

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 моделей та методів управління змінами
при реалізації ІТ-проектів
(тема)

Виконав:
студент 2 курсу, групи УШГІТм-22-2

Сурков Єгор Миколайович
(прізвище, ім'я, по батькові)

Спеціальність 122 Комп'ютерні науки

(код і повна назва спеціальності)

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

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

Керівник зав. каф. ІУС Костянтин ПЕТРОВ
(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 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Суркову Єгору Миколайовичу
 (прізвище, ім'я, по батькові)

1. Тема роботи Дослідження моделей та методів управління змінами при реалізації ІТ-проектів

затверджена наказом університету від 01 квітня 2024 р. № 258Ст

2. Термін подання студентом роботи до екзаменаційної комісії 05 червня 2024 р.


3. Вихідні дані до роботи науково-технічна література, публікації та інтернет-ресурси, що стосуються теми кваліфікаційної роботи; матеріали звіту з науково-дослідної практики


4. Перелік питань, що потрібно опрацювати в роботі аналіз предметної області; формування проблеми щодо вдосконалення; огляд існуючих моделей, методів та технологій управління змінами при реалізації ІТ-проектів; модифікація моделі управління змінами в ІТ-проекті; проведення експериментальної перевірки розробленої моделі; аналіз отриманих результатів.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	01.04.2024	виконано
2	Пошук наукових публікацій, літературних джерел та інформаційних ресурсів у наукових виданнях і Інтернеті	02.04.2024 – 05.04.2024	виконано
3	Аналіз сучасного стану об'єкта дослідження	06.04.2024 – 10.04.2024	виконано
4	Формування проблеми щодо вдосконалення з урахуванням її актуальності	11.04.2024	виконано
5	Огляд існуючих підходів, моделей, методів та технологій управління змінами при реалізації ІТ-проектів	12.04.2024 – 15.04.2024	виконано
6	Постановка задачі дослідження	16.04.2024	виконано
7	Розробка комбінованої моделі управління змінами в ІТ-проектах	17.04.2024 – 22.04.2024	виконано
8	Практична апробація розробленої моделі	23.04.2024 – 10.05.2024	виконано
9	Аналіз отриманих теоретичних та практичних результатів використання розробленої моделі	11.05.2024 – 13.05.2024	виконано
10	Оформлення пояснювальної записки	14.05.2024 – 19.05.2024	виконано
11	Підготовка презентації	20.05.2024 – 21.05.2024	виконано
12	Надання роботи для перевірки на плагіат	31.05.2024	виконано
13	Надання роботи на підпис науковому керівникові	02.06.2024	виконано
14	Надання роботи на рецензування	03.06.2024	виконано
15	Надання підписаної завідувачем кафедри роботи в ЕК	05.06.2024	виконано
16	Захист кваліфікаційної роботи	07.06.2024	виконано

Дата видачі завдання 01 квітня 2024 р.

Студент 
(підпис)

Керівник роботи  зав. каф. ІУС Костянтин ПЕТРОВ
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи містить: 94 сторінки, 7 рисунків, 6 таблиць, 35 джерел, 1 додаток.

ЗАПИТ НА ЗМІНИ, ІТ-ПРОЄКТ, МЕТОДОЛОГІЯ 8D, МОДЕЛЬ КОТТЕРА, ПОРІВНЯЛЬНИЙ АНАЛІЗ, ПРОЦЕС УПРАВЛІННЯ ЗМІНАМИ, LEAN CHANGE MANAGEMENT, MICROSOFT PROJECT.

Об'єктом досліджень науково-дослідної роботи є процес управління змінами при реалізації ІТ-проектів.

Предметом досліджень є моделі та методи управління змінами в ІТ-проектах, їх вплив на ефективність реалізації проектів.

Метою науково-дослідної роботи є дослідження та проведення порівняльного аналізу методів і моделей управління змінами при виконанні ІТ-проектів, а також розробка власної моделі управління змінами, яка б враховувала якомога більше переваг і вирішувала недоліки існуючих.

Для досягнення поставленої мети було проведено детальний аналіз складових частин процесу управління змінами в ІТ-проектах, досліджено критерії, що використовуються у цьому процесі, та визначено їх вплив на ефективність виконання проекту. Виконано аналіз існуючих моделей і методів управління змінами при реалізації ІТ-проектів, а саме методології 8D, моделі Коттера і моделі Lean Change Management, включаючи оцінку ефективності кожної з них.

На підставі проведеного аналізу переваг і недоліків існуючих моделей розроблено власну комбіновану модель управління змінами, що враховує сучасні вимоги та специфіку ІТ-проектів. Проведено практичну перевірку ефективності розробленої моделі, а також здійснено аналіз та сформульовано рекомендації щодо впровадження та використання отриманих наукових результатів.

ABSTRACT

The explanatory note to the qualification work contains: 94 pages, 7 figures, 6 tables, 35 sources, 1 appendice.

CHANGE MANAGEMENT PROCESS, COMPARATIVE ANALYSIS, COTTER MODEL, IT-PROJECT, LEAN CHANGE MANAGEMENT, MICROSOFT PROJECT, REQUEST FOR CHANGE, 8D METHODOLOGY.

The object of scientific research work is the process of change management during the implementation of IT projects.

The subject of research are models and methods of change management in IT projects, their impact on the effectiveness of project implementation.

The purpose of the research work is to study and conduct a comparative analysis of methods and models of change management in the implementation of IT projects, as well as the development of one's own model of change management, which would take into account as many advantages as possible and solve the shortcomings of existing ones.

To achieve the goal, a detailed analysis of the constituent parts of the change management process in IT projects was conducted, the criteria used in this process were investigated, and their impact on the effectiveness of project implementation was determined. The analysis of existing models and methods of change management in the implementation of IT projects, namely the 8D methodology, the Kotter model and the Lean Change Management model, including the evaluation of the effectiveness of each of them, was performed.

Based on the analysis of the advantages and disadvantages of the existing models, an own combined change management model was developed, which takes into account modern requirements and the specifics of IT projects. A practical check of the effectiveness of the developed model was carried out, as well as an analysis was carried out and recommendations were formulated for the implementation and use of the obtained scientific results.

ЗМІСТ

Скорочення та умовні позначки	7
Вступ	8
1 Аналіз предметної області та постановка задачі дослідження	10
1.1 Огляд процесу управління ІТ-проєктом	10
1.2 Аналіз процесу управління змінами в рамках управління ІТ-проєктом	12
1.3 Опис критеріїв, які використовуються в процесі управлінні змінами, і аналіз їх впливу на ІТ-проєкт	21
1.4 Аналіз існуючих моделей, методів та технологій управління змінами в процесі управління ІТ-проєктом	24
1.4.1 Аналіз моделей і методів управління змінами в ІТ-проєктах	24
1.4.2 Аналіз технологій та інструментів управління змінами в ІТ-проєктах	30
1.5 Постановка задачі дослідження	33
2 Розробка комбінованої моделі управління змінами в ІТ-проєктах	35
3 Експериментальна перевірка працездатності та оцінка ефективності використання розробленої моделі	40
3.1 Опис умов проведення тестування	40
3.1.1 Опис ІТ-проєкту	41
3.1.2 Опис змін	45
3.2 Оцінка ефективності методології 8D	48
3.3 Оцінка ефективності моделі Коттера	52
3.4 Оцінка ефективності моделі Lean Change Management	55
3.5 Апробація комбінованої моделі управління змінами	57
4 Аналіз отриманих теоретичних та практичних результатів використання розробленої моделі	62
Висновки	65
Перелік джерел посилання	67
Додаток А Графічний матеріал	71

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

IT – інформаційні технології

ПЗ – програмне забезпечення

CAB – change advisory board

CCB – change control board

ITSM – information technology service management

LCM – lean change management

PMBOK Guide – a guide to the project management body of knowledge

PMI – Project Management Institute

RFC – request for change

SLA – service level agreement

ВСТУП

Інновації та технологічні розробки відіграють ключову роль у динамічному бізнес-середовищі сьогодення. З огляду на стрімкий розвиток інформаційних технологій та збільшення складності ІТ-проектів, компанії змушені адаптуватися до нових реалій, щоб залишатися конкурентоспроможними.

Успішність ІТ-проектів значною мірою залежить від ефективності впровадження процесу управління змінами. В умовах мінливого середовища, жорсткої конкуренції та постійного вдосконалення здатність швидко адаптуватися до змін та впроваджувати їх стає вирішальним фактором для досягнення стратегічних цілей проекту. Відповідно, використання ефективних методів і моделей управління змінами є надзвичайно важливим для забезпечення успішного завершення ІТ-проектів у визначені терміни, в рамках бюджету та згідно із встановленими вимогами.

Актуальність цього питання зростає через кілька ключових факторів. По-перше, ІТ-проекти часто впливають на основні бізнес-процеси та організаційну структуру компаній. По-друге, належне управління змінами допомагає мінімізувати ризики, пов'язані з опором співробітників, затримками у виконанні завдань та перевищенням бюджетів. По-третє, сучасні методи управління змінами орієнтовані на гнучкість та адаптивність, що є критично важливим в умовах швидкоплинних змін і невизначеності.

Об'єктом дослідження є процес управління змінами в ІТ-проектах.

Предметом дослідження є моделі та методи управління змінами в ІТ-проектах, їх вплив на ефективність реалізації проектів.

Метою науково-дослідної роботи є дослідження та проведення порівняльного аналізу методів і моделей управління змінами при виконанні ІТ-проектів, а також розробка власної моделі управління змінами, яка б враховувала якомога більше переваг і вирішувала недоліки існуючих.

Для досягнення мети необхідно вирішити наступні завдання:

- провести детальний аналіз складових частин процесу управління змінами в ІТ-проєктах;
- дослідити критерії, що використовуються в процесі управління змінами, і визначити їх вплив на ефективність виконання проєкту;
- провести аналіз існуючих моделей і методів управління змінами при реалізації ІТ-проєктів, включаючи оцінку ефективності кожного, можливості автоматизації та інтеграції з іншими процесами управління проєктами;
- розробити власну модель управління змінами в ІТ-проєкті на підставі виконаного аналізу переваг і недоліків існуючих та таку, що враховуватиме сучасні вимоги і специфіку ІТ-проєктів;
- перевірити ефективність розробленої моделі на практиці;
- провести аналіз і сформулювати рекомендації щодо впровадження та використання отриманих наукових результатів.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

1.1 Огляд процесу управління IT-проектом

У сучасному світі роль інформаційних технологій набуває все більшої ваги, оскільки суспільство постійно стикається зі складними задачами, для вирішення яких необхідно ефективно впроваджувати інноваційні технологічні рішення. Це відноситься не лише до розробки програмного забезпечення чи інформаційних систем, але й до надання широкого спектру IT-послуг, включаючи консалтинг, розробку інфраструктурних рішень, кібербезпеки, аналітики даних та ін., спрямованого на задоволення потреб клієнтів.

Сьогодні IT-рішення застосовуються майже в усіх сферах життя: медицині, логістиці, фінансах, освіті, наукових дослідженнях. Ці рішення зазвичай впроваджуються через реалізацію проєктів у сфері IT.

Згідно 7-го видання стандарту PMBOK Guide [1] Інституту з управління проєктами в США, проєкт – це тимчасова спроба створити унікальний продукт, послугу або результат. Як правило, він включає в себе різноманітні види діяльності, які спільно характеризуються такими загальними ознаками, як:

- спрямованість на досягнення конкретних цілей;
- координоване виконання численних взаємозалежних дій;
- обмеженість у ресурсах і часі.

Щоб забезпечити стійкі й позитивні результати в проєкті, необхідно мати чітко визначені цілі та методи контролю за їх досягненням. Для цього застосовується загальний процес управління проєктами – процес керівництва та координації ресурсів протягом усього життєвого циклу проєкту, в якому використовуються сучасні методи і техніки досягнення визначених результатів, таких як обсяг робіт, вартість, терміни, якість, а також задоволення інтересів всіх учасників проєкту [2].

Зазвичай процес управління проектами поділяється на такі етапи:

- ініціація;
- планування;
- реалізація;
- моніторинг і контроль;
- завершення.

На етапі ініціації формується основа ІТ-проекту, яка включає в себе встановлення чіткої мети, конкретизацію обсягу проекту, визначення точних цілей, яких потрібно досягти, ідентифікацію всіх зацікавлених сторін, аналіз їх очікувань і потреб. Результатами етапу ініціації є створення концепції та розробка статуту проекту, що встановлює основні принципи та правила його функціонування.

На етапі планування визначають стратегічні дії, необхідні для досягнення цілей проекту. Вони охоплюють ретельну розробку планів, встановлення послідовності та методики виконання робіт, розподіл завдань між учасниками команди, визначення термінів виконання, необхідних ресурсів і бюджету.

Етап реалізації характеризується виконанням запланованих дій та завдань, які були визначені на попередньому етапі. Ключовими аспектами цього етапу є координація роботи команди і контроль виконання завдань за планами.

На етапі моніторингу і контролю здійснюється систематичний збір даних і аналіз ходу виконання проекту, контроль за використанням ресурсів, бюджету, а також своєчасне виявлення проблем і вжиття заходів для коригування ходу виконання проекту, таких як зміна пріоритетів, перерозподіл ресурсів, перегляд бюджету.

Останнім етапом є завершення проекту, де проводиться детальний аналіз і оцінка результатів, складання звітності та документації, а також передача результатів робіт замовнику.

Управління проектом в ІТ включає в себе такі аспекти [3]:

- управління змістом;
- управління якістю;

- управління розкладом;
- управління вартістю;
- управління контрактами;
- управління ресурсами;
- управління зацікавленими сторонами;
- управління комунікаціями;
- управління інтеграцією;
- управління змінами;
- управління ризиками.

У рамках кваліфікаційної роботи буде детально досліджено процес управління змінами при реалізації ІТ-проектів.

1.2 Аналіз процесу управління змінами в рамках управління ІТ-проектом

Швидкі темпи технологічного розвитку та постійні зміни вимог ринку призвели до необхідності вдосконалення стратегій та методів управління проектами в галузі ІТ. Зокрема, процес управління змінами виявляється ключовим фактором успіху ІТ-проектів, оскільки якісне його впровадження дозволяє пристосовуватися до непередбачуваних ситуацій у швидкозмінному технологічному середовищі.

Згідно з Prosci [4], управління змінами – це застосування структурованого процесу і набору інструментів людьми для керування змінами та досягнення бажаного результату.

Управління змінами за РМІ – це комплексний, циклічний і структурований підхід для переходу окремих осіб, груп і організацій із поточного стану в майбутній із передбачуваними перевагами для бізнесу.

Обидва підходи мають як спільні, так і відмінні риси. Їх об'єднує структурованість, де чітко визначені всі кроки й етапи. Додатково, кожен

пропонує власний набір інструментів, які допомагають керувати і впроваджувати зміни досягаючи бажаних результатів. Основна відмінність полягає в центральному елементі та фокусі кожного підходу. Prosci ставить людей в центр свого підходу, де керування змінами відбувається через співпрацю та міжособисту взаємодію. PMI ж пропонує комплексний підхід, який враховує не лише людей, але й проекти, організації, і більше зосереджується на визначенні конкретних переваг для бізнесу та впровадженні змін, які сприятимуть досягненню первинних цілей проекту.

Під час реалізації проекту можуть відбуватися різноманітні зміни, що впливають на його різні аспекти, зокрема предметну область, її конфігурацію, час, вартість, якість, ризики, контракти, поставки, людські ресурси, комунікації та процеси управління проектом на будь-якому етапі життєвого циклу [5].

Зміни можуть бути ініційовані з будь-яких причин: зміни в стратегії компанії, внутрішні політичні або зовнішні економічні фактори. Також неоднозначні або неповні первинні вимоги часто призводять до неочікуваних проблем у реалізації. Ідентифікація нових ризиків, зміна їх пріоритету, зміни в законодавстві або стандартах можуть змусити проект змінити підходи або функціональні можливості. Нові фінансові, технічні або ресурсні обмеження можуть вимагати перерозподілу ресурсів або вплинути на обсяг робіт.

У процесі управління змінами в IT-проекті беруть участь різні зацікавлені сторони, кожна з яких може виступати ініціатором змін або сприяти їх успішному впровадженню. Загальний результат управління змінами залежить від співпраці та ефективної взаємодії між усіма учасниками процесу:

а) менеджера проекту, який несе відповідальність за планування, координацію та виконання проекту, визначає потреби в змінах, розробляє стратегії їх впровадження та відповідає за виконання планів змін;

б) замовника – організації або особи, яка ініціює проект (визначає бізнес-потреби, очікувані результати та критерії успіху);

в) спонсора – джерела фінансування проекту;

г) користувачів, які беруть участь у тестуванні і для надання зворотнього зв'язку;

г) команди проекту, що складається зі спеціалістів: програмістів, аналітиків, тестувальників, дизайнерів тощо;

д) експертної комісії за проектом;

е) менеджера зі змін (у великих проектах);

є) інших стейкхолдерів, які мають хоч якийсь вплив на проект.

Складовими процесу управління змінами є:

- ідентифікація змін;
- аналіз і оцінка впливу на проект;
- планування;
- погодження на впровадження;
- реалізація змін;
- оцінка і закріплення результатів.

На початковому етапі ідентифікації змін відбувається виявлення потенційних змін, що можуть вплинути на хід проекту. На цій стадії починається формування офіційного запиту на зміни (RFC – request for change) [6] в якому документально фіксуються деталі пропонованої зміни.

Наступним кроком є проведення ґрунтового аналізу та оцінки впливу виявлених змін на проект. Цей етап передбачає всебічний розгляд технічних, фінансових аспектів запропонованих змін, оцінки потенційних ризиків, вартості, тривалості, необхідних ресурсів та інших факторів, пов'язаних зі змінами. Результати такого аналізу слугують підставою для прийняття обґрунтованого рішення щодо доцільності впровадження змін.

За умови схвалення змін, розробляється детальний план їх впровадження. Цей план має враховувати всі необхідні коригуючі заходи, ресурси, графік та бюджет для успішної реалізації кожної зміни. Крім того, план повинен містити ефективну стратегію комунікації для інформування всіх зацікавлених сторін про поточний статус впровадження та впливу змін на проект.

Перед безпосереднім впровадженням змін, вони мають пройти етап погодження з відповідними сторонами, такими як керівництво проєкту, замовник, консультативна рада зі змін (CAB – Change Advisory Board) [7] або рада контролю змін (CCB – Change Control Board) [8]. На цій стадії ретельно аналізуються всі деталі плану впровадження змін, і приймається остаточне рішення про їх реалізацію.

Після схвалення змін відбувається їх впровадження відповідно до затвердженого плану. Цей етап може включати внесення змін до коду, документації, процесів, інфраструктури або інших складових проєкту. Під час реалізації змін, критично важливо мати належний рівень тестування, контролю якості та комунікації з зацікавленими сторонами.

Заключним кроком є оцінка та закріплення результатів, аналіз успішності та ефективності впроваджених змін. У разі позитивних результатів, зміни закріплюються в проєкті та документуються для подальшого використання. Однак, у випадку виявлення проблем або невідповідностей, можуть бути ініційовані додаткові зміни або коригувальні дії.

Управління змінами має мету розв'язати проблеми та завдання, що пов'язані із забезпеченням захисту проєкту від потенційного негативного впливу зовнішніх та внутрішніх факторів і контролю за ефективним впровадженням узгоджених змін. Як було зазначено вище, цей процес реалізовано через запити. Запит на зміну в ІТ-проєкті визначається як запит на модифікацію вже схвалених елементів проєкту, включаючи вимоги, графіки, бюджет та інші аспекти. Згідно з методологією ITSM, його складовими є:

а) основна інформація:

- 1) організація;
- 2) відділ;
- 3) підрозділ;
- 4) унікальний ідентифікатор запиту на зміну;

б) інформація про запитувача:

- 1) дата подання;

- 2) дата розгляду;
 - 3) ім'я запитувача;
 - 4) електронна пошта;
 - 5) телефон;
- в) деталі зміни:
- 1) короткий опис;
 - 2) детальний опис;
 - 3) обґрунтування;
 - 4) додатки;
- г) аналіз зміни:
- 1) категорія;
 - 2) тип;
 - 3) терміновість;
 - 4) пріоритет;
 - 5) необхідність залучення учасників проекту та врахування інтересів усіх зацікавлених сторін;
 - 6) початковий рейтинг ризику;
 - 7) короткострокова/довгострокова вигода;
 - 8) переваги та недоліки;
- г) деталі щодо погоджувального процесу та наявності САВ/ССВ;
- д) аналіз ризиків:
- 1) тип ризику;
 - 2) вплив ризику;
 - 3) ймовірність;
 - 4) наслідки ризику;
- е) планування:
- 1) план впровадження;
 - 2) план відновлення;
 - 3) план тестування;

є) фінансові деталі:

- 1) відносна вартість;
- 2) оцінене зусилля в людино-днях;
- 3) пов'язані SLA (service level agreement);
- 4) очікувана вартість);

ж) пов'язані інциденти, проблеми, зміни;

з) деталі розкладу:

- 1) запланована дата початку і дата завершення;
- 2) наявність CAB/CCB;
- 3) фактична дата початку і дата завершення;
- 4) рекомендації CAB/CCB;

и) огляд зміни після впровадження;

і) інформація про закриття зміни

- 1) код закриття;
- 2) примітки.

Існує три основні типи змін [9]:

- стандартні;
- нормальні;
- екстрені.

Стандартні зміни являють собою регулярні, передбачувані зміни, які здійснюються відповідно до встановлених операційних процедур із використанням заздалегідь підготовлених шаблонів, що дозволяє значно скоротити кількість часу, витраченого на зворотні електронні листи, і забезпечує простий спосіб відстеження стану запитів на зміни. Оскільки вони заздалегідь визначені та схвалені, ці зміни мають низький рівень ризику та впливу. Зазвичай для їхнього впровадження не потрібно повторного схвалення після початкового затвердження, якщо тіло зміни не змінилось. Таким чином, стандартні зміни значно спрощують процес управління. Прикладом стандартної зміни може бути оновлення програмного забезпечення або впровадження регулярного резервного копіювання даних.

На рис. 1.1 представлено приклад шаблону форми стандартного запиту на зміни (RFC).

CHANGE REQUEST FORM		
Change Description		
Project Name:	Change Name:	Number:
Requested By:	Contact:	Date:
Description of Change:		
Reason for Change:		
Priority [Circle One]: 1. High 2. Medium 3. Low		
Impact on Deliverables:		
Impact of Not Responding to Change (and Reason Why):		
Date Needed:	Approval of Request:	Date:
Change Impact		
Tasks/Scope Affected:		
Cost Evaluation:		
Risk Evaluation:		
Quality Evaluation:		
Additional Resources:		
Duration:		
Additional Effort:		
Impact on Deadline:		
Alternative and Recommendations:		
Comments:		
Sign Offs		
[Circle One]: 1. Accepted 2. Deferred 3. Rejected 4. More Info Requested		
Comments:		
Project Manager Signature:	Date:	
Decision Maker Signature:	Date:	

Рисунок 1.1 – Приклад шаблону для стандартного запиту на зміни (Standard RFC)

Нормальні зміни охоплюють усі зміни, які не є ані стандартними, ані екстремними, і потребують проходження повного процесу управління змінами. Нормальні зміни поділяються на незначні та основні. Незначні зміни мають низький рівень впливу та ризику, але все одно вимагають схвалення САВ.

Основні зміни, навпаки, мають високий рівень ризику та значний вплив на систему або бізнес-процеси, що може призвести до серйозних порушень у роботі, якщо вони не будуть належним чином оцінені та реалізовані. Прикладом звичайної зміни може бути зміна функціональності програмного забезпечення або міграція центру обробки даних на нову платформу.

Екстрені зміни є невідкладними і потребують негайного вирішення для мінімізації негативних наслідків для бізнесу. Вони характеризуються високим рівнем ризику та значним впливом на бізнес-операції, що може призвести до значних перебоїв у роботі. Екстрені зміни виникають раптово і потребують швидкого схвалення та авторизації, часто з боку спеціальної ради з надзвичайних змін (ЕСАВ/ЕССВ). Після впровадження такі зміни підлягають ретельному перегляду та документації для оцінки потенційних ризиків у майбутньому. Прикладом екстреної зміни може бути усунення порушення безпеки після кібератаки або термінове виправлення критичної помилки в програмному забезпеченні, яка спричинила збої у роботі системи.

На рис. 1.2 зображено шаблони нормального і екстреного запитів на зміни за методологією ITIL [10].

The image displays two ITIL Change Request Form templates side-by-side. The left template is for a 'Normal RFC' and the right is for an 'Emergency RFC'.

Normal RFC Template (Left):

- Header: <Logo>, <Company Name>, Normal
- Title: Change Request Form
- Fields: Organization, Department, Section, Sheet: 1 of 6
- Change Request No. field
- Section: Normal RFC
- Section: Change Requester Details (Date submitted, Date required, Requester name, Email, Phone, Commission, Department, Manager's name, Manager's email, Manager's phone)
- Section: Basic details (Short Description, Detailed description, Justification, Location, Attachments)
- Section: Change Management Analysis (Category, Type, Item, Impact, Urgency, Priority, Initial Risk Rating, Short term benefits, Long term benefits, Pros and Cons)
- Section: CAB details (TCAB Required?, DCAB Required?)
- Section: Evaluation (Who RAISED the change?, What is the REASON for the change?)
- Footer: Document No., Revision No., Sheet: 1 of 6, Issue Date: xx-xxx-xx

Emergency RFC Template (Right):

- Title: Change Request Form
- Fields: What is the RETURN required from the change?, What are the RISKS involved in the change?, What RESOURCES are required to deliver the change?, Who is RESPONSIBLE for the build, test and implementation of the change?, What is the RELATIONSHIP between this change and other changes?
- Section: Risk analysis (Type of risk, Risk Impact, Likelihood, Probability, Risk Consequences)
- Section: Planning details (Implementation plan, Remediation plan, Backout plan, Test plan)
- Section: Financial details (Relative cost, Estimated effort in man days, SLAs associated, Approximate cost)
- Section: Associated tickets (Associated Incidents, Associated Problems, Associated Change)
- Section: Schedule Details (Planned start date, Actual start date, Planned end date, Actual end date, CAB required, CAB Recommendations, CAB date)
- Section: PIR
- Section: Change Closure Information (Closure code, Closure notes)
- Footer: Document No., Revision No., Sheet: 2 of 6, Issue Date: xx-xxx-xx

Рисунок 1.2 – Нормальний та екстрений запит на зміни за методологією ITIL

Change Request Form

Change request form is the medium through which the change initiator can describe the details of the proposed change.

*Important details to be captured in **Normal Change tickets** are:*

RFC Number: a unique ID registered for the change
Change Description: the description of the change
Change Location: the location where the change will be implemented
Change Requester: the person who requested the change request/RFC
Change Analyst: the name of the change analyst who will analyze the change request/RFC
Change Requested Date: the date on which the change was requested
Change Triggered By: defines the sources that triggered the change like legal requirements, business requirements, etc.
Change Classification: the classification of the change like Normal, Standard, and Emergency
Category: the category of the change
Type: the type of change
Item: item of the change
Assignment group: The group assigned to own and possibly implement the Change Request
Risk analysis: describes the risks associated with the change
Business Case: the plan which defines the business justification, benefits, and resources needed
Rollback Plan: the description of the rollback plan
Risk analysis: the description of the risk analysis
Remediation Plan: the description of the remediation plan
Impacting Services: the services that will be impacted by the change
Impacting CIs: the CIs that will be impacted by the change
Relative Benefit of Implementing the Change: the benefit of implementing change
Relative Cost: This should define the relative costs
Estimated Effort in Man Days or Hours: Man days or hours
Change Approval/Rejected Date: the date and time when the change was approved/ rejected by CAB
CAB Decision: a decision made by the CAB
CAB Comments: comments given by the CAB
ECAB Decision: decision made by the ECAB
ECAB comments: comments given by the ECAB
Change Manager: name of the change manager
Impact: The number of people that will be affected by change
Urgency: how soon the change has to be implemented
Priority: It will be based on impact and urgency
SLAs Associated: SLAs associated with change management
SLA Target Date and Time: date and time when the SLAs will be breached with respect to the change
Major Change Review: This determines if it's a major change
Major Change Justification: This defines the business justification and why it should be treated as a major change
Associated Incidents: the details of the incident tickets that are associated with this change
Associated Problems: the details of the problem tickets that are associated with this change
SLAs Breach Details: the description why the SLAs were breached, and by how many minutes or hours did we breach the SLAs.
PIR: defines the lessons learnt

Document No: _____ Sheet: 3 of 6
 Revision No: _____ Issue Date: xx-xxx-xx

Change Request Form

Change Request No.: _____

Emergency RFC

Change Requester Details			
Date submitted:	Company:		
Date requested:	Department:		
Requester name:	Manager's name		
Email:	Manager's email:		
Phone:	Manager's phone:		

Basic details			
Short Description:			
Detailed description:			
Justification:			
Location:			
Attachments:			

Change Management Analysis			
Category:	Change Manager:		
Type:	Change Builder:		
Item:	Change Tester:		
Impact:	Assignment group:		
Urgency:	CAB members needed:		
Priority:	CI's involved:		
Initial Risk Rating:	Impacted Services:		
Short term benefits:			
Long term benefits:			
Pros and Cons:			

CAB details			
ECAB Required?			

Risk analysis			
Type of risk: [social, financial, organizational, external]			
Risk Impact: [trivial, minor, moderate, major, and catastrophic]			
Likelihood: [1,2,3,4]			
Probability: [1,2,3,4]			
Risk Consequences:			

Document No: _____ Sheet: 4 of 6
 Revision No: _____ Issue Date: xx-xxx-xx

Рисунок 1.2, аркуш 2

Change Request Form

Planning details			
Implementation plan			
Remediation plan			
Backout plan			
Test plan			

Associated tickets			
Associated Incidents			
Associated Problems			
Associated Change			

Scheduling Details			
Planned start date:	Actual start date:		
Planned end date:	Actual end date:		
CAB members:	CAB Recommendations:		
CAB date:			

PIR			

Change Closure Information			
Closure code:			
Closure notes:			

Document No: _____ Sheet: 5 of 6
 Revision No: _____ Issue Date: xx-xxx-xx

Change Request Form

*Important details to be captured in **Emergency Change tickets** are:*

RFC Number: a unique ID registered for the change
Change Description: the description of the change
Change Location: the location where the change will be implemented
Change Requester: the person who requested the change request/RFC
Change Analyst: the name of the change analyst who will analyze the change request/RFC
Change Requested Date: the date on which the change was requested
Change Triggered By: defines the sources that triggered the change like legal requirements, business requirements, etc.
Change Classification: the classification of the change like Normal, Standard, and Emergency
Category: the category of the change
Type: the type of change
Item: item of the change
Risk analysis: describes the risks associated with the change
Business Case: the plan which defines the business justification, benefits, and resources needed
Rollback Plan: the description of the rollback plan
Risk analysis: the description of the risk analysis
Remediation Plan: the description of the remediation plan
Impacting Services: the services that will be impacted by the change
Impacting CIs: the CIs that will be impacted by the change
Change Approval/Rejected Date: the date and time when the change was approved/ rejected by CAB
ECAB Decision: decision made by the ECAB
ECAB comments: comments given by the ECAB
Change Manager: name of the change manager
Impact: The number of people that will be affected by change
Urgency: how soon the change has to be implemented
Priority: It will be based on impact and urgency
SLAs Associated: SLAs associated with change management
SLA Target Date and Time: date and time when the SLAs will be breached with respect to the change
Associated Incidents: the details of the incident tickets that are associated with this change
Associated Problems: the details of the problem tickets that are associated with this change
SLAs Breach Details: the description why the SLAs were breached, and by how many minutes or hours did we breach the SLAs.
PIR: defines the lessons learnt

Document No: _____ Sheet: 6 of 6
 Revision No: _____ Issue Date: xx-xxx-xx

Рисунок 1.2, аркуш 3

1.3 Опис критеріїв, які використовуються в процесі управлінні змінами, і аналіз їх впливу на ІТ-проект

Управління змінами в ІТ-проектах є критичним елементом для успішної реалізації проекту. Критерії, що використовуються в процесі управління змінами, можуть варіюватися в залежності від конкретного проекту та його потреб. Нижче буде розглянуто основні критерії, які можуть бути використані як ключові принципи ефективного управління змінами в ІТ-проектах.

1. **Пріоритет зміни.** Визначає ступінь критичності для успішного виконання проекту. Зміни з високим пріоритетом, як правило, є необхідними для досягнення ключових цілей проекту і задоволення критичних вимог замовника. Затримка або відхилення таких змін створить умови для зниження якості кінцевого продукту. З іншого боку, зміни з низьким пріоритетом, як правило, не є критичними для основних цілей: їх можна відкласти або навіть відхилити, якщо вони створюють занадто великі ризики для вартості, термінів або якості проекту.

2. **Складність зміни.** Має прямий вплив на вартість і тривалість виконання проекту. Чим складніша зміна, тим більше ресурсів (часу, коштів та зусиль) знадобиться для її успішної реалізації. Крім того, якщо складність зміни недооцінено, це може призвести до неможливості її якісного впровадження аж до повного відхилення зміни.

3. **Відповідність вимогам і стратегічним цілям проекту.** Надає можливість слідувати первинним планам, але у разі виникнення необхідності внесення змін доведеться коригувати ці плани, щоб зберегти узгодженість із новими вимогами та забезпечити досягнення стратегічних цілей [11].

4. **Сумісність.** Показує чи призведе впровадження зміни до виникнення помилок, конфліктів або втрати функціональності в існуючій системі чи процесах. Цей критерій є важливим через те, що він допомагає оцінити потенційні ризики врахувавши вплив зміни на різні компоненти системи, взаємодію між ними та загальну цілісність і узгодженість.

5. Вартість. Залежить від складності зміни, обсягу роботи, необхідних ресурсів та часу. Велика вартість може призвести до виходу за рамки затвердженого бюджету, вимушеного скорочення ресурсів, зменшення обсягу робіт або функціональності кінцевого продукту, а також спричинити суттєві затримки у графіку реалізації. Наприклад, якщо до проєкту вноситься значна зміна, яка потребує додаткових розробників, це обов'язково призведе до підвищення вартості проєкту. Для мінімізації вартості змін використовують різні методи, такі як аналіз витрат та результатів, моделювання або метод нарахування.

6. Ресурси. Належна кількість, раціональне використання і ефективна координація значно знижують ризик затримок у графіку, перевищення бюджету та погіршення якості продукту, що розробляється. У результаті, процес впровадження змін стає значно простішим і більш керованим. Крім того, належне управління ресурсами сприяє підвищенню продуктивності команди, покращенню комунікації та координації зусиль [12], що, в свою чергу, зміцнює загальну стійкість проєкту до змін і підвищує ймовірність його успішного завершення відповідно до встановлених цілей і вимог.

7. Терміни. Мають прямий зв'язок зі складністю, наявністю необхідних ресурсів і бюджету, а також якістю. Будь-які затримки у впровадженні змін можуть спричинити зміщення графіку виконання робіт, порушення взаємозв'язків між задачами і необхідність перепланування ресурсів [13]. Як наслідок, це призводить до змін у планах проєкту і може потребувати залучення додаткового фінансування.

8. Якість. Зазвичай є ключовим елементом проєкту через необхідність досягнення бізнес-цілей і задоволення вимог замовника. Зміни, які негативно впливають на якість, можуть призвести до невдоволеності клієнтів, порушення функціональності, зниження надійності і помилок кінцевого продукту, або втрати репутації як ІТ-компанії-виконавця, так і компанії-замовника [14]. Для мінімізації негативного впливу змін на якість важливо застосовувати відповідні практики забезпечення якості, такі як тестування, валідація вимог, контроль версій та інші. Тому процес управління змінами в ІТ-проєкті повинен передбачати ретельний

контроль якості на кожному етапі, щоб забезпечити відповідність кінцевого продукту встановленим стандартам та очікуванням зацікавлених сторін.

9. Ефективність. Ґрунтується на таких метриках та показниках: час виконання проєкту, витрати, коефіцієнт повернення інвестицій, рівень задоволеності клієнтів і відповідності продукту вимогам, кількість помилок, продуктивність команди [15]. Може включати оптимізацію ресурсів, вдосконалення процесів або впровадження новітніх технологій, що дозволяє досягти стратегічних цілей з меншими витратами часу та зусиль. Будь-яка зміна, яка покращує ефективність виконання проєкту або кінцевого результату проєкту, є значущою та може позитивно позначитися на продуктивності та конкурентоспроможності проєкту.

10. Ризики. Розрізняють технічні (збої у роботі систем), організаційні (опір змінам з боку користувачів), фінансові (перевищення бюджету), зовнішні (зміни у стандартах або законодавстві) і ризики управління (застосування нових підходів до управління проєктом) [16]. З появою змін їх характер та ступінь впливу можуть істотно змінюватися [17]. Залежність впливу ризиків на вартість змін в проєкті представлено на рис. 1.3. За умов відсутності ефективної стратегії управління ризиками навіть незначні зміни можуть мати каскадний вплив, що в підсумку призведе до значних збитків, критичних відхилень від запланованих цілей або навіть повного провалу проєкту.

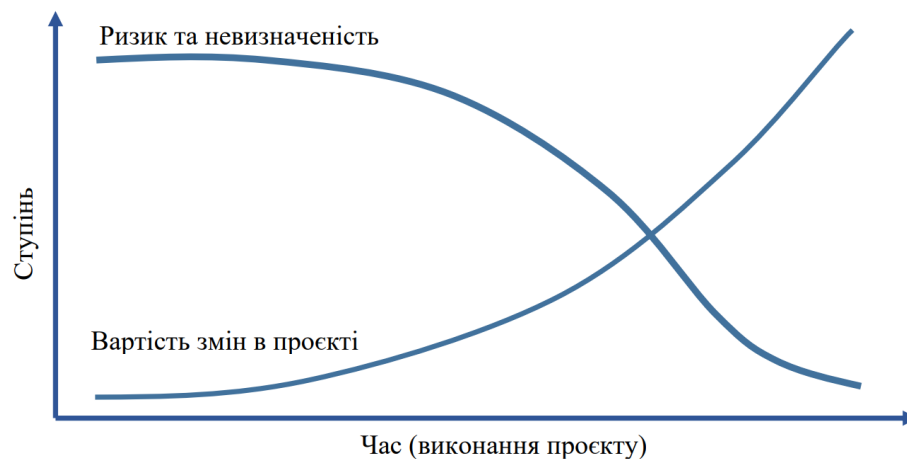


Рисунок 1.3 – Залежність ступеню впливу ризиків на вартість змін в ІТ-проєктах

Ці критерії можна використати як ключові принципи ефективного управління змінами в ІТ-проєктах, які допомагають забезпечити успішну реалізацію проєкту та досягти поставлених цілей.

1.4 Аналіз існуючих моделей, методів та технологій управління змінами в процесі управління ІТ-проєктом

1.4.1 Аналіз моделей і методів управління змінами в ІТ-проєктах

У сучасному інформаційному середовищі, яке характеризується стрімкими темпами технологічного розвитку, вибір моделей, методів і технологій управління змінами має вирішальне значення для забезпечення якості й ефективності управління ІТ-проєктами.

Одним з ключових підходів управління змінами є методологія 8D (Eight Disciplines) [18]. Вона надає структурований і систематичний підхід до вирішення проблем, залучає та розвиває колективний інтелект для пошуку рішень і має достатньо високу ймовірність успішного завершення проєкту. Її призначенням є виявлення першопричин, розробка та впровадження короткострокових і довгострокових рішень для запобігання повторним проблемам, що дозволяє командам ефективно реагувати і мінімізувати вплив проблем на проєкт, що особливо важливо в умовах невизначеності та високої складності сучасних ІТ-проєктів.

Систематичність 8D походить з логіки циклу Демінга (PDCA – Plan-Do-Check-Act) [19]: планування, виконання, перевірка та коригувальні дії. Хоча спочатку методологія складалася з восьми етапів, або «дисциплін», пізніше до нього було додано етап початкового планування D0 з метою забезпечення ретельного виконання завдань і мінімізації ризиків.

Основними етапами (надалі D – дисципліни) методології є:

- D0: створення плану вирішення проблеми та визначення передумов, забезпечення дій з реагування на надзвичайні ситуації;
- D1: створення команди людей зі знанням предметної області;
- D2: чітке визначення проблеми із специфікацією її характеристик та впливу на виробничі процеси або реалізацію проєкту;
- D3: вживання негайних заходів для уникнення подальшого поширення проблеми та захисту від впливу на виробництво або проєкт;
- D4: аналіз причин проблеми для виявлення основних факторів, що її породжують та з'ясування чому проблему не було помічено під час її виникнення;
- D5: планування заходів для вирішення кореневих причин проблеми, а також перевірка їхньої ефективності;
- D6: впровадження коригувальних заходів для вирішення проблеми на постійній основі;
- D7: розробка та впровадження заходів для запобігання повторенню проблеми в майбутньому;
- D8: звітування та визначення успішного завершення проєкту, а також виокремлення отриманих уроків та кращих практик для майбутніх проєктів.

Усі етапи методології 8D представлені на рис. 2.1.

8D Methodology

Eight Disciplines Problem-Solving

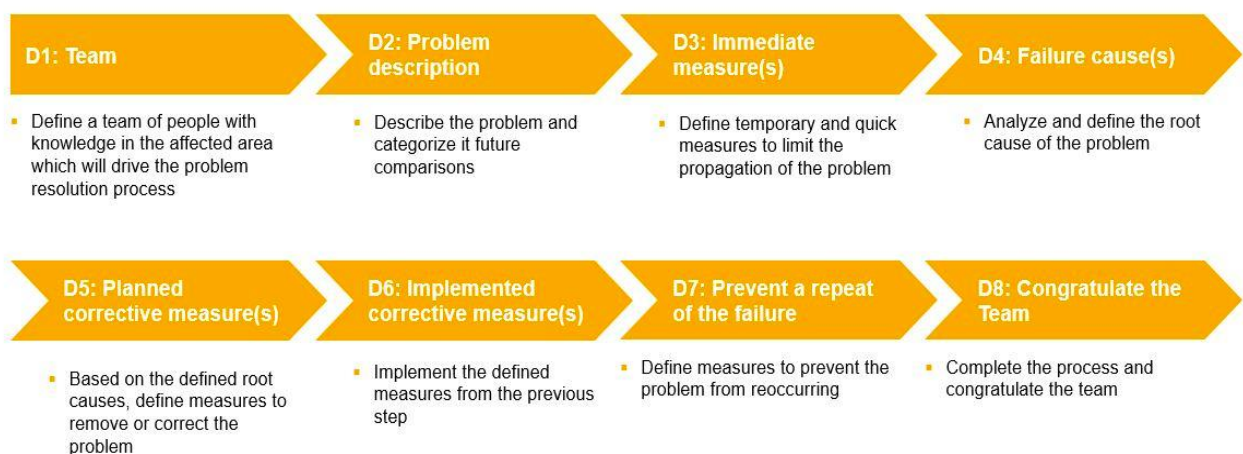


Рисунок 2.1 – Етапи методології 8D (Eight Disciplines)

Виходячи з цієї методології, у 1995 році Джон Коттер розробив покращену її версію. Модель Коттера [20] також включає в себе вісім етапів (рис. 2.2):

- визначення необхідності змін;
- формування коаліції і залучення персоналу;
- визначення стратегії змін;
- обговорення стратегії;
- усунення перешкод;
- створення короткострокових перемог;
- здійснення змін;
- закріплення змін.

Ця модель відома своєю системністю та комплексним підходом до управління змінами, що охоплює не лише технічні аспекти, але й людський фактор, ефективну комунікацію, розбудову коаліцій та подолання перешкод.

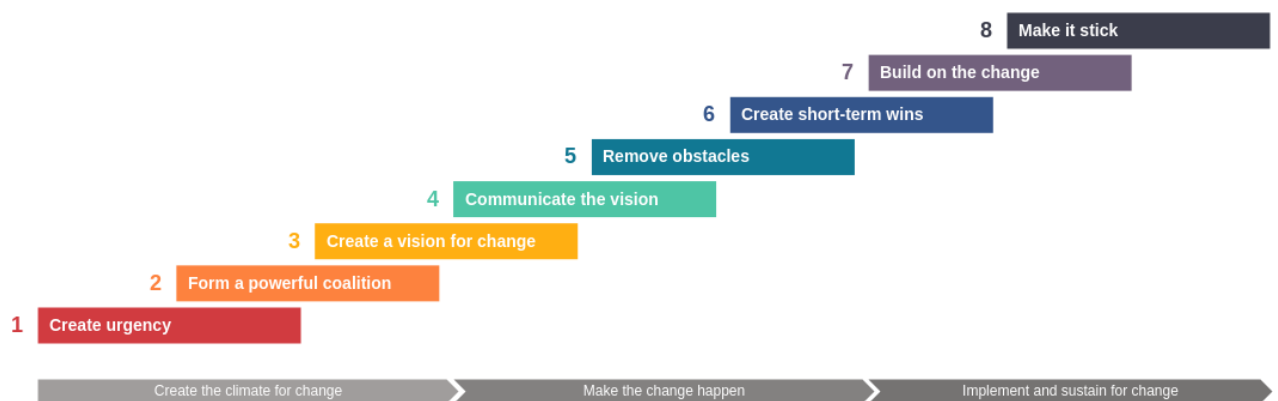


Рисунок 2.2 – Етапи моделі Коттера

Розглянемо кожен крок моделі Джона Коттера більш детально.

Крок 1. Створення нагальності.

Першим кроком є забезпечення того, щоб у всіх учасників ІТ-проекту було розуміння термінової необхідності змін. Це допомагає створити мотивацію та зацікавленість в зміні. Для того, щоб цей крок був успішним, необхідно забезпечити підтримку ініціативи змін з боку щонайменше 75% керівництва проекту та ключових зацікавлених сторін. На цьому етапі надається: обґрунтування необхідності змін, аналіз ризиків і можливості для вдосконалення.

Крок 2. Формування потужної коаліції.

Формування команди лідерів, яка буде відповідальна за реалізацію змін та мати вплив на організаційні процеси. До складу коаліції мають увійти авторитетні особи з різноманітним досвідом та ролями в проєкті, що забезпечить різнобічне бачення та підтримку змін на всіх рівнях.

Крок 3. Розробка бачення та стратегій.

Метою цього кроку є формування чіткого бачення ініціативи та планування ефективних стратегій, які допоможуть команді в її досягненні. Досяжна картина того, як буде виглядати ініціатива в кінцевому підсумку, мотивує членів команди докладати зусилля.

Крок 4. Поширення бачення змін.

Активне донесення бачення та стратегії до кожного учасника проєкту, включаючи зрозуміле пояснення цілей, переваг та очікуваних результатів змін. Керівна коаліція повинна постійно доносити своє бачення та демонструвати терміновість і переваги змін.

Крок 5. Усунення перешкод.

Ідентифікація та усунення перешкод, що заважають ефективному впровадженню змін, таких як внутрішні бюрократичні процеси, опір з боку співробітників, відсутність необхідних ресурсів чи компетенцій.

Крок 6. Встановлення короткострокових перемог.

Передбачає розробку чітких та досяжних цілей, кожна з яких повинна мати показники успіху, що дозволить учасникам проєкту побачити результати своєї роботи на ранніх етапах. Цей етап також важливий для підтримки мотивації всіх учасників проєкту.

Крок 7. Втілення змін.

Впровадження запланованих змін, забезпечення безперервності та підтримка цього процесу. На цьому етапі вкрай важливо дотримуватись визначених стратегій і підходів, адаптуючи їх за необхідності.

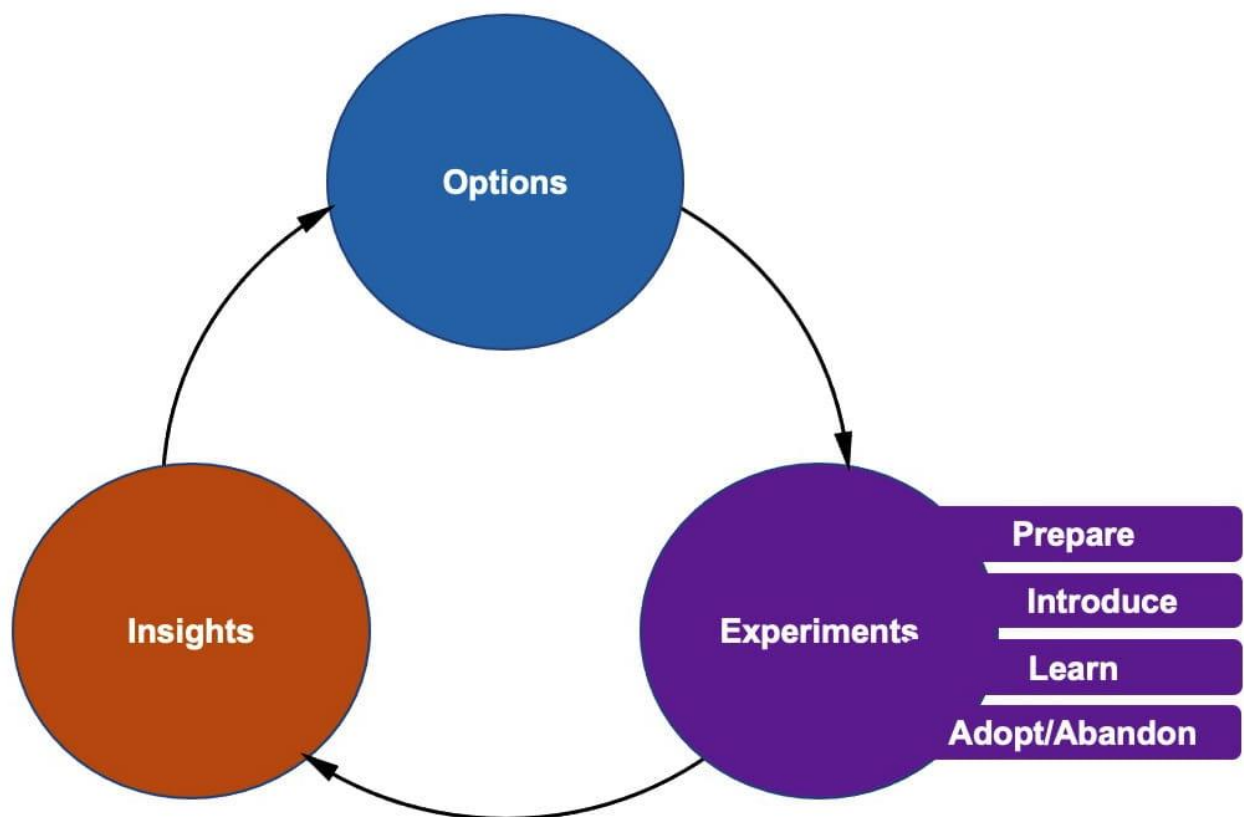
Крок 8. Закріплення змін.

Оцінка ефективності впроваджених змін і збір відгуків від учасників проекту допоможуть постійно вдосконалювати проектну діяльність.

Більш сучасною моделлю управління змінами є Lean Change Management. Ця модель поєднує концепції Lean-виробництва та Agile-розробки і спрямована на зниження витрат, усунення зайвих і провальних процесів, а також підвищення швидкості та якості впровадження змін [21].

Цикл Lean Change Management (LCM), який представлено на рис. 2.3, було розроблено Джеффом Андерсоном і Джейсоном Літлом. Він включає три основні види діяльності:

- розуміння;
- опції;
- експерименти.



© Copyright Project Management Institute
Original Copyright Jason Little

Рисунок 2.3 – Цикл Lean Change Management

Розуміння має на увазі ведення списку невиконаних завдань із інформацією про важливі речі, які потрібно втілити. Такі відомості часто ілюструють аспекти передбачуваних проблемних точок, які є перешкодою для гнучкості проєкту.

Для кожного розуміння, залежно від конкретного контексту, існують різноманітні варіанти (опції), які можуть вирішувати зазначені перешкоди. Під час оцінки кожного варіанту береться до уваги його можливі часові затримки, ступінь складності та потенційні витрати на реалізацію.

Після цього кожен варіант розглядається як потенційний «експеримент» або мінімальна життєздатна зміна, яка вимагає реалізації. Експеримент, у загальному визначенні, розглядається у чотирьох можливих стадіях – підготовці, впровадженні, вивченні та завершенні – які, як правило, відображаються за допомогою дошки Канбан.

Підготовка до кожної зміни за LCM це проведення переговорів, виокремлення переваг, планування і налагодження комунікації. Кожен учасник проєкту повинен розуміти, що зміна є лише експериментом. Для кожної зміни слід сформулювати просту гіпотезу щодо очікуваної вигоди. Це корисно для оцінки того, чи справді обраний варіант є покращенням і, отже, повинен бути прийнятий, чи зміни необхідно відкинути.

Наступним кроком є проведення експериментів зі змінами, тобто їх впровадження. Оскільки використовується гнучкий підхід, то вкрай важливо мінімізувати обсяг незавершеної роботи. Робота над багатьма змінами одночасно означає, що багато речей розпочато, але нічого не завершено, тому треба зосереджуватись на плавному потоці малих змін.

Після треба відстежити вплив змін, щоб побачити, чи досягається очікуваний позитивний результат. Цей огляд або «стан навчання» може тривати як кілька днів, так і багато місяців. Це є ключовим аспектом цієї моделі до управління змінами – не поспішати з передчасним оголошенням успіху.

Якщо експеримент став успішним або невдалим і більше не потрібно ретельно його відстежувати, то його треба перевести до статусу «завершено» (прийнято або відхилено). У разі успішності на цьому етапі також

додаються подальші робочі елементи до беклогу Kanban для розгортання змін в ІТ-проекті з необхідними комунікаціями і навчанням (у разі необхідності).

Кожна розглянута модель має власні переваги і недоліки, які було висвітлено у табл. 1.1.

Таблиця 1.1 – Переваги і недоліки методології 8D, моделі Коттера і моделі Lean Change Management

Модель	8D	Коттера	Lean Change Management
Переваги	<ul style="list-style-type: none"> – структурований і систематичний підхід; – орієнтована на процес; – детальний аналіз; – може вирішувати складні проблеми. 	<ul style="list-style-type: none"> – структурована; – робить акцент на залученні команди проекту; – має кращу адаптивність; – фокусується на довгострокових результатах. 	<ul style="list-style-type: none"> – гнучкість; – адаптивність; – мінімізує ризики і витрати; – залучення команди проекту; – ітеративність.
Недоліки	<ul style="list-style-type: none"> – обмеження часу; – обмеження ресурсів; – обмежена масштабованість; – є складною для втілення. 	<ul style="list-style-type: none"> – не завжди враховує специфіку конкретної зміни; – потребує значних ресурсів, зокрема часу; – трудомістка. 	<ul style="list-style-type: none"> – відсутність чіткої структури дій; – має ризик надмірності; – може виявитися складною для великих проектів; – потребує культурних змін в команді.

1.4.2 Аналіз технологій та інструментів управління змінами в ІТ-проектах

З огляду на швидкозмінювану природу інформаційних технологій, також доцільно використовувати сучасні технології управління змінами, такі як інструменти автоматизації процесів (Jira, ServiceNow, Microsoft Power Automate), системи відстеження змін (Git, Zendesk), системи відстеження помилок (Bugzilla,

Mantis) та аналізу ризиків (RiskWatch, SAP GRC). Використання цих технологій сприяє ефективному моніторингу та управлінню змінами в реальному часі, що дозволяє швидше реагувати на непередбачені ситуації та забезпечувати стабільність у процесі управління IT-проєктом.

Jira – потужна система автоматизації процесів [22], відстеження помилок і управління проєктами, розроблена компанією Atlassian, яка забезпечує гнучкість робочим процесам і адаптується до специфічних потреб команд. Вона надає ретельне планування, відстеження прогресу та контрольоване впровадження змін у програмне забезпечення. Система пропонує можливості для створення користувацьких дошок, призначення завдань, відстеження витраченого часу та генерації звітів про стан проєкту, що забезпечує прозорість на всіх етапах.

ServiceNow є хмарною платформою для автоматизації корпоративних процесів, яка охоплює широкий спектр сфер управління, включно з управлінням IT-послугами, операційною діяльністю та ризиками [23]. Її основною перевагою є можливість підвищення продуктивності та ефективності шляхом автоматизації повторюваних і рутинних завдань, а також централізованого керування процесами на рівні всієї організації. ServiceNow забезпечує безперебійну інтеграцію з існуючими системами та додатками, з можливістю створення уніфікованого середовища для управління IT-операціями, моніторингу та контролю змін, обслуговуванням користувачів, управлінням безпекою та іншими критично важливими процесами.

Microsoft Power Automate є інструментом для автоматизації робочих процесів, який дозволяє створювати автоматизовані послідовності дій за допомогою інтуїтивних шаблонів або візуальних конструкторів [24]. У контексті управління змінами, Power Automate забезпечує безшовну інтеграцію між різними додатками та сервісами Microsoft, а також підтримує взаємодію з численними сторонніми системами через відкриті API. Автоматизує повторювані завдання та оптимізує потоки даних, пов'язаних із впровадженням змін.

Окрім інструментів автоматизації, системи відстеження змін, такі як Git та Zendesk, відіграють важливу роль в управлінні IT-проєктами.

Git – це розподілена система контролю версій, яка дозволяє розробникам відстежувати зміни в коді, координувати роботу в команді та вести історію змін. Git забезпечує високу надійність і швидкість операцій, що робить його основним інструментом для управління версіями в багатьох ІТ-компаніях. Використання Git спрощує процеси інтеграції змін і знижує ризики конфліктів у коді.

Zendesk спеціалізується на управлінні взаємовідносинами з клієнтами і відстеженні змін у запитах та інцидентах. Платформа дозволяє централізовано управляти зверненнями клієнтів, автоматизувати відповіді та відстежувати стан виконання задач [25]. Інтеграція з іншими системами та аналітичні можливості Zendesk підвищують ефективність обслуговування клієнтів та забезпечують кращий контроль змін.

Окрім систем контролю версій для ефективного відстеження та управління змінами у програмному забезпеченні широко використовуються спеціалізовані системи відстеження помилок або багтрекери.

Bugzilla – потужний засіб для відстеження помилок та управління процесами розробки програмного забезпечення з гарним масштабуванням [26]. Вона дозволяє організаціям систематично відстежувати, пріоритезувати та призначати помилки відповідним розробникам. Багатогранні можливості Bugzilla включають відстеження змін у вихідному коді, управління доступом користувачів та генерацію звітів про стан проєкту.

Mantis – система відстеження помилок з відкритим вихідним кодом, зосереджена на простоті використання та швидкості розгортання. Вона пропонує зручний вебінтерфейс для відстеження дефектів, управління вимогами та завданнями, а також забезпечує інтеграцію з іншими інструментами розробки [27].

Для роботи з ризиками в процесі управління змінами використовують системи аналізу ризиків.

RiskWatch – це комплексне програмне рішення для управління ризиками, яке допомагає командам і організаціям ідентифікувати, аналізувати і контролювати ризики в різних сферах діяльності, включаючи ІТ-проєкти [28]. Ця система забезпечує централізоване сховище для збору та обробки даних про

ризиками, що дозволяє визначати потенційні загрози, оцінювати їх вплив і ймовірність виникнення. RiskWatch також пропонує інструменти для розробки стратегій реагування на ризики, призначення відповідальних осіб та моніторингу ефективності заходів з пом'якшення ризиків. Виявляє ризики, пов'язані з впровадженням нових функцій, технологій чи процесів, та допомагає оцінювати їх вплив на проєкт, що сприяє прийняттю обґрунтованих рішень щодо необхідних коригувальних дій та стратегій мінімізації ризиків перед впровадженням змін.

SAP GRC (Governance, Risk and Compliance) [29] – інтегрована платформа, яка поєднує модулі для ефективного управління ризиками, пов'язаних зі змінами в інформаційних системах, бізнес-процесах та інфраструктурі. Інструмент надає централізований огляд усіх ризиків, які впливають на ІТ-проєкти, включаючи ризики безпеки, операційні ризики та ризики невідповідності нормативним вимогам. Система дозволяє визначати ключові ризикові показники (KRIs) та встановлювати порогові значення для цих показників, що забезпечує безперервний моніторинг і звітність щодо ризиків. Модуль SAP Solution Manager дозволяє автоматизувати процеси управління змінами та відстежувати їх вплив на ризики, що робить підхід до моніторингу та контролю змін і ризиків в ІТ-проєкті більш комплексним.

1.5 Постановка задачі дослідження

У ході проведення аналізу сучасних моделей, методів і технологій управління змінами при реалізації проєктів у галузі інформаційних технологій було визначено їх основні недоліки:

- відсутність чіткої структури дій в процесі управління змінами;
- погана масштабованість;
- низька адаптивність, складність до розуміння;

- висока трудомісткість;
- не врахування специфіки конкретної зміни.

Метою науково-дослідної роботи є дослідження та проведення порівняльного аналізу методів і моделей управління змінами при виконанні ІТ-проектів, а також розробка власної моделі управління змінами, яка б враховувала якомога більше переваг і вирішувала недоліки існуючих.

Для досягнення сформульованої мети необхідно вирішити наступні завдання:

- провести детальний аналіз складових частин процесу управління змінами в ІТ-проектах;
- дослідити критерії, що використовуються в процесі управління змінами, і визначити їх вплив на ефективність виконання проекту;
- провести аналіз існуючих моделей і методів управління змінами при реалізації ІТ-проектів, включаючи оцінку ефективності кожного, можливості автоматизації та інтеграції з іншими процесами управління проектами;
- розробити власну модель управління змінами в ІТ-проекті на підставі виконаного аналізу переваг і недоліків існуючих та таку, що враховуватиме сучасні вимоги і специфіку ІТ-проектів;
- перевірити ефективність розробленої моделі на практиці;
- провести аналіз і сформулювати рекомендації щодо впровадження та використання отриманих наукових результатів.

Об'єктом дослідження є процес управління змінами в ІТ-проектах.

Предметом дослідження є моделі та методи управління змінами в ІТ-проектах, їх вплив на ефективність реалізації проектів.

2 РОЗРОБКА КОМБІНОВАНОЇ МОДЕЛІ УПРАВЛІННЯ ЗМІНАМИ В ІТ-ПРОЄКТАХ

У динамічному ІТ-середовищі постійно з'являються нові виклики, тому правильне формування стратегії управління змінами є критично важливим завданням будь-якого проєкту. Швидка адаптація до мінливих вимог, технологічних інновацій та ринкових тенденцій стає запорукою успішності і гарної конкурентоспроможності ІТ-проєктів.

Одними з найбільш поширених є моделі, орієнтовані на процес, такі як методологія 8D і теорія «Замороження – Рух – Відтавання» Курта Левіна [30]. Ці моделі пропонують структурований підхід до впровадження змін, розбиваючи процес на послідовні етапи, проте вони є надмірно лінійними та недостатньо гнучкими, що може бути досить неефективним рішенням використовувати їх в сучасному динамічному середовищі. Все ж таки вони більше підходять для вирішення конкретних проблем та усунення причин їх виникнення.

З іншого боку, моделі, що зосереджені на людях (наприклад, модель Коттера), акцентують увагу на залученні кожного учасника проєкту та подоланні опору змінам. Вони визнають важливість комунікації, мотивації та розвитку персоналу. Однак ці моделі можуть бути менш придатними для технічних або операційних змін.

Сучасні ітераційні та гнучкі методології, такі як Agile та Lean Change Management, набувають все більшої популярності. Вони передбачають короткі ітерації, постійне вдосконалення та залучення зацікавлених сторін. Ці підходи є ефективними для адаптації до мінливих умов, але можуть бути менш структурованими, важкими для масштабування і розуміння.

Для максимальної ефективності процесу управління змінами, деякі організації розробляють власні гібридні моделі, поєднуючи елементи різних підходів, але розробка та впровадження таких моделей управління змінами також має свої ризики. Так об'єднання різних підходів може призвести до

непослідовності та плутанини, якщо компоненти моделі не будуть ретельно продумані, інтегровані та узгоджені. Також слід пам'ятати, що створення власної гібридної моделі вимагає значних зусиль, часу та ресурсів для розробки, тестування та налаштування під специфіку кожного проекту.

Загалом ефективність кожної моделі залежить від низки факторів, таких як масштаб змін, галузь, організаційна культура та готовність зацікавлених сторін до змін. Неefективні моделі можуть бути надто жорсткими, не враховувати людський фактор або не відповідати специфіці проекту. З іншого боку, ефективні підходи повинні враховувати контекст, залучати зацікавлені сторони та забезпечувати гнучкість та безперервне вдосконалення.

На підставі проведеного аналізу та синтезу розглянутих моделей (8D, Коттера і LCM), необхідно розробити комбіновану модель управління змінами в IT-проектах, яка враховуватиме сильні сторони кожного підходу та допоможе забезпечити більш ефективне та успішне впровадження змін. Властивостями моделі, що розробляється, повинні бути:

- чітка структура дій;
- висока адаптивність та гарна масштабованість;
- простота використання і якомога менша трудомісткість;
- врахування специфіки конкретної зміни
- безперервне вдосконалення.

Результати аналізу показали, що 8D і модель Коттера є достатньо структурованими та мають чітко визначені 8 кроків, що не можна сказати про Lean Change Management, де структура циклічна і кожна зміна може мати невизначену кількість кроків, через заздалегідь невизначене число експериментів. Однак циклічність або постійне виконання процесу збору «розумінь», що представлено ітеративністю процесу LCM, надає високу адаптивність до середовища та мінімізує ризики і витрати, що треба враховувати при розробці нової моделі. Також слід врахувати ступінь залучення команди проекту, що є великою перевагою гнучких і ошадливих (Lean) методологій, а також, частково, моделі Коттера.

Всі розглянуті моделі є достатньо трудомісткими, деякі – є складними для розуміння для зацікавлених сторін або мають ризик надмірності. На жаль, трудомісткість значно знизити не вдасться, бо тоді процес стане зовсім непередбачуваним і неефективним, але позбутися від надмірності є досяжним завданням.

Масштабованість відноситься до здатності моделі ефективно працювати при збільшенні розміру або складності проекту. І методологія 8D, і модель Коттера є обмеженими щодо цього критерію, бо мають спільні проблеми – обмеження часу і потреба у ресурсах. На відміну від них LCM є достатньо масштабованою як для великих, так і для малих проектів.

Найменшу складність щодо розуміння дає модель Коттера через власну орієнтацію на команду, а не на процес. Складними в ній є створення відчуття терміновості, що може вимагати глибокого розуміння психології кожного учасника проекту, та потреба в наявності стратегічного мислення у кожного члена коаліції. Більш складною є методологія 8D, яка створює перешкоду координації, розуміння ролей і відповідальностей в команді. Для ефективного застосування 8D, необхідно мати певні технічні знання та навички. Lean Change Management, на відміну від інших моделей, не надає жорстко визначену послідовність кроків, а орієнтується тільки на принципи Lean та Agile, які можуть варіюватися залежно від конкретного контексту. Її використання потребує відповідну організаційну культуру, навички роботи в динамічних умовах і розуміння принципів Lean:

- усунення втрат;
- постійне вдосконалення й ітеративність;
- активне залучення співробітників;
- регулярний збір зворотнього зв'язку.

Враховавши всі переваги і недоліки методології 8D, моделі Коттера і Lean Change Management було розроблено комбіновану модель управління змінами, яку умовно можна представити у вигляді послідовності п'яти основних етапів, що комплексно охоплюють весь процес управління змінами.

Етап 1. Формування команди зі змін.

Першочерговим етапом у процесі управління змінами є створення потужної команди, яка буде відповідати за реалізацію змін. Вона повинна складатися з ключових зацікавлених сторін, які мають вплив та авторитет у проєкті, зі змішаним досвідом і компетенціями, що забезпечить різноманітне бачення та підтримку змін на всіх рівнях проєкту. Для ясного розуміння своїх завдань та обов'язків необхідно чітко визначити ролі та відповідальності кожного члена команди.

Етап 2. Ідентифікація і формування змін.

Наступним кроком є ідентифікація та формування змін, які необхідні для успішного розвитку проєкту. Цей процес включає детальний аналіз поточних проблем і потреб, що дозволяє виявити конкретні зміни, які мають бути здійснені. При цьому потрібно ділити великі зміни на більш менші з прив'язкою до головної, щоб якомога детальніше сформулювати бачення ініціативи. Після того, як зміну ідентифіковано і, можливо, розбито на декілька складових, вона одразу потрапляє у загальний список змін. Далі кожна зміна і її складова повинна бути чітко описана, з урахуванням її пріоритету, специфікації та характеристик, а також попередньої оцінки впливу на проєкт, включаючи технічні, організаційні та бізнес аспекти.

Етап 3. Стратегічне планування змін.

Розробка детального плану впровадження кожної зміни з встановленням чітких цілей, завдань, відповідальних осіб і термінів виконання. При формуванні цілей необхідно врахувати їх досяжність, можливі перешкоди і визначити проміжні показники успіху (короткострокові перемоги), які можуть бути досягнуті протягом процесу впровадження змін. Для усунення перешкод необхідно розробити стратегію, яка буде визначати заходи з ефективного подолання ідентифікованих на попередньому кроці потенційних проблем і ризиків. Додатково необхідно розробити стратегію комунікації щодо кожної зміни, яка забезпечить інформування команди та зацікавлених сторін про хід її

впровадження. Вона повинна являти собою регулярний зворотній зв'язок, що дозволить швидко реагувати на виникаючі проблеми та перешкоди.

Етап 4. Впровадження змін.

Етап впровадження передбачає реалізацію всіх запланованих заходів щодо кожної зміни відповідно до розробленого плану дій. Під час впровадження необхідно постійно відстежувати хід виконання плану кожної зміни, забезпечуючи своєчасне розв'язання проблем й усунення перешкод, які можуть виникнути. Інформування команди про хід роботи і досягнуті проміжні успіхи є ключовим для показників залученості та мотивації.

Етап 5. Оцінка результатів впровадження змін.

Останнім кроком є оцінка ефективності результатів впровадження змін, яка включає збір й аналіз даних і відгуків про досягнуті результати, їх порівняння з попередньою оцінкою, визначеними цілями й очікуваннями кожної зміни. Цей показник дозволяє визначити, наскільки успішно були реалізовані заплановані заходи, та які уроки можна винести для вдосконалення майбутньої проектної діяльності. У разі необхідності, на підставі отриманих результатів, можуть бути викликані додаткові коригувальні дії щодо подальшого покращення процесу управління змінами відповідно до специфіки конкретного ІТ-проєкту.

3 ЕКСПЕРЕМЕНТАЛЬНА ПЕРЕВІРКА ПРАЦЕЗДАТНОСТІ ТА ОЦІНКА ЕФЕКТИВНОСТІ ВИКОРИСТАННЯ РОЗРОБЛЕНОЇ МОДЕЛІ

3.1 Опис умов проведення тестування

Для ґрунтовного та всебічного вивчення і оцінювання ефективності застосування різних методологій управління змінами необхідно ретельно визначити відповідні умови для проведення тестування. Такі умови повинні включати опис реального ІТ-проєкту або ситуації впровадження змін в організації, а також вичерпний та деталізований опис самих змін за переліком критеріїв, який було визначено у підрозділі 1.3.

Опис ІТ-проєкту є критично важливим для створення належного контексту, у межах якого будуть застосовуватися різні методології управління змінами. Цей опис має містити таку ключову інформацію:

- передумови необхідності ініціювання проєкту;
- дані щодо поточного стану справ та існуючих проблем, які мають бути вирішені;
- чітко визначені цілі й очікувані результати;
- перелік зацікавлених сторін, їхні ролі й функції;
- часові рамки та терміни виконання робіт за проєктом;
- бюджетні обмеження та фінансові показники, що мають бути дотримані в рамках проєкту.

Крім опису самого ІТ-проєкту, вкрай важливим є надання детального опису змін, які передбачається протестувати з використанням різних методологій. Такий опис повинен ґрунтуватися на ретельному аналізі та враховувати низку ключових критеріїв:

- а) характер змін:
 - 1) технологічні;
 - 2) процесні;
 - 3) організаційні;

- 4) культурні;
- б) обсяг змін:
 - 1) кількість залучених процесів;
 - 2) пов'язані системи;
 - 3) залучені підрозділи і працівники організації;
- в) рівні критичності впливу змін на якість і ефективність проєкту;
- г) готовність учасників проєкту до впровадження запланованих змін;
- г) потенційні ризики, перешкоди та проблеми, що можуть виникнути на шляху успішної реалізації змін.

Опису ІТ-проєкту та опису змін за вищезазначеними критеріями буде достатньо для оцінки ефективності методології 8D, моделі Коттера, моделі Lean Change Management і комбінованої моделі управління змінами.

3.1.1 Опис ІТ-проєкту

Для проведення апробації розробленої моделі управління змінами було обрано нещодавно завершений ІТ-проєкт розробки вебзастосунку Dream Beauty Parlor. Цей проєкт був ініційований для однойменної компанії DreamBeautyParlor, яка представлена салоном краси і надає перукарські та косметологічні послуги. Головною метою цього проєкту було розробити невеликий вебзастосунок, який би дозволив клієнтам салону переглядати послуги, що надаються, зручно залишати заявки, і отримувати корисну інформацію за сферою краси і догляду.

Обмеженнями проєкту були:

- часові: встигнути до кінця травня 2024 року;
- фінансові: у результаті первинних переговорів з замовником було визначено бюджетні рамки $\$800 \pm \200 .

Проєкт був розпочатий 26 березня 2024 року та завершений 10 травня 2024 року. Тривалість проєкту склала 263,5 години, а загальна вартість – $\$845,58$.

Основні статті витрат розподілені таким чином:

- заробітна плата (87.5%);
- оренда віддаленого серверу для етапу впровадження системи (6.4%);
- витрати на електроенергію (3.2%);
- програмне забезпечення (2.9%).

Вимогами замовника були:

- відсутність збоїв та помилок при роботі з вебзастосунком;
- виконання проєкту у визначений термін і бюджет;
- досягнення всіх поставлених цілей даного проєкту;
- наявність каналу зв'язку з керівником проєкту задля уточнення деталей та контролю виконання проєкту;
- отримання презентацій результатів виконання робіт за проєктом.

Вимогами до кінцевого продукту були:

- вебзастосунок повинен працювати на віддаленому сервері, доступ до якого можна отримати у будь-який момент часу;
- мати гарну пропускну спроможність (до 500 користувачів одночасно);
- бути кросплатформним, з адаптацією під різні веббраузери;
- повинен реалізовувати функції запису на прийом до майстрів в салон, отримання інформації про послуги та іншу корисну інформацію з блогу;
- мати належний захист інформації;
- бути надійним та відмовостійким;
- мати швидкий і привабливий інтерфейс;
- мати можливість масштабування;
- для своєї роботи використовувати тільки ліцензоване програмне забезпечення (ПЗ) або ПЗ з відкритим вихідним кодом.

Учасників обраного ІТ-проєкту наведено у табл. 3.1.

Таблиця 3.1 – Учасники ІТ-проєкту розробки вебзастосунку Dream Beauty Parlor та попередній розподіл їх ролей

Учасник	Роль	Відповідальність
Власник ПП «DreamBeautyParlor»	Замовник, Спонсор	Надає вимоги. Консультує та визначає ключові цілі, основні етапи і терміни робіт, фінанси проєкту. Приймає результати роботи за проєктом.
Менеджер проєктів (Middle)	Керівник проєкту	Комунікація з замовником та контроль виконання проєкту
Бізнес-аналітик (Middle)	Виконавець	Виконує узгоджені роботи
Дизайнер (Senior)	Виконавець	Виконує узгоджені роботи
FE-розробник (Junior)	Виконавець	Виконує узгоджені роботи
BE-розробники (Junior, Middle)	Виконавець	Виконує узгоджені роботи
Тестувальник (Middle)	Виконавець	Виконує узгоджені роботи
Системний адміністратор (Senior)	Виконавець	Виконує узгоджені роботи

Планування ІТ-проєкту проводилося в традиційному засобі для планування – Microsoft Project [31], який являє собою систему управління проєктами, розроблену корпорацією Microsoft. Він допомагає менеджерів проєкту в розробці планів, розподілі ресурсів за роботами, відстеженні прогресу і аналізі обсягів робіт. Також можна відстежувати критичний шлях і складати розклади з урахуванням використовуваних ресурсів. Послідовність виконання робіт візуалізує діаграма Ганта.

На рис. 3.1 зображено діаграму Ганта ІТ-проєкту розробки вебзастосунку DreamBeautyParlor.

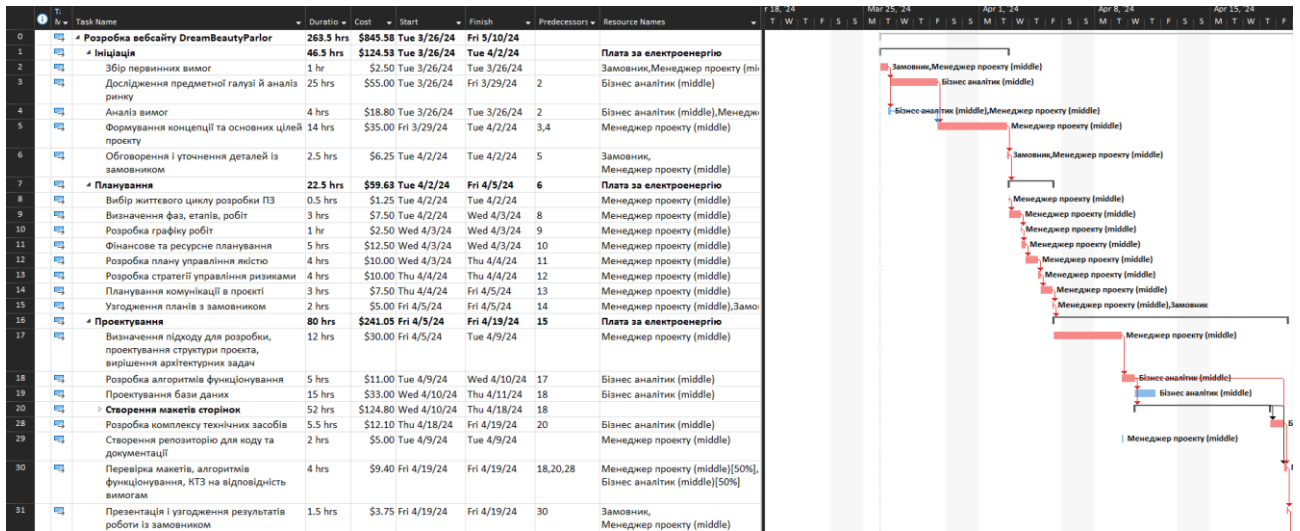


Рисунок 3.1 – Діаграма Ганта ІТ-проєкту розробки вебзастосунку DreamBeautyParlor

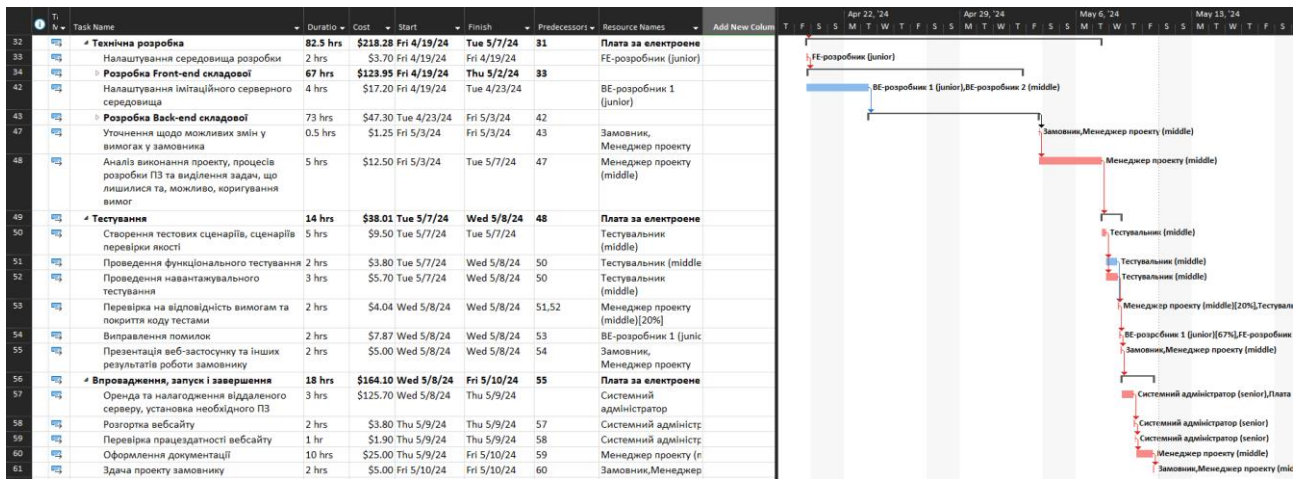


Рисунок 3.1, аркуш 2

На початковій стадії ініціації проєкту, яка тривала 46,5 годин, відбувся збір первинних вимог від замовника, після – дослідження предметної галузі й аналіз ринку бізнес-аналітиком. Менеджер проєкту проаналізував вимоги, сформував концепцію та основні цілі проєкту, а також уточнив і узгодив деталі з замовником.

Наступним кроком було ретельне планування проєкту. Менеджер визначив життєвий цикл розробки програмного забезпечення, етапи та роботи. Був розроблений графік робіт, здійснено фінансове та ресурсне планування, створені план управління якістю, стратегії управління ризиками та план комунікації.

Етап проектування, який зайняв 80 годин, визначив підхід до розробки і структуру проекту. Бізнес-аналітик розробляв алгоритми функціонування, базу даних і комплекс технічних засобів, а дизайнер створив макети сторінок. Менеджер створив репозиторій для коду та документації, а результати роботи були перевірені на відповідність вимогам та узгоджені з замовником.

Технічна розробка проекту тривала 82,5 години. Front-end розробник налаштував середовище та реалізував front-end складову, а два back-end розробники створили back-end частину, реалізували базу даних та інтегрували її з front-end.

Етап тестування тривав 14 годин. У процесі реалізації проекту тестувальник створював тестові сценарії, після провів функціональне та навантажувальне тестування, перевіряв відповідність вимогам та покриття коду тестами. Брав участь у виявленні помилок і формуванні запитів на їх виправлення. Після тестування вебзастосунок був презентований замовнику.

Впровадження та запуск проекту зайняли 18 годин. Системний адміністратор орендував і налагодив віддалений сервер, встановив необхідне програмне забезпечення, розгорнув вебсайт та перевіряв його працездатність. Після успішного впровадження було оформлено документацію і передано проєкт замовнику.

3.1.2 Опис змін

Для більш об'єктивного аналізу й оцінки ефективності розглянутих моделей та комбінованої було розглянуто три різні за складністю зміни:

- незначна: додання окремої сторінки з інформацією про акції та знижки;
- середня: інтеграція з платіжною системою;
- велика: переробка інтерфейсу користувача вебзастосунку.

Кожна з трьох змін представлена у відповідній табл. 3.2 – 3.4.

Таблиця 3.2 – Опис за критеріями незначної зміни «Додання окремої сторінки з інформацією про акції та знижки»

№	Критерій	Опис
1	Пріоритет	Низький (не критичний)
2	Складність	Незначна
3	Відповідність вимогам і стратегічним цілям проєкту	Зміна не суперечить основним бізнес-цілям проєкту і може розглядатися як доповнення
4	Сумісність	Додавання нової сторінки не повинно зумовити конфлікти з існуючою архітектурою і функціональністю, за умови належного проєктування
5	Вартість	Незначна, близько \$45, обумовлена витратами часу дизайнера на створення макету сторінки, front-end і back-end розробників на створення і інтеграцію сторінки
6	Ресурси	Бізнес-аналітик, дизайнер, Front-end і Back-end розробники, тестувальник
7	Терміни	Необхідно приблизно 18 годин для розробки і реалізації в проєкті. Загалом затримки у графіку виконання робіт невеликі
8	Якість	Не постраждає за умови дотримання тих самих підходів і стандартів, що і для іншої частини проєкту
9	Ефективність	Покращить інформативність і, трохи, задоволеність користувачів
10	Ризики	Помилки з існуючим кодом, неправильна взаємодія з базою даних

Таблиця 3.3 – Опис за критеріями зміни середньої складності «Інтеграція з платіжною системою»

№	Критерій	Опис
1	2	3
1	Пріоритет	Низький
2	Складність	Середня, через потребу в інтеграції з розробленими модулями, додаткових засобах забезпечення безпеки особистих даних і платежів
3	Відповідність вимогам і стратегічним цілям проєкту	Відповідає цілям покращення зручності та досвіду користувачів
4	Сумісність	Потребує інтеграції з існуючою системою бронювання та базою даних

Кінець таблиці 3.3

1	2	3
5	Вартість	Приблизно \$300. У цю суму входить сплата за використання платіжної системи за рік.
6	Ресурси	Бізнес-аналітик, дизайнер, Front-end і Back-end розробники, тестувальник
7	Терміни	5 днів
8	Якість	Потребує ретельного тестування безпеки
9	Ефективність	Зручність для клієнтів, збільшення прибутку
10	Ризики	Збої і вразливості платіжної системи, ризики безпеки і втрати даних

Таблиця 3.4 – Опис за критеріями складної зміни «Переробка інтерфейсу користувача вебзастосунку»

№	Критерій	Опис
1	Пріоритет	Високий
2	Складність	Висока
3	Відповідність вимогам і стратегічним цілям проекту	Відповідає стратегічним цілям проекту щодо створення привабливого та зручного продукту для користувачів
4	Сумісність	Часткова, необхідно буде забезпечити сумісність нового інтерфейсу з існуючою back-end логікою та базами даних
5	Вартість	Висока, від \$150 до \$500.
6	Ресурси	Дизайнер, Front-end і Back-end розробники, тестувальник
7	Терміни	Призведе до значних затримок терміном до місяця.
8	Якість	Необхідно врахувати всі вимоги і пропозиції замовника для розробки рішення, яке буде зваженим. Використання Usability і User Experience тестування обов'язкове.
9	Ефективність	Покращений інтерфейс може значно підвищити задоволеність зацікавлених сторін та конкурентоспроможність продукту
10	Ризики	Порушення існуючої функціональності, конфлікти, погіршення продуктивності вебзастосунку, негативне сприйняття кінцевим користувачем

Через те, що проект має статус завершений і використовується для моделювання ситуації змін, необхідно визначити на якому саме етапі ці зміни виникли. Нехай незначна зміна «Додання окремої сторінки з інформацією про акції та знижки» відбулася ще на етапі «Узгодження планів з замовником», тобто

ще до початку проєктування. Середня зміна «Інтеграція з платіжною системою» виникла як додаткова вимога в процесі проєктування, а саме під час проєктування бази даних. Необхідність великих змін («Переробка інтерфейсу користувача вебзастосунку») виникла вже на презентації готового вебзастосунку та інших результатів роботи.

3.2 Оцінка ефективності методології 8D

Використаємо методологію 8D для кожної визначеної зміни.

Розглянемо незначну зміну «Додання окремої сторінки з інформацією про акції та знижки» застосовуючи методологію 8D:

а) D1 (визначення команди):

- 1) бізнес-аналітик;
- 2) дизайнер;
- 3) front-end розробник;
- 4) back-end розробник;
- 5) тестувальник;

б) D2 (опис проблеми): необхідно додати окрему сторінку на вебсайті, яка міститиме інформацію про поточні акції, знижки та спеціальні пропозиції салону краси DreamBeautyParlor;

в) D3 (тимчасові коригувальні дії): не потрібні, оскільки це незначна зміна, яка не впливає на роботу визначеного функціоналу;

г) D4 (визначення причин): додаткова вимога від замовника щодо покращення інформативності вебзастосунку;

г) D5 (розробка постійного коригувального заходу):

- 1) бізнес-аналітик розробляє елементи бази даних і редагує алгоритми функціонування;
- 2) дизайнер створює макет нової сторінки з акціями/знижками;

- 3) Front-end розробник створює HTML/CSS/JS код сторінки;
- 4) Back-end розробник інтегрує сторінку з базою даних;
- 5) тестувальник перевіряє функціональність та якість нової сторінки;

д) D6 (впровадження заходу): команда впроваджує зміни на імітаційному середовищі, тестує їх, а після успішного тестування і завершення інших робіт з розробки – впроваджує готовий продукт;

е) D7 (запобігання повторення): на початкових етапах проєкту важливо уважно збирати усі вимоги замовника, щоб уникнути додаткових витрат на впровадження змін після завершення розробки. Виникнення додаткових вимог ймовірно сталося через недостатню інформованість замовника про всі можливі функції вебзастосування;

є) D8 (визнання команди): керівник проєкту відзначає роботу команди у впровадженні покращення функціоналу вебзастосування.

Тепер розглянемо зміну середньої складності «Інтеграція з платіжною системою» за методологією 8D:

а) D1 (визначення команди):

- 1) Front-end і Back-end розробник;
- 2) тестувальник;
- 3) системний адміністратор;
- 4) менеджер проєкту;

б) D2 (опис проблеми): замовник хоче інтегрувати вебзастосунок з платіжною системою, щоб клієнти могли оплачувати послуги салону онлайн;

в) D3 (тимчасові коригувальні дії): не потрібні через відсутність функціоналу платежів на момент виникнення ініціативи;

г) D4 (визначення причин): додаткова вимога замовника для покращення зручності користувачів;

г) D5 (розробка постійного коригувального заходу):

- 1) бізнес-аналітик визначає безпекові заходи;
- 2) дизайнер створює відповідну форму оплати, яка після реалізується на front-end;

3) back-end розробник інтегрує API платіжної системи з базою даних замовлень;

4) тестувальник створює тест-кейси і перевіряє сумісність, безпеку та коректність роботи платежів;

д) D6 (впровадження заходу): команда впроваджує зміни на тестовому середовищі, проводить навантажувальне тестування, а після успішного тестування і завершення інших робіт з розробки – впроваджує готовий продукт;

е) D7 (запобігання повторення): більш ретельний збір вимог на початку та регулярна їх обговорення з замовником. Ймовірно ця зміна була викликана з тих самих причин, що і попередня (через недостатню інформованість замовника про всі можливі функції вебзастосунку);

є) D8 (визнання команди): менеджер проєкту відзначає роботу команди у впровадженні важливої функціональної можливості.

Перейдемо до розгляду складної зміни за методологією 8D:

а) D1 (визначення команди):

- 1) дизайнер;
- 2) front-end розробник;
- 3) back-end розробник;
- 4) тестувальник;
- 5) бізнес-аналітик;
- 6) менеджер проєкту;

б) D2 (опис проблеми): під час презентації результатів проєкту замовник висловив незадоволення поточним інтерфейсом і вимагав його повної переробки для покращення юзабіліті та досвіду користувачів;

в) D3 (тимчасові коригувальні дії): відкласти офіційний реліз вебзастосунку до впровадження змін;

г) D4 (визначення причин): неврахування потреб і особистих вподобань замовника на етапі проєктування інтерфейсу;

г) D5 (розробка постійного коригувального заходу):

- 1) бізнес-аналітик аналізує вимоги до нового інтерфейсу;

- 2) дизайнер створює макети сторінок;
- 3) front-end розробник реалізує UI;
- 4) back-end розробник інтегрує його з бізнес-логікою;
- 5) тестувальник перевіряє юзабіліті та сумісність;
- 6) менеджер узгоджує і презентує результати роботи;

д) D6 (впровадження заходу):

- 1) повторне представлення і узгодження макетів із замовником;
- 2) у разі узгодження макетів – втілення змін і повторна презентація готового застосунку, у разі відхилення макетів – переробка за останніми вимогами замовника;

3) після позитивного зворотнього зв'язку за кінцевим продуктом – його повне розгортання;

е) D7 (запобігання повторення): регулярно залучати замовника і цільових користувачів до процесу проєктування UI/UX на ранніх етапах;

є) D8 (визнання команди): менеджер відзначає зусилля команди у виконанні масштабного завдання в стислі терміни.

Застосування методу 8D для управління незначною і середньою зміною було досить ефективним. Кроки були чіткими, залучені всі необхідні ролі, зміни впроваджені якісно і швидко без значних затримок проєкту. Єдиним недоліком можна назвати те, що на етапі збору вимог замовник не озвучив потреби відразу і підвищені ризики, пов'язані з безпекою платежів, які потрібно було ретельно враховувати і тестувати. На жаль зі складною зміною виникли значні проблеми. Методологія допомогла структурувати процес, проте мало місце серйозне недопрацювання на етапі проєктування, через що виникло багато ризиків і необхідність значних додаткових зусиль. Позитивним є залучення замовника до процесу проєктування на ранніх етапах.

3.3 Оцінка ефективності моделі Коттера

Використаємо модель Коттера для кожної визначеної зміни. Результати застосування моделі Коттера для змін незначного, середнього та важкого рівня складності представлено у табл. 3.5.

Таблиця 3.5 – Результати застосування моделі Коттера для змін незначного, середнього та важкого рівня складності

Крок моделі Коттера	Незначна зміна «Додання окремої сторінки з інформацією про акції та знижки»	Зміна середньої складності «Інтеграція з платіжною системою»	Складна зміна «Переробка інтерфейсу користувача вебзастосунку»
1. Створення нагальності	Замовник і керівництво проєкту має донести необхідність додання інформаційної сторінки про акції/знижки для покращення інформативності сайту та задоволеності клієнтів, продемонструвати переваги цієї зміни.	Керівництво проєкту має донести необхідність інтеграції з платіжною системою для забезпечення можливості онлайн-оплати клієнтами послуг салону. Це підвищить зручність і може збільшити продажі та прибутки.	Керівництво проєкту має продемонструвати гостру необхідність переробки користувацького інтерфейсу через його незадовільну якість та невідповідність баченню замовника.
2. Формування потужної коаліції	Створюється команда, відповідальна за реалізацію зміни – бізнес аналітик, дизайнер, front-end, back-end розробники, тестувальник. Керівник проєкту забезпечує підтримку з боку замовника.	Створюється команда у складі замовника, менеджера проєкту.	Формується потужна коаліція у складі менеджера проєкту, бізнес-аналітика, дизайнера і замовника.

Продовження таблиці 3.5

3. Розробка бачення та стратегій	Формулюється чітке бачення – створити окрему сторінку з актуальним контентом про акційні пропозиції салону. Розробляється стратегія реалізації – макети, розробка, тестування, впровадження.	Бачення – інтегрувати на сайт надійну і зручну платіжну систему, адаптовану під інтерфейс і вимоги безпеки. Стратегія реалізації включає вибір платіжного провайдера, розробку, тестування і впровадження.	Бачення – створити новий зручний, сучасний та інтуїтивний інтерфейс відповідно до побажань замовника з використанням найкращих практик UI/UX. Буде використано стратегію поетапної переробки з регулярним залученням зворотнього зв'язку з боку замовника.
4. Поширення бачення змін	Керівник доносить до всіх учасників проєкту бачення щодо зміни та її переваги – покращення досвіду клієнтів, підвищення конверсій та лояльності.	Менеджер проєкту доносить до всіх учасників переваги інтеграції системи платежів: зручність для клієнтів, підвищення конверсій і прибутків компанії.	Керівна коаліція активно транслює бачення нового інтерфейсу як вирішальний крок для підвищення задоволеності замовника, його клієнтів, конверсій та лояльності.
5. Усунення перешкод	Виявлення та усунення потенційних перешкод – конфлікти з наявним дизайном/функціоналом, узгодження контенту тощо.	Виявлення ризиків безпеки, потенційних конфліктів з системою бронювання та іншим функціоналом. Можливо залучення експертів з безпеки.	Виявляються та долаються потенційні перешкоди – конфлікти з існуючим функціоналом, ризики порушення відмовостійкості, недостатні ресурси та компетенції в команді.

Кінець таблиці 3.5

6. Короткострокові перемоги	Визначаються показники успіху, наприклад, сторінку з акціями впроваджено за 4 дні після узгодження вимог.	Інтеграція з платіжною системою на тестовому середовищі за 5 днів та проходження базового тестування протягом 2-х тижнів.	Розробка прототипів ключових елементів нового інтерфейсу, узгодження з замовником та їх успішне usability-тестування користувачами протягом місяця.
7. Втілення змін	Команда реалізує зміну згідно розробленої стратегії – створення макетів, front/back-end розробка, інтеграція з базою даних, тестування.	Back-end команда інтегрує API обраного платіжного провайдера, фронтенд створює платіжну форму, проводиться ґрунтовне тестування безпеки та навантаження.	Поетапне впровадження нового інтерфейсу згідно стратегії - проектування, front/back-end розробка, регулярне тестування користувачами, інтеграція з існуючою системою.
8. Закріплення змін	Після впровадження аналізуються метрики переглядів нової сторінки, збирається фідбек від клієнтів. У разі успіху – зміни закріплюються.	Збір фідбеку від клієнтів, моніторинг метрик онлайн-продажів. За позитивних результатів повне впровадження інтеграції.	Моніторинг ключових метрик – конверсій, відвідуваності, поведінкових факторів користувачів. Збір відгуків. За успішних результатів – завершити зміну.

Модель Коттера виявилась неефективною для впровадження незначної зміни через свій масштабний характер. Для змін середньої і високої складності відпрацювала добре. В усіх випадках вона забезпечила системний підхід з залученням керівництва, формуванням бачення, опрацюванням ризиків, усуненням перешкод і зворотнім зв'язком.

Головним недоліком моделі Коттера виявилась її висока тривалість відносно кожної зміни, що призводить до сповільнення проєкту.

3.4 Оцінка ефективності моделі Lean Change Management

Застосуємо модель Lean Change Management для кожної з трьох змін. Почнемо з незначної зміни «Додання сторінки з акціями та знижками».

1. Розуміння: додаткова сторінка з інформацією про акції та знижки може принести цінність для користувачів, покращивши інформативність вебзастосунку.

2. Опції: розглядаються різні варіанти реалізації – статична сторінка (низька складність, низька вартість, 1-3 дні), динамічно оновлювана з бази даних (середня складність, середня вартість, 1-2 тижні), створення системи управління акціями (висока складність, висока вартість, 1-2 місяці) тощо.

3. Експерименти:

а) підготовка: команда узгоджує деталі змісту та функціоналу сторінки з замовником, формулюється гіпотеза очікуваних переваг;

б) впровадження: дизайнер створює макет, front-end та back-end розробники втілюють сторінку, інтегрують з базою даних;

в) вивчення: пілотне тестування сторінки для оцінки зручності та достовірності наданої інформації.

г) завершення: за позитивних результатів – розгортання, в іншому випадку – відмова від зміни.

Розглянемо зміну середньої складності «Інтеграція з платіжною системою» за моделлю LCM:

1. Розуміння: наявність можливості приймати онлайн-платежі на вебсайті може принести значну цінність бізнесу та покращити досвід користувачів, однак це вимагає інтеграції з платіжною системою.

2. Опції:

а) інтеграція з популярними платіжними системами типу PayPal, Stripe;

б) використання платіжного шлюзу банку замовника;

в) розробка власного платіжного модуля.

3. Експерименти:

а) підготовка: визначається найбільш оптимальний платіжний провайдер, узгоджуються вимоги безпеки та інтерфейсу з замовником, формулюється гіпотеза щодо зростання конверсій після інтеграції платежів;

б) впровадження: бізнес-аналітик визначає безпекові заходи, дизайнер створює відповідну форму оплати, яка після реалізує front-end, Back-end розробники інтегрують API платіжної системи, робота ведеться ітераційно, з малими циклами;

в) вивчення: функціональне і навантажувальне тестування платіжного модуля, перевірка безпеки і відповідності вимогам; пілотний запуск на обмежений час;

г) завершення: у разі успішних результатів пілотного терміну – повне розгортання платіжної системи.

Перейдемо до розгляду складної зміни з використанням Lean Change Management:

1. Розуміння: розроблений інтерфейс вебзастосунку не задовольняє очікування замовника з точки зору зручності та досвіду. Переробка користувацького інтерфейсу може значно підвищити задоволеність і конкурентоспроможність продукту.

2. Опції:

а) косметичні зміни в існуючому UI;

б) часткова переробка UI з урахуванням зворотного зв'язку;

в) повна заміна UI на новий дизайн.

3. Експерименти:

а) підготовка: збір детальних вимог до нового UI у замовника та цільових користувачів, формулювання гіпотези про покращення ключових метрик після зміни UI;

б) впровадження: дизайнер створює нові макети сторінок, front-end реалізує новий UI, back-end інтегрує з існуючою бізнес-логікою;

в) вивчення: usability тестування нового UI, презентація і збір фідбеку, A/B тестування для виміру впливу зміни на метрики;

г) завершення: позитивні результати і узгоджено замовником – розгортання нового UI, у іншому випадку – доопрацювання або відмова від змін.

За результатами можна зробити висновок, що застосування Lean Change Management ефективно для впровадження незначних і середніх змін. Ітераційний підхід дозволяє швидко реалізувати зміну і мінімізувати ризики. Залучення замовника на етапі підготовки і вивчення допомагає досягти відповідності вимогам і очікуванням. Але значним недоліком є потенційна надмірна витрата часу на пілотне тестування в процесі «вивчення» простої зміни.

Для масштабних змін модель LCM показує часткову ефективність. Ключовим недоліком виявилось те, що LCM погано відстежує ризики для існуючої функціональності в рамках гнучкого процесу. Також проблемою може стати велика тривалість циклу до остаточного впровадження.

3.5 Апробація комбінованої моделі управління змінами

Застосуємо розроблену комбіновану модель управління змінами для кожної визначеної зміни.

Розглянемо незначну зміну «Додання окремої сторінки з інформацією про акції та знижки».

Етап 1. Формування команди зі змін.

Команда зі змін: замовник, керівник проєкту.

Етап 2. Ідентифікація і формування змін.

Зміна полягає у додаванні нової сторінки на вебсайт з інформацією про акції та знижки салону краси. Вона має низький пріоритет, незначну складність, не суперечить вимогам проєкту. Очікується низька вартість та незначний вплив на терміни.

Етап 3. Стратегічне планування змін.

Ціль: Додати сторінку з акціями/знижками до існуючого вебзастосунку протягом 3 робочих днів.

Завдання: керівник проєкту займається переплануванням і контролем за виконанням призначених завдань команді проєкту, бізнес-аналітик аналізує вимоги, дизайнер створює макет, розробники реалізують та інтегрують сторінку, тестувальник перевіряє на відповідність вимогам.

Відповідальні особи та терміни визначені.

Стратегія комунікації: щоденні короткі наради команди.

Етап 4. Впровадження змін.

Команда реалізує заплановані дії згідно з планом під керівництвом менеджера проєкту.

Етап 5. Оцінка результатів впровадження.

Оцінка успішності зміни за критеріями:

- дотримання термінів;
- відповідність вимогам;
- задоволеність замовника;
- виділення уроків для покращення процесу впровадження незначних змін.

Розглянемо зміну середньої складності з використанням комбінованої моделі управління змінами.

Етап 1. Формування команди зі змін.

Команда зі змін залишається: замовник, керівник проєкту.

Етап 2. Ідентифікація і формування змін.

Зміна полягає в інтеграції вебзастосунку з платіжною системою для надання користувачам можливості оплати послуг салону онлайн. Зміна має низький пріоритет, середню складність через питання безпеки й інтеграцію з існуючими модулями, відповідає цілям підвищення зручності для користувачів, вартість середня (близько \$300 з оплатою платіжної системи на рік), залучує різні ролі в проєкті, термін – близько 5 днів.

Етап 3. Стратегічне планування змін.

Ціль: інтегрувати безпечну платіжну систему до кінця травня.

Завдання: керівник проєкту займається переплануванням і контролем за виконанням призначених завдань команді проєкту, бізнес-аналітик аналізує вимоги, дизайнер – розробляє UI, розробники інтегрують, тестувальник перевіряє за вимогами і стандартами безпеки.

Відповідальні та терміни призначені.

Стратегія комунікації: щотижневі наради з командою.

Етап 4. Впровадження змін.

Команда реалізує заплановані завдання під керівництвом менеджера проєкту відповідно до плану та стратегій.

Етап 5. Оцінка результатів впровадження.

Оцінка ефективності зміни за показниками:

- виконання у визначені терміни;
- відповідність функціональним та нефункціональним вимогам;
- належний рівень безпеки платежів;
- задоволеність замовника та користувачів;
- уроки для покращення процесу впровадження подібних змін.

Перейдемо до розгляду складної зміни «Переробка інтерфейсу користувача вебзастосунку» з використанням комбінованої моделі управління змінами:

Етап 1. Формування команди зі змін.

Для реалізації такої масштабної зміни бажано залучити більше зацікавлених сторін, хоча можна обмежитись і попереднім варіантом. Команда зі змін: замовник, керівник проєкту, бізнес-аналітик, дизайнер, front-end розробники, back-end розробники, тестувальник.

Етап 2. Ідентифікація і формування змін.

Зміна полягає у повній переробці інтерфейсу користувача вебзастосунку для покращення зручності використання та зовнішнього вигляду. Її характеристики:

- високий пріоритет;
- висока складність через масштаб змін;

- відповідає стратегічним цілям продукту;
- часткова сумісність з існуючою системою;
- висока вартість від \$150 до \$500;
- значний вплив на терміни – затримка до місяця;
- потреба ретельного тестування.

Етап 3. Стратегічне планування змін.

Ціль: розробити новий сучасний і зручний UI до кінця травня.

Детальний план роботи команди: бізнес-аналітик проводить аналіз і формалізацію вимог. Дизайнери розробляють нові макети інтерфейсу, керуючись вимогами та принципами зручності використання. Менеджер презентує нові макети замовнику і у разі узгодження розробники програмно його реалізують. Тестувальники перевіряють новий інтерфейс користувача на відповідність вимогам та зручність використання, взаємодіючи з розробниками для виправлення виявлених дефектів. Після успішного тестування проходить фінальна верифікація замовником і введення в експлуатацію.

Виконавці призначені, визначаються проміжні контрольні терміни і додатково щотижневі наради прогресу та обов'язкові демонстрації.

Етап 4. Впровадження змін.

Команда реалізує розроблений план під пильним керівництвом менеджера. Відбувається регулярне спостереження за прогресом та вирішення виникаючих проблем і ризиків.

Етап 5. Оцінка результатів впровадження.

Оцінка ефективності зміни за показниками:

- дотримання вимог, бюджету і узгоджених термінів;
- покращення зручності використання, задоволеність замовника;
- виявлення проблемних місць у плануванні та впровадженні значних змін;
- рекомендації щодо вдосконалення процесу складних змін в майбутньому.

Можна впевнено сказати, що комбінована модель забезпечила достатню структурованість і керованість процесу впровадження змін навіть для рівня незначної складності. Формування спеціальної команди, ретельне планування і

моніторинг допомогли впровадити зміну вчасно та з належною якістю. Проте для незначної зміни ця модель може бути дещо надлишковою через свою деталізованість. Однак від деталізації залежить і якість, тому її спрощення або скорочення можливо призвело б до деякої невизначеності.

Щодо змін середньої та великої складності, то модель відпрацювала на відмінно. Чітке розподілення обов'язків між залученими сторонами і детальне планування допомогли впоратися зі збільшеною складністю. Регулярні наради та демонстрації забезпечили належний моніторинг та залучення зацікавлених сторін.

4 АНАЛІЗ ОТРИМАНИХ ТЕОРЕТИЧНИХ ТА ПРАКТИЧНИХ РЕЗУЛЬТАТІВ ВИКОРИСТАННЯ РОЗРОБЛЕНОЇ МОДЕЛІ

Для кращих результатів оцінювання ефективності розробленої моделі було проведено оцінку ефективності кожної з розглянутих моделей, а потім оцінку комбінованої. Умови були однакові: один і той же ІТ-проект з розробки вебзастосунку DreamBeautyParlor і однаковий набір змін з різним ступенем складності (від незначного до високого).

Застосування методології 8D виявилось доволі ефективним для незначних та середніх змін. Чіткі кроки, залучення необхідних зацікавлених сторін, якісне і своєчасне впровадження без суттєвих затримок. Єдиними недоліками стали відсутність окремих вимог замовника на початкових етапах та підвищені ризики безпеки платежів, що потребували ретельного врахування і тестування. Однак зі складною зміною виникли значні проблеми: недопрацювання на етапі проектування призвело до виникнення неврахованих ризиків і необхідності значних додаткових зусиль.

Модель Коттера виявилася неефективною для впровадження незначних змін через її масштабний характер, проте добре спрацювала для середніх та великих змін. У всіх випадках вона забезпечила системність підходу, залучення керівництва, формування бачення, опрацювання ризиків, подолання перешкод та використання зворотного зв'язку. Головним недоліком став тривалий термін реалізації кожної зміни, що призводило до уповільнення проекту.

Модель LCM ефективно працює для незначних та середніх змін. Ітераційний підхід дозволяє швидко реалізувати зміни та мінімізувати ризики. Залучення замовника на етапах підготовки і постійне вдосконалення сприяє відповідності вимогам та очікуванням. Однак виникає ризик надмірної витрати часу на пілотне тестування простих змін.

Для масштабних змін модель LCM продемонструвала часткову ефективність. Ключовим недоліком виявився недостатній рівень відстеження

ризиків для існуючої функціональності в рамках гнучкого процесу, а також тривалий цикл до остаточного впровадження.

Комбінована модель вирішила всі недоліки обраних моделей і врахувала їх переваги. У процесі управління змінами при реалізації ІТ-проєкту розробки вебзастосунку DreamBeautyParlor, вона забезпечила належну структурованість і керованість процесу впровадження змін, навіть для незначного рівня складності. Формування спеціальної команди, ретельне планування і моніторинг допомогли своєчасно та якісно впровадити зміни. Проте для незначних змін ця модель може бути дещо надлишковою через свою деталізацію, яка водночас є запорукою якості. Для середніх та великих змін модель відпрацювала на відмінно. Чіткий розподіл обов'язків між залученими сторонами та детальне планування допомогли впоратися зі збільшеною складністю. Регулярні наради та демонстрації забезпечили належний моніторинг та залучення зацікавлених сторін.

Виходячи з результатів, можна виділити наступні характеристики розробленої моделі:

- чітка структура дій;
- масштабованість;
- адаптивність;
- зрозумілість;
- простота використання;
- врахування специфіки конкретної зміни.

Комбінована модель управління змінами демонструє гарну масштабованість, оскільки її етапи можуть бути адаптовані як для невеликих, так і для великих проєктів. Для маленьких проєктів формування команди може бути менш масштабним, з меншою кількістю ключових зацікавлених осіб, тоді як для великих проєктів команда може включати більшу кількість учасників з більш складною структурою ролей. Ідентифікація та формування змін також може варіюватися за обсягом і деталізацією залежно від розміру проєкту. Стратегічне

планування та впровадження можуть бути більш простими для менших проєктів і складнішими для великих, але основні принципи залишаються однаковими.

Розроблена модель має високу адаптивність, оскільки її можна налаштовувати залежно від специфіки проєкту та вимог команди. Кроки моделі структуровані та логічно послідовні, що робить її зрозумілою для команди та зацікавлених сторін. Розробка стратегії комунікації та залучення зацікавлених сторін на ранніх етапах сприяє підвищенню зрозумілості та підтримці змін. Оскільки кожен крок включає чіткі завдання і підходи, модель легко адаптується до різних контекстів та потреб.

Модель має помірну трудомісткість у порівнянні з іншими методологіями управління змінами, такими як 8D, модель Коттера та LCM. Модель Коттера, наприклад, включає більше етапів і може бути більш трудомісткою через свою складність до розуміння і деталізацію. Lean Change Management фокусується на швидких ітераціях і постійних покращеннях, що може знижувати трудомісткість, але вимагає високої адаптивності та швидкого реагування від команди. 8D більше спрямована на вирішення проблем і може бути більш формалізованою та структурованою. Розроблена модель, хоч і вимагає певного рівня зусиль на кожному етапі, але підтримує високу якість процесу управління змінами і забезпечує баланс між деталізацією та гнучкістю, що робить її менш трудомісткою в порівнянні з деякими іншими підходами.

Комбінована модель добре враховує специфіку конкретної зміни через деталізований підхід до ідентифікації та формування змін. На етапі ідентифікації змін проводиться детальний аналіз поточних проблем та потреб, що дозволяє виявити конкретні зміни, які необхідні для проєкту. Оцінка впливу кожної зміни включає технічні, організаційні та бізнес аспекти, що забезпечує всебічне розуміння її специфіки. Стратегічне планування також передбачає врахування особливостей кожної зміни через встановлення чітких цілей та розробку відповідного плану дій. Це забезпечує належну увагу до кожної зміни та її специфіки, що підвищує ефективність впровадження.

ВИСНОВКИ

У ході виконання науково-дослідної роботи було проведено детальне вивчення складових частин процесу управління змінами при реалізації ІТ-проектів, досліджено критерії, що використовуються в процесі управління змінами, та їх вплив на ефективність виконання проекту.

Основною метою дослідження було вивчення та порівняльний аналіз існуючих методів і моделей управління змінами, а також розробка власної вдосконаленої моделі, яка враховувала б переваги попередніх розробок та усувала виявлені недоліки.

Для досягнення поставленої мети було виконано такі ключові завдання: проведено ґрунтовний аналіз наявних моделей і методик управління змінами в ІТ-проектах, оцінено їх ефективність, можливість інтеграції з іншими процесами управління проектами; досліджено критерії, які використовуються для прийняття рішень при управлінні змінами, та їхній вплив на результативність проекту; розроблено власну комбіновану модель управління змінами, що акумулювала переваги розглянутих підходів та пододала виявлені недоліки, зокрема відсутність чіткої структури дій, погану масштабованість, низьку адаптивність, високу трудомісткість та неврахування специфіки змін.

Теоретичне дослідження показало, що ефективне управління змінами є критично важливим для успішної реалізації ІТ-проектів. Було детально вивчено складові частини цього процесу, зокрема ідентифікацію та аналіз змін, планування їх впровадження, контроль та моніторинг. Це дозволило визначити основні підходи та принципи побудови дієвої моделі управління змінами в умовах мінливого проєктного середовища.

Емпіричне дослідження, під час якого розроблена комбінована модель управління змінами була апробована на практиці, продемонструвало її високу масштабованість, адаптивність та зрозумілість для всіх зацікавлених сторін. Порівняно з іншими розглянутими моделями, запропонований підхід виявив

помірну трудомісткість та забезпечив всебічний погляд на реалізацію змін з урахуванням їх специфіки.

Підсумовуючи, можна стверджувати, що в результаті виконаного дослідження було досягнуто поставлену мету шляхом розробки ефективної комбінованої моделі управління змінами в ІТ-проектах. Запропонована модель пройшла успішну апробацію та може бути рекомендована до практичного використання організаціями, що реалізують ІТ-проекти, для підвищення їх результативності в умовах невизначеності проектного середовища.

За тематикою кваліфікаційної роботи було опубліковано тези доповіді на тему «Дослідження моделей управління змінами при реалізації ІТ-проектів» на 28-му Міжнародному молодіжному форумі «Радіоелектроніка та молодь у ХХІ столітті» (16 – 18 квітня 2024 р.) [32].

Кваліфікаційна робота була виконана згідно з вимогами методичних вказівок щодо розробки та оформлення кваліфікаційної роботи другого (магістерського) рівня [33]. Пояснювальна записка оформлена за стандартом ДСТУ 3008:2015 «Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлення» [34]. Перелік джерел посилання оформлено згідно стандарту ДСТУ 8302:2015 «Бібліографічне посилання. Загальні положення та правила складання» [35].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. The standard for project management and a guide to the project management body of knowledge (PMBOK® Guide) – Seventh Edition [Текст] / USA: Project Management Institute, 2021. 370 с.

2. Управління реалізацією інвестиційних проєктів. Зміст управління інвестиційними проєктами / Електронна бібліотека Pidru4niki.com: навчальні матеріали онлайн. URL: https://pidru4niki.com/13331222/investuvannya/upravlinnya_realizatsiyeyu_investitsiynih_proektiv (дата звернення 14.04.2024)

3. Блага Н. В. Управління проєктами: навч. посібн. / Львівський державний університет внутрішніх справ. Львів: ЛДУВС, 2021. 152 с.

4. Creasey T. Definition of Change Management. *Prosci Blog*. 2022. URL: <https://www.prosci.com/blog/definition-of-change-management> (дата звернення 14.04.2024)

5. Кузьмініх В. О., Тараненко Р. А. Основи управління ІТ проєктами: навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки» / Уклад.: В. О. Кузьмініх, Р. А. Тараненко. Київ: КПІ ім. Ігоря Сікорського, 2019. 75 с.

6. Віжан К. Процес управління змінами в ІТ-компанії, або Як побороти страх роботи з чендж реквестами. *DOU: спільнота програмістів*. 2023. URL: <https://dou.ua/forums/topic/42501> (дата звернення 14.04.2024)

7. Change Advisory Board: Insights and Best Practices / Atlassian: ITSM for high-velocity teams. URL: <https://www.atlassian.com/itsm/change-management/change-advisory-board> (дата звернення 22.04.2024)

8. Schwartz B. Change Control Board: Roles, Responsibilities & Processes. *ProjectManager: project management software for business excellence*. 2022. URL: <https://www.projectmanager.com/blog/change-control-board-roles-responsibilities-processes> (дата звернення 22.04.2024)

9. ITIL Change Types: Standard vs Normal vs Emergency. *Freshworks*. URL: <https://www.freshworks.com/freshservice/change-management-software/itil-change-types> (дата звернення 22.04.2024)

10. Best ITIL Change Management with Free Templates, Samples & Checklist. OCM Solution. URL: <https://www.ocmsolution.com/itil-change-management-process-guide> (дата звернення 22.04.2024)

11. Що таке управління вимогами: визначення, переваги, найкращі інструменти / *Visure Solutions: Requirements Management and Traceability Essential Guide*. URL: <https://visuresolutions.com/uk/requirements-management-traceability-guide/requirements-management> (дата звернення 22.04.2024)

12. Секрети ефективного менеджменту проєктів в ІТ. DAN IT Education. 2023. URL: <https://dan-it.com.ua/uk/blog/sekreti-efektivnogo-menedzhmentu-proiektiv-v-it/#4> (дата звернення 22.04.2024)

13. Гнучке управління проєктами: методичні рекомендації до самостійних робіт з дисципліни «Гнучке управління проєктами» для здобувачів вищої освіти «Магістр» за спеціальністю 051 «Економіка» / Розроб.: Ю. В. Богоявленська. Житомир: Держ. ун-т «Житомирська політехніка», 2021. 59 с.

14. Кузьмініх В. О., Коваль О. В., Тараненко Р. А. Моделі та засоби управління ІТ-проєктами: навч. посіб. для студ. спеціальності 121 «Інженерія програмного забезпечення» / Уклад.: В. О. Кузьмініх, О. В. Коваль, Р. А. Тараненко. Київ: КПІ ім. Ігоря Сікорського, 2023. 222 с.

15. Князева Н. Як проджект-менеджеру визначати ключові метрики проєкту: інструкція від Head of IoT у Kyivstar. 2021. URL: <https://laba.ua/blog/2691-kak-prodzhekt-menedzheru-opredelyat-klyuchevye-metriki-proekta> (дата звернення 22.04.2024)

16. Лаговська О. А., Грабчук І. Л., Лоскоріх Г. Л. Класифікація ризиків ІТ-підприємств: обліковий аспект. *Проблеми теорії та методології бухгалтерського обліку, контролю і аналізу*. № 3 (47) С. 28-32.

17. Войтенко О. С. Управління проєктами: навч. посіб. / О.С. Войтенко. Київ: КНУБА, 2020. 276 с.

18. Zosym M. Методологія восьми дисциплін (Eight Disciplines Methodology – 8D). 2023. URL: <https://www.maxzosim.com/eight-disciplines-methodology-8d> (дата звернення 22.04.2024)
19. Plan, Do, Check, Act (PDCA) / *Lean Enterprise Institute*. URL: <https://www.lean.org/lexicon-terms/pdca> (дата звернення 22.04.2024)
20. Zosym M. 8-кроковий процес змін Джона Коттера (John Kotter's eight steps). 2022. URL: <https://www.maxzosim.com/john-kotters-eight-steps> (дата звернення 22.04.2024)
21. Lean Change Management / *Project Management Institute*. URL: <https://www.pmi.org/disciplined-agile/wow/leanchange> (дата звернення 22.04.2024)
22. Поган О. Що таке Jira і як з нею працювати. *Tech skills блог IAMPM*. 2023. URL: <https://iampm.club/ua/blog/shho-take-jira-i-yak-z-neyu-praczyuvati> (дата звернення 23.04.2024)
23. Else J. What is ServiceNow? A Complete Guide. *FlyForm*. 2023. URL: <https://flyform.com/insights/articles/what-is-servicenow> (дата звернення 23.04.2024)
24. Microsoft Power Automate – Process Automation Platform. URL: <https://www.microsoft.com/en-us/power-platform/products/power-automate> (дата звернення 23.04.2024)
25. What is Zendesk? Powerful customer experience software. URL: <https://www.zendesk.com/what-is-zendesk> (дата звернення 23.04.2024)
26. About – Bugzilla. URL: <https://www.bugzilla.org/about> (дата звернення 23.04.2024)
27. Mantis Bug Tracker / From Wikipedia, the free encyclopedia. URL: https://en.wikipedia.org/wiki/Mantis_Bug_Tracker (дата звернення 23.04.2024)
28. RiskWatch: Global Leader in Risk and Compliance Management Solutions. URL: <https://www.riskwatch.com> (дата звернення 23.04.2024)
29. What is GRC? How to Protect Your Business. URL: <https://www.sap.com/products/financial-management/what-is-grc.html> (дата звернення 23.04.2024)

30. Буднік М. М., Іванова Я. Ю. Підходи та моделі управління змінами на підприємстві. *Бізнес Інформ*. 2020. №10. С. 370–376. DOI: 10.32983/2222-4459-2020-10-370-376

31. Добровська Л. М., Аверьянова О. В. Управління ІТ-проектами в Microsoft Project: Комп'ютерний практикум: навчальний посібник для студентів спеціальності 122 «Комп'ютерні науки» для всіх спеціалізацій / Л.М. Добровська, О.В. Аверьянова; КПІ ім. Ігоря Сікорського. – Київ: КПІ ім. Ігоря Сікорського, 2020 – 152 с.

32. Сурков Є. М. Дослідження моделей управління змінами при реалізації іт-проектів / Є. М. Сурков, наук. керівник д.т.н. проф. К. Е. Петров // *Радіоелектроніка та молодь у ХХІ столітті. Конференція «Інформаційні інтелектуальні системи»: матеріали 28-го Міжнар. молодіж. форуму (16–18 квіт. 2024 р.)* / М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків : ХНУРЕ, 2024. – Т. 6. – С. 268-269. – DOI: 10.30837/IYF.IIS.2024.268.

33. Методичні вказівки щодо розробки та оформлення кваліфікаційної роботи другого (магістерського) рівня вищої освіти за освітньо-науковою програмою «Управління проектами в галузі інформаційних технологій» / Упоряд.: Петров К.Е., Левикін В.М., Чалий С.Ф., Євланов М.В., Міхнов Д.К., Міхнова А.В., Чала О.В. – Харків: ХНУРЕ, 2024. – 24 с.

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

35. ДСТУ 8302:2015. Бібліографічне посилання. Загальні положення та правила складання. [Чинний від 01.07.2016]. Вид. офіц. Київ: ДП «УкрНДНЦ», 2016. 20 с.