. Skrynnik O. DevOps Foundation Courseware – English : Van Haren Publishing, 2019. 156 p.

48. Shrivastava S., Srivastav N. Solutions Architect's Handbook : Packt Publishing. 2020. 490 p.

49. UAT testing // Сайт «Highload.today». URL: <https://highload.today/uk/user-acceptance-testing-uat-prijmalne-testuvannya-ta-jogo-tsili/> (дата звернення: 25.05.2024).

50. Kirupa C. JavaScript, A Beginner's Guide, Third Edition : Pearson Education, 2022. 624 p.
51. React. React documentation main page // Сайт «Uk.legacy.reactjs». URL: <https://uk.legacy.reactjs.org/> (дата звернення: 23.05.2024).
52. Darmawikarta D. Oracle SQL, A Beginner's Tutorial, Second Edition : Brainy software, 2016. 150 p.
53. w3school. CSS Tutorial page // Сайт «W3schools». URL: <https://www.w3schools.com/css/> (дата звернення: 23.05.2024).
54. Guizzi G., Fujita H. New Trends in Software Methodologies, Tools and Techniques : IOS Press, 2023. 304 p.
55. Roberts P. The Economist Guide to Change and Project Management : Profile, 2020. 341 p.
56. Lunka R. Adobe Experience Manager : Pearson Education, 2013. 368 p.
57. Ali J. Instant Node Package Manager : Packt Publishing, 2013. 56 p.
58. NPM. Home page // Сайт «Npmjs». URL: <https://www.npmjs.com/> (дата звернення: 22.05.2024).
59. Silhavy P., Silhavy R., Prokopova Z. Software Engineering Application in Systems Design : Springer International Publishing, 2022. 855 p.
60. Bott F. Professional Issues in Software Engineering : Taylor & Francis, 2005. 266 p.
61. Попова А. В., Васильцова Н.В. Аналіз використання дескрипторного підходу при управлінні змінами в ІТ-проекті // XXVIII Міжнародний молодіжний форум «Радіоелектроніка та молодь у XXI столітті». Зб. матеріалів форуму. Т. 6., Ч.2. (16-18 квітня 2024р.) Харків: ХНУРЕ, 2024. С. 241-143.
62. Попова А. В., Васильцова Н. В. Оцінювання успішності управління змінами в довгострокових ІТ-проектах // XXVI International scientific and practical conference «Theoretical and Practical Aspects of Modern Research» (June 5-7, 2024). Ottawa, Canada, 2024. С. 120-124.

63. Методичні вказівки до організації виконання та захисту кваліфікаційної роботи за першим (бакалаврським) рівнем вищої освіти спеціальності 122 Комп'ютерні науки за освітньою програмою «Інформаційні технології управління» для студентів усіх форм навчання / Упоряд.: К.Е. Петров, А.В. Міхнова, М.С. Кудрявцева, М.В. Євланов, Т.І. Борисенко. Харків: ХНУРЕ, 2023. 68 с.

64. ДСТУ 8302:2015. Бібліографічне посилання. Загальні положення та правила складання. / Видання офіційне. – К.: ДП «УкрНДНЦ», 2016 – 20 с.

65. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання, Чинний від 22.06.2015. Київ: ДП «УкрНДНЦ», 2016, 26 с.