

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

Кваліфікаційна наукова праця  
на правах рукопису

Близнюкова Ірина Олександрівна

УДК 005.8:005.95:004.9

**ДИСЕРТАЦІЯ**

**ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ КРЕАТИВНОГО УПРАВЛІННЯ  
КОМАНДАМИ ІТ-ПРОЄКТІВ**

122 – Комп'ютерні науки

12 – Інформаційні технології

Подається на здобуття наукового ступеня доктора філософії

Дисертація містить результати власних досліджень.

Використання ідей, результатів і текстів інших авторів

мають посилання на відповідне джерело

\_\_\_\_\_ І.О Близнюкова

Науковий керівник Данченко Олена Борисівна, доктор технічних наук, професор

Черкаси – 2023

## АНОТАЦІЯ

**Близнюкова І.О. Інформаційна технологія креативного управління командами ІТ-проектів.** – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 122 Комп'ютерні науки (12 Інформаційні технології). Черкаський державний технологічний університет, Міністерство освіти і науки України, Черкаси, 2023.

В дисертаційній роботі вирішено актуальне науково-прикладне завдання, що полягало в розробці та вдосконаленні існуючих моделей, методів та інформаційних засобів управління командами ІТ-проектів з інноваційною складовою.

В роботі було проаналізовано визначення та особливості сучасних ІТ-проектів, досліджено рівні їхньої інноваційності, проаналізовано інформаційні засоби управління такими проектами. Проведено огляд підходів до управління командами ІТ-проектів та проаналізовано гнучкі технології розробки програмного продукту. Розглянуто сучасні технології креативного управління ІТ-проектами, що мають у собі інноваційну складову. Це технології дизайн-мислення, MVP, та Proof of Concept.

Проведений аналіз інформаційних технологій управління проектами показав, що використання автоматизованих засобів управління проектами призводить до знеособлення та віддалення учасників проектів. В роботі показано, що у клієнт-орієнтованому та ціннісно-орієнтованому середовищі відсутність емпатії команди до проблем замовника може призвести до провалу проекту.

В роботі показано, що клієнт-орієнтовні техніки гнучкого управління ІТ-проектами, а саме технологія дизайн-мислення, концепція створення мінімально-життєздатного продукту та концепція Proof of concept є найбільш ефективним середовищем для розробки ІТ-проектів з інноваційною складовою.

Автором розроблено концептуальну модель креативного управління командами ІТ-проектів з інноваційною складовою.

В роботі розроблено семіотичну модель управління командою ІТ-проєкту та формальну ситуативну модель, яка слугує їй базисом.

Потреба у вдосконаленні та розробці нових методів з'явилася тому, що команди, які потрібні для реалізації ІТ-проєктів з інноваційною складовою, мають бути креативними, самокерованими та саморганізованими. Це суттєво відрізняє їх від класичних методів формування та управління командами ІТ-проєктів, у яких головними чинниками та критеріями успішності були: спланованість, керованість, підпорядкованість. А головними критеріями відбору були виключно професійні здібності претендентів у члени команди.

В роботі вдосконалено метод формування команди ІТ-проєкту, який у процесі відбору та формування команди спирається на креативні здібності претендентів. Автором запропонована процедура формування мінімально-життєздатної команди, в якій будуть сформовані сприятливі умови для розвитку процесів самокерованості та самоорганізації.

В роботі запропоновано та обґрунтовано застосування генетичного алгоритму в процесах формування команди ІТ-проєкту. Показано, що така задача формування є багатокритеріальною, а в рішенні відсутній головний критерій, який зазвичай вважається критерієм успішності. Тому особа, що приймає рішення, має орієнтуватися на власний досвід, уміння та компетентності.

Саме для виключення суб'єктивного впливу експерта, що має приймати рішення, і було запропоновано використання генетичного алгоритму.

Також було розроблено метод креативного управління командою ІТ-проєкту, який включає в себе продукування професійного зростання членів команди; розвинення їх емпатії (групової емпатії); розвиток емоційного інтелекту через застосування емоційного командного менеджменту; та досягнення спільної командної мети, яка включає в себе індивідуальні цілі кожного члена команди. Особливістю методу є те, що зазначені кроки стануть не лише здобутками членів команди, а й єдиною можливим важелем у забезпеченні успішного завершення ІТ-проєктів з інноваційною складовою.

На прикладі ІТ-проєкту було проаналізовано та визначено вимоги до технічних та психологічних компетентностей членів команди. Крім того, презентований підхід до створення команди управління ІТ-проєктом з використанням матриці відповідальності RACI. В четвертому розділі представлено результати розробки інформаційної технології управління креативною командою ІТ-проєкту, яка містить у собі результати дисертаційного дослідження.

Автором представлено розроблений веб-додаток, в якому реалізовано генетичний алгоритм, за допомогою якого менеджерам проєкту пропонується проєктне рішення щодо формування креативної команди ІТ-проєкту.

Практичне значення результатів роботи підтверджується впровадженням їх в процеси управління ІТ-проєктами шляхом використання розробленої інформаційної технології та веб-додатку в українських ІТ-компаніях.

Використання запропонованих автором моделей та методів креативного управління командами ІТ-проєктів дозволяє підвищити імовірність успішного завершення ІТ-проєктів з інноваційною складовою.

**Ключові слова:** ІТ-проєкт, команда ІТ-проєкту, емпатія, дизайн-мислення, мінімально-життєздатний продукт, мінімально-життєздатна команда, креативне управління командою ІТ-проєкту, управління ІТ-проєктами.

## ABSTRACT

***Blyzniukova I.O. Information technology for creative management of IT-project teams. - Qualifying scientific work on manuscript rights.***

Dissertation for obtaining the scientific degree of Doctor of Philosophy in the specialty 122 Computer Sciences (12 Information Technologies). Cherkasy State Technological University, Ministry of Education and Science of Ukraine, Cherkasy, 2023.

In the dissertation work, an actual scientific and applied task was solved, which consisted in the development and improvement of existing models, methods and information tools for managing IT-project teams with an innovative component.

The paper analyzed the features of modern IT-projects, the levels of their innovativeness were investigated, and analyzed the means of managing such projects. A review of IT-project team management was conducted and teams that use flexible software product development technologies were analyzed. Modern technologies of creative management of IT-projects, which have an innovative component, are considered. These are design thinking, MVP, and Proof of Concept technologies.

The analysis of project management information technologies showed that the use of automated project management tools leads to depersonalization and alienation of project participants. The work shows that in client-oriented and value-oriented environment, the lack of empathy of the team to the customer's problems can lead to the failure of the project.

The work shows that client-oriented techniques of flexible management of IT-projects: design thinking technology, the technology of creating a minimum viable product and the concept of Proof of concept are the most effective environment for the development of IT-projects with an innovative component.

The author has developed the conceptual model of creative management of the IT-project team with an innovative component.

The work developed: a semiotic model of IT-project team management and a formal situational model, which serves as its basis.

The need for improvement and development of new methods appeared because the teams needed to develop projects with an innovative component must be creative, self-directed and self-organized. This significantly distinguishes them from the classic methods of forming and managing IT-project teams, in which the main factors and success criteria were: planning, controllability, and subordination. And the main selection criteria were solely the professional abilities of the applicants as team members.

The work improved the method of forming an IT-project team, which relies on the creative abilities of applicants in the process of selecting and forming a team. The author proposed a procedure for forming a minimally viable team, which will create

favorable conditions for the development of self-management and self-organized processes.

The paper proposes and substantiates the application of the genetic algorithm in the processes of IT-project team formation. It is shown that such a formation task is multi-criteria, and the solution lacks the main criterion, which is usually considered a success criterion. Therefore, the decision-maker should focus on his own experience, skills and competences.

It is to exclude the subjective influence of the expert who has to make a decision that the use of a genetic algorithm was proposed.

The method of creative management of the IT-project team was developed, which includes the production of professional growth of team members; development of their empathy (group empathy); development of emotional intelligence through the use of emotional team management; and achieving a common team goal that includes the individual goals of each team member. The peculiarity of the method is that the specified steps will become not only the achievements of the team members, but also the only possible leverage in ensuring the successful completion of IT-projects with a innovative component.

Using the example of IT-projects the requirements for technical and psychological competencies of team members were analyzed and determined. In addition, an approach to creating an IT-project management team using the RACI responsibility matrix is presented. The fourth chapter presents the results of the development of information technology management by the creative team of the IT-project, which includes the results of the dissertation research.

The author presents a developed web-application in which a genetic algorithm is implemented, with the help of which project managers are offered a project decision regarding the formation of an IT-project team.

The practical significance of the work results is confirmed by their implementation in IT-project management processes through the use of the developed information technology and web-application.

Using the models and methods of creative management of IT-project teams proposed by the author increases the probability of successful completion of IT-projects with innovative component.

**Keywords:** IT-project, IT-project team, empathy, design thinking, minimum viable product, minimum viable team, IT-project team creative management, IT-project management.

### Список опублікованих праць за темою дисертації:

– *статті у наукових фахових виданнях України, в яких опубліковані основні наукові результати дисертації:*

1. Близнюкова І.О., Семко І.Б., Кійко С.Г. Огляд сучасних методологій управління командами ІТ-проектів. *Управління розвитком складних систем*. Київ, 2020. №43. С. 60-66. (0,4 д. а.). DOI: 10.32347/2412-9933.2020.43.60-66. URL: <http://mdcs.knuba.edu.ua/article/view/219835>. Фахове видання України категорії Б (включене до Google Scholar).

*Особистий внесок автора полягає у визначенні особливостей та сучасних тенденцій в управлінні ІТ-проектами та становить 0,3 друк. арк.*

2. Близнюкова І.О. Концепція створення мінімально життєздатного продукту та дизайн-мислення в управлінні командою ІТ-проекту / Близнюкова І.О. Тесленко П.О., Данченко О.Б., Меленчук В.М. *Вісник національного технічного університету «ХПІ» : Зб.наук.пр. Серія : Стратегічне управління, управління портфелями програмами та проектами*. Харків: НТУ «ХПІ», 2021. №2(4). С. 11 – 17. (0,4 д. а.). DOI: 10.20998/2413-3000.2021.4.2. URL: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/52329>. Фахове видання України категорії Б (включене до Index Copernicus, WorldCat, Directory of Open Access Scholarly Resources, Open Access Infrastructure for Research in Europe, Google Scholar, Ulrich's Periodicals Directory, Polska Bibliografia Naukowa, Bielefeld Academic Search Engine, Наукова періодика України).

*Особистий внесок автора полягає у визначенні основних підходів до управління командами ІТ-проектів з використанням дизайн-мислення та технології MVP та становить 0,3 друк. арк.*

3. Близнюкова І.О., Тесленко П.О., Малахова Д.О. Особливості формування команди управління ІТ-проектом. *Вісник національного технічного університету «ХПІ» : Зб.наук.пр. Серія : Стратегічне управління, управління портфелями програмами та проектами*. Харків: НТУ «ХПІ», 2022. № 2 (6). С. 14 – 20. (0,4 д. а.).DOI:10.20998/2413-3000.2022.6.3.URL: <http://pm.khpi.edu.ua/article/view/262312>. Фахове видання України категорії Б (включене до Index Copernicus, WorldCat, Directory of Open Access Scholarly Resources, Open Access Infrastructure for Research in Europe, Google Scholar, Ulrich's Periodicals Directory, Polska Bibliografia Naukowa, Bielefeld Academic Search Engine, Наукова періодика України).

*Особистий внесок автора полягає у визначенні основних підходів до формування команди ІТ-проекту та становить 0,3 друк. арк.*

4. Близнюкова І.О. Метод формування креативної команди ІТ-проекту. *Вісник національного технічного університету «ХПІ» : Зб.наук.пр. Серія : Стратегічне управління, управління портфелями програмами та проектами*. Харків: НТУ «ХПІ», 2023. № 1(7). С. 12–18. (0,45 д. а.). <http://pm.khpi.edu.ua/article/view/286393>. Фахове видання України категорії Б (включене до Index Copernicus, WorldCat, Directory of Open Access Scholarly Resources, Open Access Infrastructure for Research in Europe, Google Scholar, Ulrich's Periodicals Directory, Polska Bibliografia Naukowa, Bielefeld Academic Search Engine, Наукова періодика України).

*Особистий внесок автора полягає у визначенні креативної команди ІТ-проекту та методу її формування та становить 0,45 друк. арк.*

5. Blyznyukova, I., Teslenko, P. (2023). Formation of a minimum viable IT project team using the genetic algorithm. *Technology Audit and Production Reserves*, 2(2(70)). (0,45 д. а.). DOI: 10.15587/2706-5448.2023.277930. URL: <https://journals.uran.ua/tarp/article/view/277930>. Фахове видання України категорії Б (включене до Google Scholar).



*Особистий внесок автора полягає у розробці генетичного алгоритму задля формування мінімально життєздатної команди IT-проєкту та становить 0,35 друк. арк.*

**– наукові праці, що засвідчують апробацію матеріалів дисертації:**

6. Близнюкова І.О., Данченко О.Б., Тесленко П.О. Підвищення рівня задоволеності замовника проєкту з використанням методології дизайн-мислення. *Тези доповідей V Міжнародної науково-практичної конференції РЗМ-2020*. Відповідальний за випуск Тесленко П.О. Одеса: Бондаренко М.О., 2020. С. 22 – 26. (0,3 д. а.).

[URL:https://drive.google.com/drive/u/2/folders/1ubmCVXUPHkEeIczv52D3NFPQL\\_14Kr31](https://drive.google.com/drive/u/2/folders/1ubmCVXUPHkEeIczv52D3NFPQL_14Kr31).

*Особистий внесок автора полягає у визначенні основних підходів до застосування методології дизайн-мислення для забезпечення задоволеності замовника проєкту та становить 0,2 друк. арк.*

7. Близнюкова І.О., Данченко О.Б., Тесленко П.О. Аналіз сучасних визначень IT-проєктів. Управління проєктами у розвитку суспільства. Тема: «Управління проєктами в умовах пандемії COVID-19»: тези доповідей. Київ : КНУБА, 2021. С.100-104. (0,25 д. а.).

[URL:http://eprints.kname.edu.ua/58522/1/%D0%A2%D0%B5%D0%B7%D0%B82021.pdf](http://eprints.kname.edu.ua/58522/1/%D0%A2%D0%B5%D0%B7%D0%B82021.pdf).

*Особистий внесок автора полягає в обґрунтування визначення IT-проєкту яке використовується в дисертаційному дослідженні та становить 0,2 друк. арк.*

8. Близнюкова І.О., Тесленко П.О., Данченко О.Б. Інструменти управління IT-проєктами з інноваціями. *Збірка тез VIII Міжнародної НТК «Інформатика. Культура. Технології» ІКТ-2021*. Одеса : ІКС, 2021. С. 84-86. (0,15 д. а.). [URL: http://ics\\_conf.tilda.ws/ict\\_eng](http://ics_conf.tilda.ws/ict_eng).

*Особистий внесок автора полягає у визначенні основних інструментів для управління IT-проєктами, що мають інноваційну складову та становить 0,1 друк. арк.*

9. Близнюкова І.О. Технології дизайн-мислення в управлінні командою ІТ-проєкту / Близнюкова І.О., Данченко О.Б., Тесленко П.О., Куваєва В.І. *Управління проєктами: стан та перспективи : Матеріали XVII Міжнародної науково-практичної конференції*. Миколаїв : Видавець Торубара В.В., 2021. С. 13 – 14. (0,1 д. а.).

*Особистий внесок автора полягає в обґрунтуванні застосування технології дизайн-мислення для управління командою ІТ-проєкту та становить 0,08 друк. арк.*

10. Близнюкова І.О. Концептуальна модель креативного управління командою ІТ проєкту/ Близнюкова І.О., Данченко О.Б., Тесленко П.О., Заруцький С.О. *Тези доповідей VI Міжнародної науково-практичної конференції РЗМ-2021*. Відповідальний за випуск Тесленко П.О. Одеса: ІШПР, 2021. С. 81-83. (0,15 д. а.).  
[URL:https://drive.google.com/drive/u/2/folders/1A19cjGj7O0NouNq9exxMQfd19AugkhLg](https://drive.google.com/drive/u/2/folders/1A19cjGj7O0NouNq9exxMQfd19AugkhLg).

*Особистий внесок автора полягає у розробці концептуальної моделі креативного управління командою ІТ проєкту та становить 0,1 друк. арк.*

11. Близнюкова І.О., Тесленко П.О., Харута В.С. Продуктування навиків емпатії у команди управління ІТ-проєктом. *Управління проєктами у розвитку суспільства. Тема: «Управління проєктами в очікуванні глобальної кризи»: тези доповідей*. Київ: КНУБА, 2022. С. 75 – 77. (0,1 д. а.).

*Особистий внесок автора полягає у визначенні основних підходів до підвищення рівня емпатії членів команди ІТ-проєктів та становить 0,08 друк. арк.*

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	13
ВСТУП.....	15
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ .....	22
1.1 Сучасні моделі, методи, стандарти управління ІТ-проєктами .....	22
1.1.1 Особливості сучасних ІТ-проєктів .....	22
1.1.2 Аналіз особливостей управління ІТ-проєктами.....	28
1.2. Особливості управління командами ІТ-проєктів .....	41
1.3 Аналіз інформаційних технологій управління ІТ-проєктами .....	51
1.4 Постановка задачі дослідження.....	54
1.5 Висновки до першого розділу .....	55
Список використаних джерел до розділу 1 .....	57
РОЗДІЛ 2 МОДЕЛІ КРЕАТИВНОГО УПРАВЛІННЯ КОМАНДАМИ ІТ-ПРОЄКТІВ .....	63
2.1 Методологія та архітектура дослідження.....	63
2.1.1 Архітектура дисертаційного дослідження.....	63
2.1.2 Технологія дизайн-мислення .....	68
2.1.3 Технологія створення мінімально життєздатного продукту .....	72
2.1.4 Методика Proof of concept.....	73
2.1.5 Креативне управління ІТ-проєктами.....	74
2.2 Концептуальна модель креативного управління командою ІТ-проєкту.....	76
2.3 Обґрунтування гомоморфного перетворення моделі креативного управління командою ІТ-проєкту .....	83
2.4 Семіотична модель управління командою ІТ-проєкту .....	87
2.5 Висновки до другого розділу.....	94

Список використаних джерел до розділу 2 .....	96
<b>РОЗДІЛ 3 МЕТОДИ КРЕАТИВНОГО УПРАВЛІННЯ КОМАНДАМИ ІТ-ПРОЄКТІВ .....</b>	<b>103</b>
3.1 Удосконалення методу формування креативної команди ІТ-проєкту з інноваційною складовою .....	103
3.2 Застосування генетичного алгоритму для мінімізації кількісного складу команди ІТ-проєкту .....	113
3.3 Метод креативного управління командою ІТ-проєкту з інноваційною складовою .....	121
3.4 Висновки до третього розділу .....	129
Список використаних джерел до розділу 3 .....	131
<b>РОЗДІЛ 4 ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ КРЕАТИВНОГО УПРАВЛІННЯ КОМАНДОЮ ІТ-ПРОЄКТУ З ІННОВАЦІЙНОЮ СКЛАДОВОЮ .....</b>	<b>137</b>
4.1 Формалізація даних для застосування інформаційної технології .....	137
4.1.1 Опис проєкту .....	137
4.1.2 Формалізація технічних компетентностей команди проєкту .....	139
4.1.3 Формалізація психологічних компетентностей команди проєкту .....	141
4.2 Приклад формування команди ІТ-проєкту .....	143
4.3 Розробка інформаційної технології креативного управління командою ІТ-проєкту .....	150
4.4. Реалізація розроблених моделей та методів у вигляді інформаційної системи .....	155
4.5 Висновки до 4 розділу .....	162
Список використаних джерел до розділу 4 .....	164
<b>ВИСНОВКИ .....</b>	<b>166</b>
<b>ДОДАТКИ .....</b>	<b>169</b>

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ІТ – інформаційні технології

МЖП – мінімально життєздатний продукт

MVP – minimum viable product

ТЗ – технічне завдання

PMI – Project Management Institute

IEEE – Institute of Electrical and Electronics Engineers

UML – Unified Modeling Language

SWEBOK – Software Engineering Body of Knowledge

PMBok – Project Management Body of Knowledge

ПП – програмні проекти

LOE – level-of-effort

ЖЦП – життєвий цикл проекту

ITSM – IT-Service Management

ТСПП – технологія створення програмного продукту

ТСПП – технологія створення програмного продукту

ПЗ – програмне забезпечення

MSN – Microsoft Network

RUP – Rational Unified Process

MSF – Microsoft Solutions Framework

XP – Extreme Programming

PSP – Personal software process

FIRO-B – Fundamental Interpersonal Relations Orientation Behavior

ККІТП – координатор команди ІТ-проекту

T2M – time-to-market

ППП – продукт проекту

HCD – Human-centered design

PoC – Proof of concept

WBS – work breakdown structure

OBS – organizational breakdown structure

RBS – resource breakdown structure

SCi – complete situation

SRi – Current situation

OSSi – One step solution

МЖК – мінімально життєздатна команда

ДМ – дизайн-мислення

IoT – Internet of things

ІС – інформаційна система

MVT – minimum viable team

СУБД – система управління базами даних

ML – machine learning

## ВСТУП

### Актуальність теми

Процеси формування та управління командою будь-якого проєкту є слабо структурованими та невизначеними. Обрання критеріїв та стратегії формування команди залежить від значної кількості факторів: від цілей та задач проєкту, від бачення його учасників, взаємодії стейкхолдерів, галузі застосування продукту, від особистісних характеристик замовника, керівника команди проєкту та інших ключових учасників.

ІТ-галузь, як будь-яка інша, має свою специфіку. З одного боку, існують розроблені та протестовані методики створення та управління ІТ-проєктами: від водоспадної до гнучкої. З іншого боку, перелік робіт ІТ-проєктів є, зазвичай, жорстко формалізованим: збір вимог, формування ТЗ, розробка архітектури, кодування, тестування, впровадження продукту у замовника. Цей перелік робіт формує перелік функціональних вимог та обмежень до ролей майбутньої команди проєкту, яка зазвичай формується з персоналу ІТ-компанії та управляється за прийнятою в компанії методологією.

Критерії оцінки успішності проєкту неухильно зміщуються в бік забезпечення задоволеності його головних стейкхолдерів. Для цього розробляють нові, або застосовують раніше не використовувані засоби. Серед них можемо відзначити методику дизайн-мислення та концепцію МЖП. ІТ-проєкт - це завжди командна робота, в якій присутні дві ролі: «програміст» і «споживач». «Програміст» з використанням інтелектуального капіталу та когнітивних процесів створює програмний продукт, при цьому використовує інноваційні рішення, та передає готовий продукт «споживачеві».

Для успішного завершення таких проєктів з високою долею інновацій, необхідна команда, що самоорганізується та самокерується. Така команда сама приймає рішення щодо структури, методології та форми реалізації проєкту. У традиційних командах та великих ІТ-компаніях фазі реалізації передують планування (навіть, якщо мова йде про гнучкі технології розробки). Чим більше

приймається рішень щодо самої команди, концепту майбутньої розробки, тим більш така команда обмежена, тим менш вона керує власним простором. Це обмежує можливість самоорганізації такої команди та перешкоджає управлінню проектами з високою часткою інновацій.

У зв'язку з цим виникає важлива науково-прикладна задача щодо ефективного управління командами ІТ-проектів з високою часткою інновацій в переліку їх проектних рішень за рахунок розробки нових моделей, методів та інформаційних засобів креативного та гнучкого управління такими проектами.

Така проблема існує в діючих ІТ-компаніях та заважає успішному завершенню проектів зазначеної категорії. Вирішенню цієї проблеми передують її докладне вивчення та визначення супутніх перешкод. Це потребує аналізу існуючих засобів управління та формування пропозицій, спираючись на узагальнення існуючого досвіду.

За результатами проведеного аналізу публікацій була виявлена відсутність наукових та практичних рекомендацій щодо застосування технології дизайн-мислення для управління ІТ-проектом, та з використанням технології створення мінімально-життєздатного продукту в якості базисного елементу управління.

Тому актуальним є застосування технології дизайн-мислення та концепції створення МЖП до процесів управління командою ІТ-проекту з інноваційною складовою задля забезпечення успішного ІТ-проектів через створення креативної команди.

Науковими дослідженнями в області управління командами проекту займалися такі вітчизняні фахівці, як Бушуєв С.Д., Колеснікова К.В., Морозов В.В., Прокопенко Т.О., Бушуєва Н.С., Бедрій С.Д., Бушуєв Д.А., Кузьмінська Ю.М., Рач В.А.

Значний вклад в розвиток моделей, методів та засобів управління ІТ-проектами внесли Чумаченко І.В., Кононенко І.В., Цюцюра С.В., Данченко О.Б., Тесленко П.О.



Креативними технологіями управління проєктами займались Бушуев С.Д., Бушуева Н.С., Бабаев И.А., Яковенко В.Б., Гриша Е.В., Дзюба С.В., Войтенко А.С., Кузьмінська Ю.М., Вереніч О.В., Данченко О.Б., Польшаков І.В.

**Зв'язок роботи з науковими програмами, планами, темами.**

Дисертаційне дослідження виконано в рамках науково-дослідної роботи «Розробка інформаційних технологій цифрової трансформації соціо-економічних систем» (№ державної реєстрації 0120U100963, 2020 - 2023 р.р.) відповідно до тематичного плану науково-дослідних робіт Черкаського державного технологічного університету, в якій автор була виконавцем.

**Мета і завдання дослідження.** *Метою дисертаційної роботи є підвищення ефективності управління командою ІТ-проєктів з інноваційною складовою за рахунок розробки та вдосконалення креативних моделей, методів та інформаційних засобів управління з використанням техніки дизайн-мислення та концепції створення МЖП.*

Для досягнення вказаної мети в дисертаційній роботі виділено наступні задачі дослідження:

- аналіз предметної області, а саме особливостей ІТ-проєктів з інноваційною складовою та особливостей управління ними, огляд сучасних моделей, методів та інформаційних засобів управління командами таких проєктів;
- розробка концептуальної моделі креативного управління командою ІТ-проєкту з інноваційною складовою;
- розробка семіотичної моделі управління командою ІТ-проєкту з інноваційною складовою;
- удосконалення методу формування креативної команди ІТ-проєкту з інноваційною складовою;
- розробка методу креативного управління командою ІТ-проєкту з інноваційною складовою;
- вдосконалення термінологічної основи управління креативною командою ІТ-проєкту з інноваційною складовою;

- розробка інформаційної технології креативного управління командою ІТ-проєкту з інноваційною складовою;
- застосування на практиці розроблених інструментів креативного управління командою ІТ-проєкту з інноваційною складовою.

**Об’єкт дослідження** – процеси управління командами ІТ-проєктів з інноваційною складовою.

**Предмет дослідження** – моделі, методи та інформаційні засоби креативного управління командами ІТ-проєктів з інноваційною складовою.

**Методи дослідження.** Науковим і теоретичним підґрунтям дисертаційної роботи стали наукові основи управління ІТ-проєктами, фахова література, наукові праці вчених, практики управління командами. В роботі було використано: системний підхід для дослідження та визначення характеристик процесів управління командою, порівняльний аналіз для формалізації показників системи управління командою ІТ-проєктів, методи управління проєктами для ідентифікації та оцінки необхідних компетентностей претендентів до команди, прогнозування для визначення мінімально-життєздатної команди проєкту, оптимізаційні методи, генетичний алгоритм та технологія дизайн-мислення для розробки підходів та методів формування ефективної команди ІТ-проєкту, ситуаційне та семіотичне моделювання для побудови моделі креативного управління командою ІТ-проєкту. Інформаційним підґрунтям роботи є дані статистики ІТ-галузі.

**Наукова новизна одержаних результатів.**

**Вперше:**

- розроблено концептуальну модель креативного управління командами ІТ-проєктів з інноваційною складовою, яка базується на технологіях розробки продукту проєкту: дизайн-мислення та технології MVP, що дає змогу сформувати креативну, гнучку, мінімально-життєздатну команду управління ІТ-проєктами з інноваційною складовою;
- розроблено метод креативного управління командою ІТ-проєкту з інноваційною складовою, що включає в себе створення мінімально

життєздатної команди проєкту, спираючись на технологію дизайн-мислення та MVP, що на відміну від існуючих враховує індивідуальні та групові пріоритети членів команди, емпатію та емоційний інтелект.

***Одержали подальший розвиток:***

- метод формування команди IT-проєкту з інноваційною складовою на основі використання генетичного алгоритму задля створення мінімально-життєздатної команди, який на відміну від існуючих враховує не єдиний інтегральний показник компетентностей команди проєкту, а набір показників, який відображає поточний стан команди та проєкту;
- семіотична модель управління командою IT-проєкту з інноваційною складовою, яка створена на базі формальної ситуаційної моделі і, на відміну від існуючих, дозволяє зміну набору правил через базові елементи, семантичні та синтаксичні правила, аксіоми, що дозволить гнучко вносити зміни як в склад команди проєкту, так і в процес прийняття рішень щодо управління командою на протязі ЖЦП.

***Удосконалено:***

- термінологічну основу креативного управління командою IT-проєктів з інноваційною складовою, дано визначення: *«Координатор команди IT-проєкту»*, *«Креативна команда IT-проєкту»* та *«Процес створення креативної команди IT-проєкту»* задля ефективного застосування на практиці розроблених моделей та методів управління IT-проєктами з інноваційною складовою.

**Практичне значення одержаних результатів**

Практичне значення результатів дисертаційного дослідження підтвердилися при застосування результатів дослідження при управлінні IT-проєктом на практиці через розроблену інформаційну технологію управління креативною командою.

Практичне значення результатів роботи підтверджується впровадженням їх в практику управління IT-проєктами в компанії «Visual Craft Inc.», м. Черкаси (акт використання результатів від 05.11.2022), в ТОВ «Фірма «МегаСтайл», м. Черкаси

(акт впровадження № 57 від 12.06.2023), в ТОВ «МІФ Проджектс», м. Кропивницький (акт впровадження від 20.04.2023) (додаток А).

Результати роботи можуть бути використані в ІТ-компаніях, що займаються розробкою інноваційних програмних продуктів та інформаційних систем, які використовують гнучкі засоби управління.

### **Особистий внесок здобувача**

Всі наукові результати, що представлені до захисту, одержані здобувачем самостійно. У статтях, опублікованих у співавторстві, особисто дисертанту належать: визначено особливості та сучасні тенденції в управлінні ІТ-проєктами та обґрунтовано визначення ІТ-проєкту, яке використовується в дисертаційному дослідженні [1, 7]; сформовано основні підходи до управління командами ІТ-проєктів з використанням дизайн-мислення та технології MVP [2]; визначено критерії до формування команди ІТ-проєкту [3]; розроблено генетичний алгоритм для формування мінімально життєздатної команди ІТ-проєкту [5]; розроблено основні підходи до застосування методології дизайн-мислення для забезпечення задоволеності замовника проєкту [6]; розроблено інструменти для управління ІТ-проєктами, що мають інноваційну складову [8]; обґрунтовано застосування технології дизайн-мислення для управління командою ІТ-проєкту [9]; розроблено концептуальну модель креативного управління командою ІТ проєкту [10]; запропоновано підходи до підвищення рівня емпатії членів команди ІТ-проєктів [11].

### **Апробація результатів дисертації**

Результати дисертаційних досліджень обговорювалися на міжнародних науково-технічних конференціях: V та VI Міжнародній науково-практичній конференції РЗМ, м. Одеса, Інститут штучного інтелекту та робототехніки ОНПУ, (2020 – 2021 рр.); Міжнародна наукова конференція, Управління проєктами у розвитку суспільства. К.: КНУБА, (2021 – 2022 р.); VIII Міжнародній НТК «Інформатика. Культура. Технології» ІКТ-2021. Одеса, ІКС, ОНПУ; XVII Міжнародній науково-практичній конференції Управління проєктами: стан та перспективи, Миколаїв, НУК ім. Адмірала Макарова, 2021.

**Публікації.**

За результатами дослідження опубліковано 11 наукових праць: 5 наукових статей у фахових виданнях України; 6 тез доповідей на наукових конференціях.

**Структура дисертації.**

Дисертація включає вступ, 4 розділи, висновки та 3 додатки. Обсяг дисертації – 175 сторінок, з них основного тексту – 134 сторінки. Дисертація містить 28 рисунків, 2 таблиці в основному тексті та посилання на 128 використаних джерел.

## **РОЗДІЛ 1**

### **АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ**

#### **1.1 Сучасні моделі, методи, стандарти управління ІТ-проєктами**

##### **1.1.1 Особливості сучасних ІТ-проєктів**

Пошук ефективних механізмів управління системою зазвичай треба розпочинати з виокремлення та характеристики предмету дослідження. В даному випадку мова йде про сутність та характеристики ІТ-проєктів [1]. Аналізуючи доступні до розгляду наукові та професійні джерела з управління ІТ-проєктами було виявлено що найменш дві тенденції [2].

Перша — це випадок, коли ІТ-проєкт розглядається, як комплекс заходів в середині НЕ ІТ-компанії, яка бажає автоматизувати власні бізнес-процеси, зробити їх більш ефективними, розробивши та впровадивши в себе самотужки нові ІТ [3]. Такі ІТ-проєкти можуть бути реалізовані у будь-якій галузі економіки, наприклад в енергетичній [4]. «Практично кожна організація в процесі своєї діяльності стикається з тим чи іншим ІТ-проєктом, причому ІТ-проєкт в організації повинен розглядатися як частина великої системи», як це показано наприклад в [3].

Друга — це застосування для виокремлення особливостей та характеристик ІТ-проєкту критеріїв, що застосовують до звичайних інвестиційних проєктів, наприклад, проєктів будівництва. Це потребує, на нашу думку, уточнити визначення ІТ-проєкту за рахунок його значущих особливостей. Тому далі представлено результати проведеного аналізу.

Існує значна група робіт, в яких визначення ІТ-проєкту сформовано за класичним підходом на основі РМВоК, без жодного врахування особливостей галузі. Так, в [5] ІТ-проєкт — «це тимчасове підприємство, спрямоване на розробку унікального продукту, що має чітко визначений термін виконання,

обмеження за ресурсами, свої критерії якості і поняття про успішне завершення. Далі, авторка зазначає, що «відповідно до цільової спрямованості, ІТ-проект — це проект, в рамки якого входять роботи, пов'язані з інформаційними технологіями, які в свою чергу спрямовані на створення, розвиток і підтримку інформаційних систем». Такі визначення не досить повні, як для виокремлення ІТ-проектів із загального пулу інших проектів, так і для вибору або формування засобів управління, що враховуватимуть ІТ-особливості, задля забезпечення успішного завершення ІТ-проектів.

І навіть доповнення, що подано в [5, 6] у вигляді особливостей ІТ-проектів, не дають чіткого уявлення про їхню сутність - «нестандартний життєвий цикл, який може включати в себе також тестовий, гарантійний та післягарантійний етапи розробки». Тут слід зауважити, що гарантійне або постгарантійне обслуговування не можуть бути етапами розробки. Крім того, в дисертації приймемо, що ЖЦП завершується після передачі його замовнику та введення до експлуатації. За необхідності будь-якого супроводу, за узгодженням сторін, укладається окремий договір, та засоби, що будуть розроблені для управління ІТ-проектом не розповсюджуватимуться на гарантійне та постгарантійне обслуговування. Однією з характеристик ІТ-проекту в [5] зазначають «... необхідність чіткого визначення, вже на етапі ініціації, вимог до ІТ-проектів, незважаючи на рухливість і неоднозначність...», але, в сучасних проектах вимоги можуть формуватися аж до його завершення.

Класичний підручник з управління ІТ-проектами [7] представляє визначення таким чином: «Проект - це скінчений в часі захід, призначений для створення унікальних продуктів або послуг. Унікальність продуктів або послуг проекту обумовлює необхідність послідовного уточнення їхніх характеристик протягом виконання проекту». Аналогічно, у фундаментальному виданні [8] визначення ІТ-проекту подано у досить широкому та загальному сенсі. «Проект — це унікальна, орієнтована на досягнення мети, тимчасова та обмежена умовами дія».

Далі автори подають розширення значень проєкту: « ... це велика або важлива дія (послідовність дій)», як зазначають самі автори, такі визначення є декілька примітивним, та надалі використовують визначення пропоновані інститутами PMI та IEEE.

Мюрей Кантор в [9] не формує у явному вигляді визначення ІТ-проєкту, але він дає значущі характеристики, які будуть корисними у подальшому розгляді предмету дослідження. Отже, М. Кантор характеризує програмні проєкти, як такі, що регулюються переліком вимог, але ці вимоги дуже часто змінюються, або додаються протягом циклу розробки. Тому встановлення стабільних вимог до проєктів розробки програмного забезпечення є проблемою. Крім того, навіть після завершення програмного проєкту вимоги можуть ще виявлятися або вдосконалюватися. Ці зміни зазвичай виявляють під час аналізу проєкту, раннього тестування або несподіваної взаємодії компонентів системи.

Наступною характеристикою, яка на нашу думку пов'язана з попередньою, є критичність комунікацій між замовниками та розробниками ІТ-проєкту. З огляду на складність уточнення вимог, важливо підтримувати із замовником відкриті та конструктивні комунікації.

І ще однією характеристикою ІТ-проєкту автор [9] зазначає постійне коригування припущень, за якими буде виконуватися проєкт. Наприклад, можуть змінюватися компроміси щодо бюджету, графіку та функціонального змісту проєкту.

У [10] «... успішні ІТ-проєкти повинні мати добре позначену основну контрольну точку в здійсненні проєкту, після досягнення якої відбувається відчутний перехід від стану досліджень до стану виробництва». В своїй книзі [10] Уокер Ройс не дає визначення проєкту в явному вигляді. Замість того в тексті присутні визначення: «Проєктування», «Проєктної моделі» та «Проміжної контрольної точки», які у сукупності окреслюють проєктний простір.

До огляду сучасних робіт з управління ІТ-проєктами, для порівняння, вважаємо за необхідне додати аналіз статті, що вийшла у IEEE WESCON, вже більше ніж 50 років тому [11]. Слід зазначити, що вже тоді лунала критика щодо



застосування водоспадної моделі в управлінні ІТ-проєктами. Так, автори [11] виділяли у розробці комп'ютерних програм два основних етапи: аналіз та кодування. Задля ефективного управління ІТ-проєктом в умовах невизначеності автори пропонували ввести кілька «понаднормових» етапів, що включають в себе визначення системних вимог, визначення вимог до програмного забезпечення, розробку програми і тестування. Ці етапи доповнювали б етапи аналізу і кодування. Оцінку ефективності ІТ-проєктів через систему критеріїв подано у [12].

В [13] зазначено, що «.. проєкти розробки програмного забезпечення або ІТ-проєкти залишаються, в більшості випадків, ЗАГАДКАМИ – важко передбачуваними, важко реалізованими і важкокерованими». Тому далі є доцільним розглянути безпосередньо стандарти з досліджуваної предметної області, а саме, згаданий вище стандарт Програмної інженерії SWEBOOK V3 [14], та розширення PMBoK для програмних проєктів [15].

Так, згідно [15], « ... проєкт — це тимчасова робота (endeavor undertaken), що робиться для створення унікального продукту, послуги або результату». Як і всі інші, ПП реалізуються для досягнення певної мети [15]. Крім розробки нових продуктів, вони модифікують існуючі, розширюють їх можливості, інтегрують нововведення до існуючої програмної інфраструктури.

Роботи з програмування можуть позиціонуватися як дії рівня зусиль (level-of-effort, або LOE), наприклад, задоволення запитів на обслуговування, забезпечення операційної підтримки, тощо. Але, якщо вони будуть обмеженими у часі для забезпечення результатів, то їх можна вважати ІТ-проєктами.

Програмні проєкти складні, тому що продукт проєкту нематеріальний та податливий. В стандарті використано термін «malleable» [15], у якості доповнення до перекладу можна сказати, що продукт програмного проєкту по-перше володіє властивістю легкої зміни, та по-друге, може бути зміненим розробниками. Продукт проєкту розробки програмного забезпечення є прямим продуктом когнітивних процесів людини [15], та створюється на основі інтелектуального

капіталу команди проєкту. Зазвичай, створення таких продуктів проєктів вимагає інноваційних рішень.

Комунікації всередині команди проєкту розробки програмного забезпечення та комунікації зі стейкхолдерами характеризуються як такі, в яких «.. не вистачає ясності та однозначності», це характеризує проєкт як систему, зі складністю взаємодії з її частинами.

Таким чином, програмні проєкти збільшують причини появи ризиків та невизначеності за рахунок інновацій, нематеріальності продукту, неузгодженості потреб стейкхолдерів та чіткого бачення продукту [2].

Крім збільшення ризиків, ці причини ускладнюють планування та попередню оцінку програмних проєктів. Складним є й формування кількісної оцінки та метрик якості програмного забезпечення, вимірювання поточного стану продукту (метод освоєного обсягу тут не працює) [2]. Ще однією причиною складності проєктів розробки програмного забезпечення є безперервне оновлення та розвиток процесів, методів та інструментів, які повинні постійно опановувати члени команди, іноді і впродовж ЖЦП.

Аналіз сучасного стану ринку ІТ-проєктів виявив новий акцент проєктної діяльності в ІТ-галузі. Це так звані ІТ-сервіси.

Компанії, для досягнення своїх бізнес-цілей у більшості випадків застосовують ІТ у вигляді якісних ІТ-послуг, які відповідають цілям бізнесу, вимогам і очікуванням замовника, при цьому останнім часом все більша увага приділяється не розробці ІТ-рішень (наприклад, бізнес-додатків), а управління послугами з їх супроводу, що гарантує високу доступність рішення для кінцевих користувачів [16]. При цьому в життєвому циклі ІТ-рішень на їх експлуатацію доводиться 70-80% часу і фінансових коштів і лише 20-30% часу і коштів витрачається на розробку (придбання) та впровадження продукту.

Відзначимо, що сьогодні керівники багатьох компаній незадоволені якістю ІТ-послуг, що надаються їхніми власними ІТ-підрозділами. Для цього є багато причин. Дуже часто ІТ-проєкти не завершуються у запланований термін та бюджет [16].

Вирішення зазначених питань потребує структурованого підходу до управління ІТ-послугами, що дозволяє зробити роботу ІТ-підрозділу ефективною і раціональною. Такий підхід носить назву ITSM, а його основним принципом є розгляд ІТ-служби як підрозділу, постійно орієнтованого на потреби своїх користувачів. Зазначений принцип організації діяльності бажано застосовувати для компаній будь-якого масштабу і не залежить від того, чи входить ІТ-служба до складу компанії, або є зовнішнім постачальником ІТ-послуг.

Далі представлено аналіз визначень ІТ-проектів за різними джерелами. Загальний перелік визначень ІТ-проекту не матиме сенсу, його необхідно згрупувати за певними критеріями, до окремих груп яких можна буде застосовувати схожі засоби управління.

Так автор [17] у якості критерія для виділення типів проектів використовує кількісний склад команди. Загалом, за цим критерієм представлено три типи проектів.

1. «Супермен-одинак» [17]. Проект простий, бере участь одна людина, яка робить все сама, від проведення ринкових досліджень до написання коду. Використовує власні джерела фінансування.

2. Невелика команда, що працює за контрактом. ІТ-компанія складається з п'яти – десяти програмістів і керівника, найнята замовником для створення ІТ-продукту. Працюють за контрактом, по завершенні якого усі зв'язки обриваються.

3. Численна команда штатних співробітників, кількістю зі ста чоловік, працюють в корпорації по найму. ІТ-продуктом може бути: призначений для продажу (так званий коробковий) продукт; корпоративна система управління; система автоматизації бізнесу, тощо.

Ці різновиди представлені з метою виокремлення важливих відмінностей в підходах до управління ІТ-проектом.

Проаналізувавши теоретичні та професійні джерела та стандарти, була виявлена необхідність обмежити область подальшого пошуку з величезного розмаїття визначень та тлумачень ІТ-проекту [18].

Зв'яжемо ІТ-проект з професійним програмуванням. Надалі будемо розглядати сегмент чи сукупність ІТ-проектів, як діяльність, що направлена на отримання доходів за допомогою програмування. Тут доцільно провести аналогію з проектно-орієнтованою організацією, яка майже увесь свій дохід отримує за рахунок реалізації проектів [18].

Значущою характеристикою проектів, що будуть розглядатися, є наявність обов'язкової ролі споживача, який готовий платити за розробку програмного продукту. Тоді ІТ-проект - це завжди командна робота, в якій присутні дві ролі: «програміст» і «споживач».

«Програміст» з використанням інтелектуального капіталу та когнітивних процесів створює програмний продукт, при цьому за необхідністю використовує інноваційні рішення, та передає його «споживачеві».

### 1.1.2 Аналіз особливостей управління ІТ-проектами

Наступним кроком аналізу проблеми стало пошук та дослідження засобів управління ІТ-проектами, що мають особливості, які було виокремлено в підрозділі 1.1.1.

Саме для таких проектів застосовують «правило хреста» [19]: якісно, швидко, дешево, готово. Це правило є адаптацією чи розвитком відомого проектного трикутника: якість (обсяг робіт), час, вартість [20] (рис. 1.1).

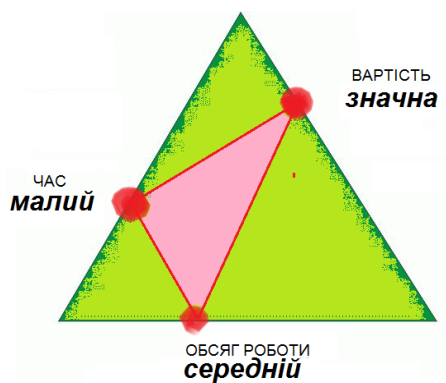


Рис. 1.1. Проектний трикутник: Рис. 1.2. Визначення якості робіт у

**обсяг робіт, час, вартість****проектному трикутнику**

На перетині прямих, які відображають кількість кожного із зазначених параметрів, формують фігуру, площа якої може бути інтерпретована, як якість робіт, що виконується за обраними обмеженнями (рис.1.2). Можна вважати, що «правило хреста» модернізувало поняття «якості проекту» із загальноуправлінської площини в поняття «готовності ІТ-продукту», тобто працездатності програмного забезпечення.

«Правило хреста» можна інтерпретувати наступним чином: «Обирай будь-яких три пункти, але четвертий буде не під силу» [19]. Проект може бути одночасно якісним, дешевим, швидко виконуватися, але його ніколи не буде завершено. Або, навпаки, проект буде завершений швидко та дешево, але якісним зробити його не вийде.

«Правило хреста» може бути прийнятим як компоненти загальної системи управління ІТ-проектами з точки зору можливості їх спостерігати та керувати ними. Тобто компонентам системи управління треба задати наступні властивості: деякі компоненти можуть бути виявлені (бо вони мають суттєвий вплив на успішне завершення ІТ-проекту), деякі з них ми можемо спостерігати (у загальному випадку не всі з виявлених), деякими з них ми можемо керувати. Визначення цих компонентів та їх властивостей завершується формуванням стратегії управління кожним окремо взятим проектом. Тому далі в дослідженні було проаналізована сутність управління ІТ-проектами.

З огляду на це, по перше, слід визначитися з терміном «Управління ІТ-проектами», тому що зазначена вище особливість ІТ-проектів позначається і на тлумаченні управління ними.

Але по-перше, визначимо відмінності між декількома термінами: 1) Управління ІТ-проектами, 2) Технологія створення програмного продукту (ТСПП), 3) Розробка (створення) ПЗ.

Одразу зауважимо, що в літературних джерелах зустрічається ототожнення усіх трьох термінів. Деякі автори використовують їх, як синоніми, або використовують один з них, наприклад «створення ПЗ», але при цьому, мають на

увазі повний набір процесів, від ініціації до завершення, та передачі продукту проєкту замовнику.

Ще зауважимо на те, що означені терміни мають різні строки появи загально прийнятого вживання. Першими стали застосовувати терміни «Розробка/створенням ПЗ», пізніше з'явилися ТСПП та «Управління ІТ-проєктами». Але, поява терміну не визначає особливостей управління, тому надалі в дисертаційному дослідженні не будемо приділяти цьому увагу.

Натомість приймемо наступне: будемо вважати у рамках представленого дослідження що, розробка або реалізація ПЗ є одним із складових етапів ТСПП, а ТСПП є складовою процесу управління проєктом створення програмного продукту, або управління ІТ-проєктами. Але в поданому нижче аналізі літературних джерел, авторські терміни залишилися без змін.

Найбільш вагомим, з точки зору компетентності визначень стосовно управління саме ІТ-проєктами, вважаємо «Розширення до РМВoК при створенні програмного забезпечення» (Software Extension to the PMBoK) [15].

Там зазначено [15], що «... управління проєктами — це застосування знань, навичок, інструментів та прийомів для проєктної діяльності для задоволення вимог проєкту». «Управління проєктами здійснюється шляхом використання логічно згрупованих процесів управління проєктами, що складаються у групи: ініціації, планування, виконання, моніторингу та контролю, і закриття» [15]. Згадане розширення [15] згідно специфіки ІТ-проєктів формує особливості управління які загалом сформовані через обмеження. Вони звичайно формуються ТСПП, і в [15] згруповані через технологічні фактори, які ці обмеження продукують.

До технологічних факторів, які можуть накладати обмеження на програмні проєкти та програмні продукти, автори [15] відносять: стан апаратних та програмних технологій; апаратні платформи, програмні платформи, операційні системи та протоколи зв'язку; цілісність, обмеження та протоколи архітектури ІТ та інші.

До нетехнологічних факторів, що можуть накладати обмеження на проекти програмного забезпечення, автори [15] відносять: відповідність безпеці, надійності, доступності, масштабованості, продуктивності, тестування, забезпечення інформації, локалізації, ремонтпридатності, підтримки, правил, політики клієнтів, підтримки інфраструктури, доступність та вміння членів команди, середовище та методи розробки програмного забезпечення, а також організаційна зрілість та можливості.

Ці фактори та обмеження формують специфіку та особливості ІТ-проектів, що притаманні лише їм, та вимагають спеціальних засобів та підходів в управлінні ІТ-проектами.

Окремо автори [15] роблять наголос на складність управління ІТ-проектами і формують перелік факторів, що ускладнюють управління ІТ-проектами. В рамках дослідження були виокремлені ті з них, наявність яких вимагатиме від команди проекту додаткових зусиль та, відповідно, додаткових заходів з управління проектом. Серед них:

- нематеріальність та податливість (гнучкість) продукту проекту;
- розробка ПЗ часто характеризується як процес навчання команди проекту, в процесі якого здобуваються знання;
- нелінійне масштабування ресурсів ІТ-проекту, виміру проекту і продукту, вхідна невизначеність в проекті і в змісті його продукту;
- вимоги до ПЗ часто змінюються в процесі його розробки, під час накопичення знань та масштабування проекту і продукту;
- інтелектуальний капітал команди ІТ-проекту, яка займається розробкою ПЗ, є основним капіталом для програмних проектів та ІТ-компаній, що займаються розробкою ПЗ, оскільки ПЗ є прямим продуктом когнітивних процесів людини;
- в комунікаціях та координації всередині команд розробників ПЗ і з зацікавленими сторонами проекту часто не вистачає ясності;

- багато інструментів та методів, що застосовують в розробці ПЗ, призначені для організації та управління проектною комунікацією та координацією, які не є складовими ТСПП;
- створення ПЗ вимагає інноваційних підходів при вирішенні проблем. Більшість ПП надають унікальні продукти і вони більше схожі на проекти досліджень і новітніх розробок, ніж на будівельні або виробничі проекти;
- зазначені вище інновації, нематеріальність продукту, стейкхолдери, які не можуть чітко сформулювати або узгодити вимоги, які повинні бути задоволені за допомогою програмного продукту - все це формує додаткові ризики та невизначеності, які у свою чергу напряду впливають на команду ІТ-проекту;
- первісне планування і оцінка ПП є складним завданням, тому що ці дії залежать від вимог, які часто є неточними або є частиною історичних даних, які часто відсутні або непридатні;
- точне оцінювання проектних показників також є складним завданням, оскільки ефективність і результативність розробників ПЗ дуже різняться;
- об'єктивна кількісна оцінка і вимір якості ПЗ складні через нематеріальну природу ПЗ;
- розробники ПЗ використовують процеси, методи і інструменти, які постійно розвиваються і часто оновлюються;
- технології, платформи, ПЗ інфраструктури часто змінюються або оновлюються, що може вимагати внесення змін до ПЗ, що розробляється.

Таким чином, можна зазначити, що нематеріальна природа продукту ІТ-проекту створює додаткові проблеми в управлінні, при вимірюванні поточного стану продукту, що, в свою чергу, ускладнює моніторинг і контроль впродовж ЖЦП. Традиційні підходи управління проектами, такі, як структурна декомпозиція робіт, календарні розклади, сітьові діаграми, звіти про прибутки та збитки, погано адаптуються до потреб ПП, в яких, впродовж ЖЦП необхідна постійна демонстрація частково завершеного продукту проекту. ПЗ є прямим



продуктом когнітивних процесів людей, залучених в інтелектуальну, інноваційну командну роботу.

Це робить командну роботу в ІТ-проекті за своїм характером відмінною від командної роботи в інших інженерних проектах. Багато процедур і методів управління проектами розробки ПЗ призначені для полегшення спілкування і координації між членами команди, для забезпечення скоординованої та інтелектуально-інтенсивної роботи.

Зазначимо ще одну особливість ІТ-проектів, яка вимагає особливого управління. Продуктивність робіт ПЗ включає як кількісні, так і якісні показники. Але кількість написаного ПЗ (кількість рядків коду) не є достатнім показником продуктивності програміста. Програміст, який пише невелику ефективну (або, можливо, інноваційну) програму, більш продуктивний для досягнення успішного результату, ніж програміст, який пише велику неефективну (або непрацюючу) програму. Також, програміст, який поспішає виконати завдання, але робить багато помилок, які потребують виправлення, менш продуктивний, ніж програміст, який діє повільніше, але створює програму з меншою кількістю дефектів [15]. В [8] розробку ПЗ позиціонують як регламентований, системний підхід до розробки, оперування, обслуговування та припинення експлуатації ПЗ за допомогою практичного застосування наукових знань і процесів. Тут мова йде про повний життєвий цикл ПЗ від розробки до припинення експлуатації.

В [17] під діяльністю з управління проектами, Скотт Беркун, якій у свій час приймав участь у розробці Windows, MSN, Internet Explorer, працював у Microsoft, а зараз є консультантом та тренером з управління ІТ-проектами, розуміє « ... роботу з управління командою при уточненні деталей проекту (загальне планування, складання календарного плану, вироблення вимог), проведення проекту через етапи проектування та розробки (ведення переговорів, прийняття рішень, вироблення стратегії), доведення проекту до завершення (лідерство, вирішення критичних ситуацій)».

Звернемося ще раз до джерела [11], де зазначено, що вже на той час (1970 р.) класичний, чи « .. основний підхід в управлінні проектами, що описується

водоспадної моделлю, є досить ризикованим і допускає невдале завершення». Така фаза ІТ-проєкту, як тестування знаходиться в кінці циклу розробки, і то є перший момент у часі, коли можна оцінити реальну тривалість «.. виконання, обсяг займаної пам'яті, швидкість введення/виведення і т.і., щоб порівняти їх із значеннями, встановленими при аналізі. Зміни, внесені в програму, можуть виявитися настільки руйнівними, що вимоги до ПЗ, на яких ґрунтувалася розробка програми, виявляться невиконаними. В такому випадку доведеться або переглянути вимоги, або внести істотні зміни в структуру програми». Тобто, ще у 1970 році автори [11] звертали увагу на неефективність та ризикованість застосування водоспадної моделі в управлінні ІТ-проєктами.

Фахівці відзначають, що програмування - це і не мистецтво і не наука, що це є ремесло, майстерність до якого приходять лише із досвідом. А досвід програміста - це постійне самовдосконалення та саморозвиток. І саме тут слід обрати моделі та засоби управління командою проєкту [18].

Модель процесу створення ІТ-продукту – це, по-перше, формалізація процесу розробки [18]. Керівництво SWEBOOK V3 має декілька областей знань для вдалої розробки ПЗ, та окремо визнає управління ІТ-проєктами задля досягнення зовнішніх, щодо розробки, цілей [21]. Між тим, головною причиною провалів більшості ІТ-проєктів є саме застосування неадекватних методів управління розробкою [11], бо те, що створюють програмісти, є нематеріальним продуктом, а результатом колективної ментальної роботи, у вигляді програмного коду.

Можливо, розробляти ПЗ не складніше, ніж створювати літаки, але досвід проєктів матеріального виробництва, відображений у стандарті PMBoK [19], не сприяє успіху в управлінні ПП [18]. Управляти ІТ-проєктами треба інакше, з урахуванням творчості, як інтелектуальної діяльності членів команди. Але, нажаль, аналітичний вираз «творчості» дотепер відсутній. При тому відомо, що як і в програмуванні, так і в творчості, рушійну силу створює натхнення, і чим складніший проєкт, тим важче «управляти» натхненням [23]. Повертаючись у площину об'єктивної реальності, зазначимо, що члени команди ІТ-проєкту повинні застосовувати як абстрактне так і системне мислення для моделювання у

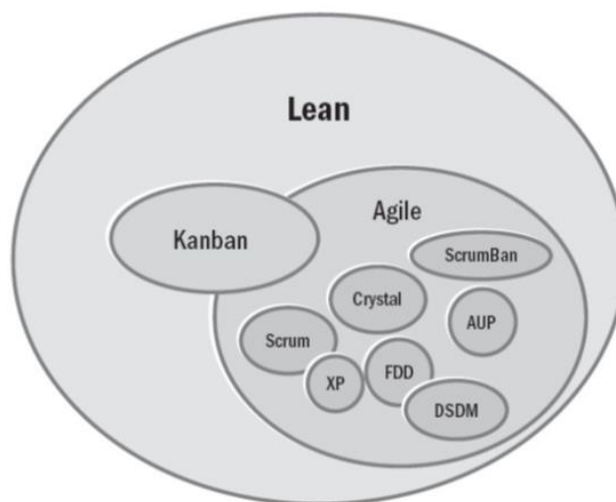
власній уяві абстракції майбутнього продукту проєкту, зв'язків між членами команди, замовниками, стейкхолдерами, тощо.

Додатковою складністю ІТ-проєктів є зміна у часі їх структури та зв'язків між зазначеними абстракціями. І усім цим необхідно управляти для забезпечення успішного завершення ІТ-проєкту. Класичні засоби управління проєктами в таких умовах або мало ефективні, або ж неефективні цілком. Загальновідоме співвідношення розпочатих та успішно завершених проєктів. Зазвичай, воно знаходиться в інтервалі 10%-30%. І, як було сказано вище, причиною провалів є неадекватні засоби управління. Для ІТ-галузі виділяють чотири варіації управління за критеріями жорсткості та точності [24]: 1) як вийде, 2) жорсткий водоспад, 3) гнучке управління, 4) метод частих ітерацій. Для зазначеної вище групи проєктів, в яких є задіяним інтелектуальний капітал команди для формування інноваційних проєктних рішень, перша варіація — «як вийде» буде не бажаною, друга взагалі не працюватиме, тому розглянемо засоби [24], що забезпечать гнучке управління або метод частих ітерацій.

Враховуючи це, PMI у 2017 створив Agile practice guide як додаток до шостого PMBoK [25]. Автори [25] відзначають, що гнучкі підходи і гнучкі методи «... це загальні терміни, що охоплюють безліч структур і методів».

**Agile** - це загальний термін, що стосується будь-якого підходу, техніки, структури, методу або практики, які відповідають цінностям і принципам Agile Manifesto та принципам ощадливого виробництва. Схематично це показано на рис. 1.3.

Під бережливим виробництвом автори [25] мають на увазі: «акцент на цінності», «невеликі партії» та «усунення відходів». Agile — це і підхід, і метод, і практика, і техніка, і фреймворк. Може застосовуватися будь-який термін в залежності від конкретної ситуації та конкретного проєкту [25].



**Рис. 1.3. Agile як загальний термін для багатьох підходів [25]**

Розглянемо та проаналізуємо коротко деякі з представлених компонентів за критерієм обраного об'єкту дослідження.

**Crystal** – сімейство методологій розробки ПЗ із загальним генетичним кодом, що включає: часту доставку, особисті комунікації та вдосконалення через рефлексію [26].

Автором Crystal є Алістер Кокберн, один із співавторів маніфесту гнучкого програмування [26]. Для кожного типу проєкту Crystal він пропонував обирати власну методику, що базуватиметься на ключових правилах розробки:

- «людино-орієнтовані» методології приносять більше користі, ніж «процесно-орієнтовані»;
- методологія повинна бути підібрана індивідуально під проєкт і команду — немає універсальної методики розробки додатків.

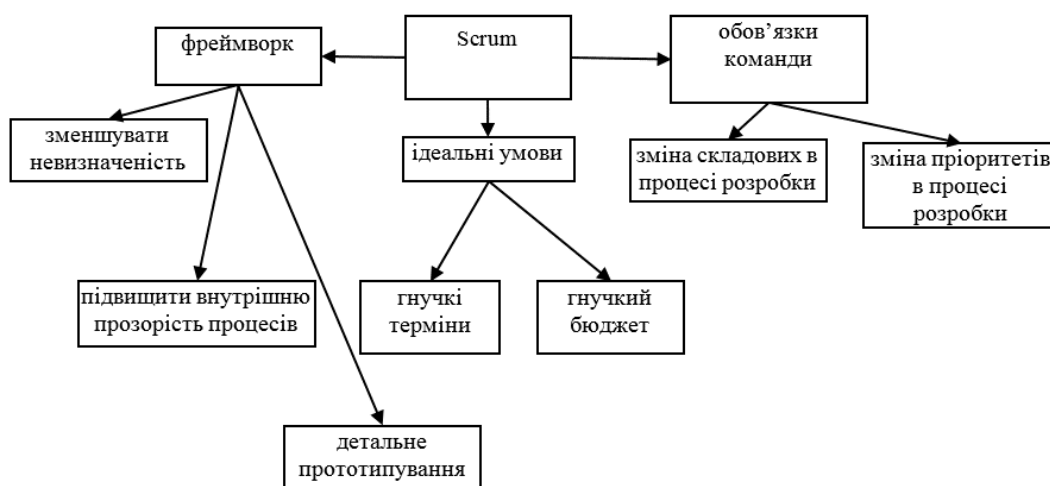
Crystal класифікована за критерієм кількості людей в команді проєкту: Clear – від 2 до 8 осіб, які сидять разом в одному або суміжних офісах; Yellow – 10-20 осіб; Orange – 20-50 осіб; Red – від 50 до 100 осіб. Для масштабних проєктів використовують додаткові кольори: Maroon, Blue і Violet.

**Scrum** - це гнучка методологія Agile, філософія розробки ПЗ, сукупність методів розробки і управління, гармонічно поєднаних між собою, в основі яких лежать принципи виробничої ітеративності, що в свою чергу, дозволяють замовнику частіше отримувати певний робочий функціонал в рамках розробки

проєкту [27]. Відповідно до методології Scrum [28], всі члени команди проєкту повинні бути взаємно замінні і рівнозначні. Якщо команді не достатньо знань для виконання проєкту, то якість проєкту може бути під загрозою. Однак, метод характеризується відсутністю жорстко заданого розподілу ролей та обов'язків в команді. Принцип «всі відповідальні за все» працює далеко не завжди. Крім того, в Scrum добре реалізований і сам процес збору функції та її розподіл по ітераціях. Як і багато інших гнучких методологій, Scrum вимагає наявності представника замовника, або представника ринку, але замовник не завжди може визначити, що саме йому потрібно, а часта зміна вимог лише уповільнює роботу [29].

Підсумовуючи, можна сказати, що Scrum – це набір принципів, на яких будується процес розробки ПЗ, який дозволяє в короткий проміжок часу (спринт), надавати замовнику працююче ПЗ з доданими можливостями, для яких був визначений найбільший пріоритет. Потрібний функціонал для реалізації в черговому спринті визначається до його початку на етапі планування і не може змінюватися на протязі всього спринта. При цьому жорстко фіксований невеликий термін спринта додає процесу розробки гнучкості [30]. Так, у [31] подана графоаналітична модель ситуаційного управління ІТ-проєктами, які застосовують для розробки технологію Scrum.

Особливості Scrum згруповані на рис. 1.4 [32].



**Рис. 1.4. Основні особливості методології Scrum**

Scrum має і недоліки:

- простота та мінімалізм ведуть до більш жорстких вимог, що створює «конфлікт» з головним принципом клієнтоорієнтованості;
- самоорганізація та багатофункціональність команди в Scrum веде до підвищення витрат на відбір персоналу, мотивацію, навчання.

Методологія **Kanban** запроваджує систему «ощадливого виробництва», та провадить ідею створення єдиного потоку (процесу) без простоїв від незавершених або неузгоджених завдань. Це дозволило уникнути незапланованих витрат, покращити якість продукту, знизити вартість та скоротити терміни виконання [33]. Kanban добре працює на стартапах, тобто там, де не має чіткого плану, але активно працюють над розробкою [32].

Kanban-дошка – обов’язковий елемент гнучкої технології для відстеження оперативної ситуації всередині ІТ-компанії, яка застосовує ітеративну модель. Кожен член команди отримує доступ до дошки у будь-який час і бачить, на якому етапі перебуває завдання.

План робіт кожного проєкту аналізують та поділяють дошку на стовпці-етапи: To do – показує кількість завдань в необхідних для виконання; In progress – завдання, які вирішуються в даний момент; Ready for deploy – завдання, які вже виконані, але не представлені в тестовому оточенні; QA – завдання в процесі тестування; PM approving – готові завдання проходять перевірку проєктним менеджером; Done – виконані (завершені) завдання поточного спринту.

Формалізація процесів в спринті відбуваються наступним чином: взявшись до роботи, програміст перетягує завдання з колонки «To do» в «In progress». В цей момент він має лише одну задачу, концентрується на її ефективному вирішенні, так як паралельне виконання завдань не допускається. Після закінчення цієї роботи, він перетягує її в колонку «Ready for deploy» [30].

Особливості Kanban подано на рис. 1.5.

Kanban має і свої недоліки:

- команди повинні бути чисельністю не більше 5 осіб;
- не призначений для довгострокового планування.



**Рис.1.5. Основні особливості методології Kanban**

Зазвичай, Kanban використовують у стартапах, які не мають чіткого плану, але активно працюють над розробкою.

Методологія **Lean** використовує концепції зменшення витрат та ощадливого виробництва, допомагає знаходити проблеми в процесі з подальшим вирішенням за допомогою різних практик та інструментів з постійним вдосконалення на всіх етапах реалізації [32]. Основні принципи Lean корелюють із завданнями ІТ-проєкту [34]:

- визначити цінність конкретного продукту;
- визначити потік створення цінності для цього продукту;
- забезпечити безперервне (протягом всього потоку) створення цінності продукту;
- дати змогу споживачеві тестувати продукт;
- прагнути досконалості.

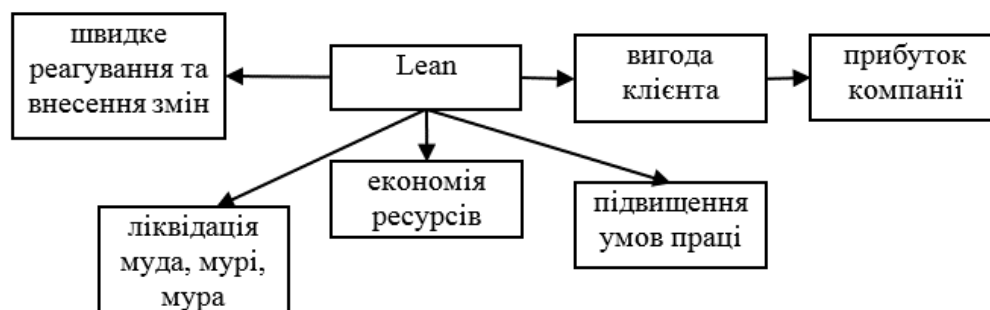
Методологія передбачає за японською традицією, постійне вдосконалення процесу через призму недопущення непотрібних витрат, виправлення попередніх помилок в управлінні (муда 無駄); нерівномірності та невідповідності навантаження (мура 無理); недоцільності, складності в роботі команди (мурі 斑).

На рис. 1.6 надана схема основних особливостей методології Lean.

Недоліками Lean є [34]:

- необхідний постійний контроль управління з метою виявлення потенційних проблем та підтримки ефективності проєкту;

- на відміну від Scrum, не пропонує чіткого робочого процесу для реалізації частин проєкту, що веде до збільшення строку виконання проєкту;
- модифікація може привести до нерівномірності навантаження на команду, що в свою чергу веде до невдоволення співробітників.



**Рис. 1.6. Основні особливості методології Lean**

**RUP**, Раціональний уніфікований процес — це одна з спіральних методологій розробки ПЗ, в якій для моделювання використовується UML. RUP передбачає поділ проєкту на кілька підпроєктів, які виконуються послідовно у вигляді ітерацій. Кожна з них визначена набором цілей, які повинні бути досягнуті в кінці ітерації. Цілі кінцевої ітерації мають цілком збігатися з набором цілей, зазначених замовником/споживачем продукту ІТ-проєкту, тобто усі вимоги мають бути виконані [23, 24].

**MSF** (Microsoft Solutions Framework) є однією з інтерпретацій спіральної моделі розробки додатків у вигляді набору концепцій і рекомендованих моделей, на основі технологій і інструментальних засобів Microsoft. MSF базується на організації розподілених обчислень і застосування технологій «клієнт-сервер» компанії Microsoft, її партнерів і замовників [23, 24].

**XP** (Extreme Programming), або екстремальне програмування має ітеративний фактор прискорення розробки: розробка ведеться короткими ітераціями при наявності активної взаємодії із замовником. Використовується індивідуальний процес розробки, що спрямований на покращення власної продуктивності членів команди ІТ-проєкту. Використовує також принципи зрілості процесів до практики одного розробника.



Всі гнучкі технології розробки ПЗ ґрунтуються на ітеративності, інкрементності та самокерованості команди ІТ-проєкту [23, 24].

В якості висновків до підрозділу відзначимо, що траєкторія успіху в управлінні ІТ-проєктами з творчою та інноваційною специфікою проходить через область психології управління людьми, з використанням гнучких технологій управління. В дисертаційній роботі «інноваційну специфіку» пропонуємо розуміти наступним чином. Усі проєкти, що розглядаються в роботі, мають у собі ознаки інновації. Це можуть бути інноваційні проєкти: продукт проєкту дотепер не створювався, технологія розробки такого продукту відсутня, команда проєкту подібний продукт дотепер не створювала. Або може бути проєкт із часткою інновації, тобто присутні не всі вище перелічені ознаки, а лише частина з них. В подальшому дисертаційному дослідженні ставиться задача розробити моделі, методи та засоби управління лише для інноваційних ІТ-проєктів, або до ІТ-проєктів, що мають інноваційну складову.

## **1.2. Особливості управління командами ІТ-проєктів**

ІТ-проєкти, безумовно, характерні для компаній з різноманітними організаційними структурами управління, кількістю працюючих, потенційного масштабу проєкту та профілем діяльності [32], але всі вони мають деякі загальні риси, що були проаналізовані у підрозділі 1.1.

ІТ-проєкти найбільш ризикові, динамічні проєкти, що характеризуються значною кількістю проблем, високим рівнем напруженості та конфліктів [35], часто вимагають рішення нетипових завдань, і за статистикою, лише третина з них завершується цілком успішно [36]. Успіх впровадження та управління ІТ-проєктами напряму залежить від команди виконавців, їх інформативності, розподілу ролей, повноважень, відповідальності.

Процес створення команди проєкту ускладнюється кваліфікацією та компетентностями спеціалістів, участь яких протягом проєкту буде необхідна. Створення команди ІТ-проєкту може ускладнюватися і тем фактором, що є

випадки, коли команда не має прямого контролю над всіма спеціалістами, в роботі яких є потреба. Ще одним з факторів, що впливає на успішність виконання ІТ-проектів, є відсутність ідеальної системи управління проектами, яка б на 100 % була прийнятною і керівникам і всім членам команди [37]. Тому, питання формування команди є досить актуальним задля забезпечення завершення проекту в межах терміну і бюджету, та задоволення очікувань замовника.

Теоретичні основи командної роботи висвітлені в роботах М. Амстронга, Джона С. Максвелла, Річарда Л. Дафта, дослідженнями в області управління командами проекту займалися такі вітчизняні фахівці, як Бушуєв С.Д., Колеснікова К.В., Морозов В.В., Прокопенко Т.О., Бушуєва Н.С., Бедрій С.Д., Бушуєв Д.А., Кузьмінська Ю.М., Рач В.А.

Значний вклад в розвиток моделей, методів та засобів управління ІТ-проектами внесли Чумаченко І.В., Кононенко І.В., Цюцюра С.В., Данченко О.Б., Тесленко П.О.

Креативними технологіями управління проектами займалися Бушуєв С.Д., Бушуєва Н.С., Бабаєв І.А., Яковенко В.Б., Гриша Е.В., Дзюба С.В., Войтенко А.С., Кузьмінська Ю.М., Вереніч О.В., Данченко О.Б., Польшаков І.В.

Розглянемо визначення команди проекту, за різними джерелами. Так РМВоК [22], визначає команду проекту як «групу осіб, яка підтримує керівника проекту у виконанні проекту для досягнення цілей проекту». Там же, окремо виділяють «команду управління проектом (Project Management Team). Це члени команди проекту безпосередньо зайняті в операціях з управління проектом». В наступній версії РМВоК, сьомій [38], команда проекту інтерпретується, як «набір осіб, які виконують роботу над проектом для досягнення його цілей».

Команда створюється на період реалізації проекту та поєднання її членів повинно формувати синергічний ефект. Особливості роботи в команді допускають, що кожен окремо взятий учасник команди не обов'язково повинен мати повний набір навичок та досвіду для виконання проекту. В індивідуальних особливостях виконавців є причина створення команд, де сильні сторони кожного співробітника доповнюють слабкі сторони колег. Обмеженість ресурсів може

компенсуватися за рахунок професіоналізму та ефективності сумісних дій усіх членів [39].

Життєвий цикл команди, за класичним підходом (PMBok) [22] послідовно проходить декілька етапів, які змінюють один одного:

- формування команди (постановка цілей, розподілення ролей в команді);
- притирання (осмислення цілей, визначення спільного вектору руху);
- нормалізація (досягнення цілей внаслідок компромісів та налагодження комунікацій);
- функціонування (збільшення продуктивності за рахунок оптимізації процесів розробки і самоорганізації членів команди);
- розформування команди (отримання результатів, завершення проєкту).

Однак, спочатку слід визначитися з методологією розробки продукту проєкту, а вже потім обирати варіанти створення команди ІТ-проєкту.

В управлінні командами проєктів можуть бути різні підходи до цього процесу: жорсткі (орієнтовані за задачу), гнучкі (орієнтовані на команду) та незалежні команди з високим ступенем незалежності [40].

Методи, за якими працюють команди ІТ-проєктів, різнопланові, серед них виділяють: Scrum, Kanban, XP, Lean та інші. В дисертаційному дослідженні проблеми вибору методології розробки та управління ІТ-проєктом не розглядається, але тип проєктів, який було виокремлено у підрозділі 1.1, містить проєкти із значною долею інновацій та передбачає застосування саме гнучких технологій розробки та управління. Тому далі акцент буде зроблено саме на такі проєкти і такі команди [41].

Класичний Scrum містить три ролі в команді, що відображено на рис. 1.7.

Scrum Майстер, найважливіша роль (менеджер проєкту) в методології, є з'єднуючою ланкою між замовником та командою. Основні обов'язки пов'язані із створенням робочої атмосфери довіри, ліквідації перепонів, дотримання практик і процесу в команді.

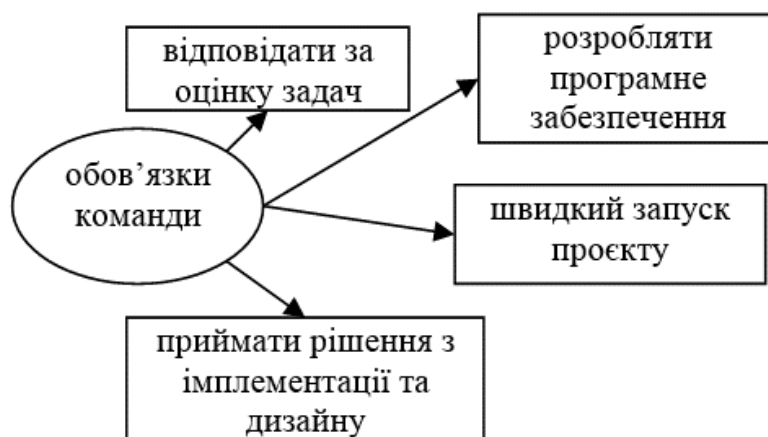
Замовник (власник продукту) відповідає за розробку продукту. Як правило, це менеджер продукту або представник замовника, якщо це аутсорс продукт.

Замовник приймає кінцеві рішення для команди в проєкті, відповідає за формування і бачення продукту, керує рентабельністю та очікуваннями замовників та всіх стейкхолдерів, координує і розставляє за пріоритетами потік задач, взаємодіє з командою і замовниками.



**Рис. 1.7. Склад команди в методології Scrum [42]**

В методології Scrum команда є самоорганізованою та самокерованою, бере на себе обов'язки виконання задач перед Замовником. Основні обов'язки команди відображені на рис. 1.8.



**Рис. 1.8. Схема основних обов'язків команди [42]**

Kanban працює з командами підтримки [33], серед яких:

- групи підтримки програмного забезпечення, де важлива швидкість реагування на зміни;
- групи тестування, що працюють окремо від груп розробки;
- служби підтримки;
- інші групи «неосновних виробництв».

Порівняння методологій Scrum та Kanban з точки зору вимог до команди проєкту подано у табл. 1.1.

Далі представимо результати аналізу джерел, які рекомендують підходи та засоби управління командою ІТ-проєкту, які використовують гнучкі технології. Так, в [43] автор, описуючи алгоритм управління командою ІТ-проєкту, ще у 2002 році, зробив наголос на необхідність використовувати та розвивати творчу, креативну та самоорганізаційну складові, задля підвищення імовірності успішного завершення ІТ-проєкту. «Для управління командою потрібно піти далі інженерії та управління ІТ-проєктом — в сферу відносин між людьми. Це важливий аспект проєкту, і його ігнорування може закінчитися серйозними проблемами для проєкту і його керівника» [43], тому автор пропонує наступний підхід, щодо управління командою ІТ-проєкту.

*Таблиця 1.1.*

**Порівняльна характеристика методологій Scrum та Kanban**

Характеристика	Kanban	Scrum
Темп	Повторювані спринти фіксованої тривалості	Безперервний процес
Випуск релізу	В кінці кожного спринту після схвалення проєктним менеджером (власником товару)	Потік триває без перерв або на розсуд команди
Ролі	Власник продукту, Scrum-майстер, команда розробників	Команда під керівництвом проєктного менеджера; в деяких випадках залучаються тренери по Agile / Kanban
Головні показники	Швидкість команди	Провідний час
Прийнятність змін	У ході спринту зміни небажані, так як можуть привести до невірної оцінки задач	Зміни можуть трапитися в будь-який момент

Визначаючи структуру команди, менеджер проєкту враховує особисті і професійні потреби членів команди, беручи до уваги наступні людські фактори членів команди, такі як: кваліфікацію, підготовку і досвід; особисті прагнення і кар'єрне зростання; потреби в наставництві та розвитку. Така команда, що кілька місяців буде працювати разом заради досягнення спільної мети, становитиме єдине ціле, з повним взаєморозумінням між її членами. Спілкування в таких командах повинно бути двох типів: 1) пов'язане з проєктом і 2) що знімає напругу.

Інформованість членів команди про прогрес проєкту та його проблеми здійснюється через їх доступ до звітів про стан проєкту та до відгуків стейкхолдерів через: дошки оголошень, поштову розсилку, веб-сайт, наради та презентації найкращих практик. Другий тип спілкування рекомендовано здійснювати через: свята для членів команди, святкування днів народжень, цікаві вікторини, неформальні вечери, тощо. Крім того, до задач управління командою ІТ-проєкту автори [43] відносять удосконалення професійних навичок молодих членів команди, і це завдання всієї команди.

Автори [8], окрім технічних компетентностей членів команди розробки ПЗ при формуванні та управлінні командою ІТ-проєкту звертають особливу увагу на необхідність враховувати: індивідуальні особливості розробників та загальний потенціал команди. А процес проєктної розробки автори трактують виключно як «спільну діяльність».

В якості інструментів відбору до команди проєкту автори [8] застосовують: індикатор типу особистості Майерса-Бріггса; модель FIRO-B, що визначає та аналізує фундаментальні міжособистісні відносини; модель сортування темпераментів Кірсі; модель міжпроцесної взаємодії Келера.

В стандарті [38] вказано, що проєктні команди працюють у власному організаційному і професійному культурному просторі, і те, що середовище спільної роботи проєктної групи повинно сприяти: узгодженості з іншими організаційними культурами; індивідуальному та командному навчанню і розвитку; оптимальному внеску кожного з членів команди задля досягнення успішного завершення проєкту.

Все це регламентується командною угодою, яка являє собою «...набір поведінкових параметрів і робочих норм, встановлених командою проєкту і підтримуваних індивідуальними зобов'язаннями і зобов'язаннями команди проєкту» [38]. РМВоК 7 [38] розглядає разом загальні процеси управління та лідерство, як тандем, який забезпечить успішне завершення проєкту. Ці процеси пов'язані між собою, та впливають один на одного. Наприклад, централізоване або розподілене управління проєктом вимагають відповідних типів лідерства та управління командою.

Обраному для дослідження типу проєктів більш притаманне розподілене між командою управління проєктом, коли усі члени команди беруть на себе відповідальність за виконання робіт, а не лише один проєктний менеджер. В цьому випадку для завершення проєкту команда самоорганізується. Замість призначеного менеджера проєкту хтось із команди, по черзі, може виконувати функції фасилітатора, забезпечуючи спілкування, співробітництво та взаємодію.

Але команді потрібен керманич, РМВоК 7 [38] називає такий стиль керівництва «Servant leadership», що можна перекласти, як «службове керівництво, або службове лідерство». Тоді особу, яка застосовує «Servant leadership», в рамках дисертаційного дослідження будемо називати «координатор команди ІТ-проєкту» (ККІТП), на якого на відміну від керівника команди, покладають такі обов'язки:

- забезпечувати індивідуальне зростання членів команди проєкту;
- слідкувати, чи стають члени команди проєкту більш здоровими, мудрими, вільними і самостійними;
- відстежувати, чи стануть члени проєктної групи «служуючими лідерами»?

**Визначення 1.1. Координатор команди ІТ-проєкту (ККІТП)** - особа, яка використовує стиль керівництва проєктом «Servant leadership», дозволяє проєктним командам самоорганізовуватися, підвищувати рівень автономії, передаючи прийняття рішень членам команди ІТ-проєкту.

Лідерська поведінка ККІТП включає в себе:

- усунення перешкод, які можуть заважати роботі команді ІТ-проєкту;

- створення «антидиверсійного щита» від внутрішніх і зовнішніх чинників, які відволікають команду проєкту від поточних цілей;
- ККІТП застосовує інструменти та підтримку, щоб команда проєкту залишалася задоволеною і продуктивною, вивчає, що мотивує членів проєктної групи та які способи винагороди за якісну роботу допомагають зберегти задоволеність членів проєктної групи.

Динаміку розвитку команди впродовж ЖЦП описують дві моделі: «Сходи Такмана» та «Модель ефективності команди Drexler/Sibbet».

Модернізована модель Брюса Такмана з доданою п'ятою стадією [38], включає в себе етапи: формування, штурму, нормування, виконання та перерви (додаток Б). В моделі автор виділив психо-емоційні стани особи, яка попадає до проєктної команди. Модель відповідає предмету дисертаційного дослідження, але в ній відсутні механізми формування команди, управління під час штурму, та покрокового переходу до фази нормування та виконання. Крім того, що модель припускає кооперацію, взаємодію членів команди проєкту задля отримання синергії, вона не враховує особливості управління ІТ-проєктами, де команда працює на постійній основі, не розформовується після закінчення проєкту, та відповідно не збирається «з нуля». Можливо, за особливостями ІТ-проєкту можуть бути додані відповідні ролі для врахування саме цих особливостей. Така ситуація має місце, коли команда (можливо команди) є постійними співробітниками ІТ-компанії. Тому доробку та вдосконалення моделі потрібно направити на згадані вище особливості через механізми формування культури та зрілості команди ІТ-проєкту.

Друга модель - це модель ефективності команди Drexler-Sibbet [38]. Автори розробили модель командної роботи, що складається з семи кроків та описує поведінковий стан команди з прив'язкою до етапів ЖЦП. Перші кроки (1 – 4) описують етапи створення проєктної групи, останні (5 – 7) описують стан команди при створенні продукту проєкту (додаток В).



До обрання засобів формування команди IT-проєкту слід визначитися із загальною формою організації команди, яка буде найкращою для нового проєкту та оптимальною в межах корпоративної культури IT-компанії.

Software Extension to the PMBoK [15] рекомендує такі форми організації команди - спеціалізована або універсальна, в першому випадку члени команди мають вузьку спеціалізацію та працюють над одним, своїм, проєктом. В другому - члени команди мають універсальні широкі компетентності та працюють одночасно над декількома проєктами. Вузька спеціалізація та окремі функціональні підрозділи додають складності в площині економічних рішень управління командою та проєктом. Тому менеджер проєкту повинен знаходити компроміс задля забезпечення успішного завершення проєкту.

Наступна форма організації команди проєкту - це віртуальна, або розміщена в одному офісі. Складні проєкти інноваційного типу, які є предметом дисертаційного дослідження, вимагають нетривіальних рішень, формування яких відбувається колегіально, при особистому спілкуванні усіх членів проєктної команди разом. І в цьому випадку фахівці широкого профілю можуть генерувати значно більше рішень, ніж вузькоспеціалізовані фахівці.

Наступна форма організації команди проєкту - це постійна діюча, або тимчасова, для кожного нового проєкту.

За класичним підходом для кожного ПП створюється нова проєктна команда і розпускається, коли продукт проєкту буде переданий замовнику.

Перевагою постійно діючих команд є збереження знань, що утворилися від взаємодії, навчання команд та досягнутої синергії. Завдяки цьому команда підтримує високий рівень продуктивності, її поведінка стає більш передбачуваною [15].

Але, як було зазначено вище, постійно діюча команда вимагає більших фінансових витрат та управлінського часу.

Agile Practice Guide Bundle [25] робить наголос на індивідів в команді проєкту та їх взаємодії. При тому, що Agile оптимізує потік цінності, через швидку доставку MVP клієнту, а не через визначення трудових ресурсів та їх

завантаженності. Автори [25] рекомендують «.. створювати проєкти навколо мотивованих людей, надавши їм середовище, необхідну підтримку, та довіру».

Якщо команда навчена та орієнтована на збільшення потоку цінності, вона починає краще співпрацювати, швидше закінчувати роботи, за рахунок того, що члени команди не виконують багатозадачні роботи.

Agile команди зосереджуються на швидкому створенні продукту, задля отримання зворотного зв'язку від Замовника. На практиці ефективні Agile команди мають від трьох до дев'яти членів [25]. Agile заохочує самоуправляючі команди, де члени групи самостійно вирішують, хто виконуватиме роботу в межах, визначених наступним періодом. У Agile командах використовуються три загальні ролі: міжфункціональні члени команди (Cross-functional team member), власник продукту (Product owner) та фасілітатор команди (Team facilitator), або як було запропоновано вище - координатор команди ІТ-проєкту.

Міжфункціональні члени команди - то є професіонали, що регулярно мають розробляти та поставляти Замовнику поточну робочу версію продукту, зазвичай то є дизайнери, розробники, тестувальники та інші необхідні для створення продукту проєкту ролі.

Власник продукту несе відповідальність за створення продукту. Це проявляється через ранжування робіт відповідно до комерційної вартості. Для цього вони працюють зі своїми командами щодня, надаючи відгуки про продукт і визначаючи напрямки щодо наступного функціоналу, який необхідно розробити, та із зацікавленими сторонами, клієнтами та командами, щоб визначити напрямки продукту [25].

Зазвичай Product owner повинні мати бізнес-компетентності та досвід у прийнятті рішень. За необхідності власник продукту проводить консультації з фахівцями, що мають досвід роботи в галузі розробки, або у предметній галузі клієнтів. Все це потребує постійного навчання задля надбання нових компетентностей.

Усі Agile команди потребують ККІТП для: формування лідерських навичок у членів команди; навчання; усунення перешкод.

### 1.3 Аналіз інформаційних технологій управління ІТ-проєктами

Сьогодні ефективне управління проєктами неможливе без використання сучасних програмних засобів, оскільки зростають розміри проєктів, частота їх виконання, обсяги інформації. Перші програми для управління проєктами були розроблені майже сорок років тому, на початку 60-х років. В основу цих систем були покладені алгоритми сіткового планування і розрахунку параметрів проєкту у часі за методом критичного шляху. Пізніше в системи додані можливості ресурсного і бюджетного планування, засоби контролю за ходом виконання проєкту. Особливо значного розвитку вони набули у 80–90-х роках минулого століття.

Автоматизовані системи управління проєктами містять такі структурні елементи:

- засоби для календарно-сіткового планування;
- засоби для вирішення окремих завдань (розробка бюджетів, аналіз ризиків, управління контрактами, часом тощо);
- засоби для спрощення і обмеження доступу до проєктних даних;
- засоби для організації комунікацій;
- засоби для інтеграції з іншими прикладними програмами.

Найбільш відомий та доступний програмний засіб з управління проєктами є MS Project. Цей програмний продукт призначений для менеджерів проєктів для розробки планів проєкту, розподілу ресурсів і фінансів відповідно до поставлених завдань [44, 45]. Додаток відрізняється простотою і зручністю, дозволяє відстежити хід виконання конкретного проєкту з детальним аналізом обсягу виконаної роботи, та можливістю формування необхідних звітів.

Кожен член команди проєкту в режимі онлайн може вносити власні пропозиції і бачити отримані зміни. Результатом впровадження даного рішення є робота, виконана в повному обсязі і в точно встановлені терміни.

В процесі роботи програми, призначеної для управління окремими проєктами, менеджеру доступний широкий функціонал, заснований на роботі з

екранами, які містять фільтри, таблиці і угруповання. У програмі Microsoft Project є кілька десятків екранів, здатних організувати відомості, призначені для різних цілей [44, 45].

ProjectLibre призначений для управління проєктами та має відкритий код. Додаток працює на платформі Java, що дозволяє його запускати в різних операційних системах [46]. Вважається, що ProjectLibre є основною альтернативою з відкритим кодом для Microsoft Project, та повністю сумісна з версіями 2003, 2007 та 2010.

Trello - багатоплатформна система з управління проєктами, яка є безкоштовною [47]. Управління проєктами в ній ґрунтується на технології Kanban. Проєкти розміщуються на типових Kanban-дошках зі списками. Списки містять карти, або завдання, які перетягуються за технологією Kanban, відображаючи формування продукту проєкту (або його окремої функції) від ідеї до тестування [47].

Pivotal Tracker позиціонується як перевірений засіб управління проєктами [48]. Додаток забезпечує спільний погляд на пріоритети команди, процеси співпраці, та інструменти для аналізу прогресу. Tracker візуалізує сферу діяльності, чітко визначає пріоритети та дозволяє команді зосередитись на можливих змінах. Pivotal Tracker має інструменти для адаптації та розвитку, через формування пріоритетів на керовані фрагменти та інше [48].

Ganttter - це додаток з управління проєктами на основі діаграм Ганта, реалізує створення та редагування планів проєктів, які можуть бути повністю інтегровані з Google. Має всі можливості настільних програмних продуктів з управління проєктами, таких як MS Project, та, разом з тим, має всі переваги хмарного сховища. Ganttter був розроблений для управління проєктами в Інтернеті. Користувачам Google надається двонаправлена синхронізація завдань із розкладів Ganttter в їх календарі Google, зберігання файлів на Google Drive і Google Team Drive, спільне редагування в режимі реального часу, вбудовані коментарі Google, можливість запуску чату Google з членами команди з їх розкладу Ganttter [49].

Worksection - український онлайн сервіс з управління проєктами, мобільна та багатомовна система управління [50]. Серед функціональних можливостей слід відзначити вбудований тайм-трекінг з розрахунком витраченого часу у форматі час/гроші, або лише у годинах, що ідеально для компаній з погодинними ставками. Worksection може бути застосований як для невеликих команд, так і великого бізнесу без прив'язки до сфери діяльності.

Для проєктів, в яких для розробки застосовано гнучкі технології, в якості засобів автоматизації управління необхідно обирати відповідне програмне забезпечення. Розробником Jira є компанія Atlassian (Австралія). Система призначена для відстеження помилок, управління завданнями, управління agile-проєктами з хмарною і серверною версіями. Вихідний код є відкритим для скачування [51].

Youtrack призначена для гнучкого управління проєктами для розробників ПЗ. Розробник — JetBrains (Чехія). Доступна хмарна та десктопна версії. Вартість користування зростає пропорційно кількості користувачів, для команди до 10 осіб - безкоштовна, також як і Jira [52].

Не менш інтуїтивно зрозумілим для проєктної діяльності є Бітрікс24. Це професійне програмне забезпечення з великою кількістю функцій, що спрямоване на цілісне управління проєктами та співпрацю з виконавцями, клієнтами залежно від специфіки проєкту. Указане програмне забезпечення спрямоване на покращання співпраці в проєкті, оскільки поєднує різноманітні засоби комунікації в одному програмному рішенні. Вираження цього відбувається через вбудований інструмент для спілкування в режимі чату та відеоконференцій для спілкування в режимі реального часу. Серед вбудованих функцій варто виокремити наявність календарів, як приватних, так і спільних. Означене програмне забезпечення може перетворити надісланий електронний лист у завдання для виконавців проєкту, а також функціонувати як приватна соціальна мережа. Користувачі мають змогу співпрацювати за допомогою обміну та зберігання документів. Важливим елементом є наявність

повнофункціональної системи управління персоналом, робочими звітами та автоматизацією робочого процесу.

Сучасні програмні засоби управління проектами стрімко розвиваються з врахуванням тенденцій ведення проєктів сьогодні – розподіленої роботи над проектами, віртуальних та міжкультурних команд, високої нестабільності середовища проєктів.

#### **1.4 Постановка задачі дослідження**

За результатами проведеного огляду та аналізу існуючих моделей, методів та інформаційних засобів управління командами ІТ-проєктів та проблем предметної області, була сформульована задача дисертаційного дослідження.

В дисертаційній роботі поставлено актуальне **науково-прикладне завдання**, що полягає в розробці та вдосконаленні існуючих моделей, методів та інформаційних засобів управління командами ІТ-проєктів з інноваційною складовою.

В роботі запропоновано розглядати клас ІТ-проєктів, що мають інноваційну складову, та мають визначені особливості:

- використання інноваційних технологій в розробці продукту проєкту;
- відсутність досвіду команди проєкту у виконанні задач проєкту;
- високий рівень напруженості, конфліктності в команді проєкту;
- велика кількість ризиків проєкту;
- розмитість вимог Замовника до продукту проєкту.

Саме до таких проєктів сучасна теорія управління проектами пропонує застосовувати креативні техніки, серед яких є технології дизайн-мислення, MVP, Proof of concept.

Тому за необхідне вважаємо удосконалити засоби управління командою ІТ-проєкту, що буде застосовувати новітні креативні техніки управління в межах гнучкого управління ІТ-проєктами.

**Метою дисертаційної роботи** підвищення ефективності управління командою ІТ-проєктів з інноваційною складовою за рахунок розробки та вдосконалення креативних моделей, методів та інформаційних засобів управління з використанням техніки дизайн-мислення та концепції створення МЖП.

Для досягнення вказаної мети в дисертаційній роботі виділено наступні **задачі дослідження**:

- аналіз предметної області, а саме особливостей ІТ-проєктів з інноваційною складовою та особливостей управління ними, огляд сучасних моделей, методів та інформаційних засобів управління командами таких проєктів;
- розробка концептуальної моделі креативного управління командою ІТ-проєкту;
- розробка методу управління креативною командою ІТ-проєкту;
- розробка семіотичної моделі управління командою ІТ-проєкту;
- удосконалення методу формування креативної команди ІТ-проєкту;
- вдосконалення термінологічної основи управління креативною командою ІТ-проєктів;
- розробка інформаційної технології креативного управління командою ІТ-проєкту;
- застосування на практиці розроблених інструментів креативного управління командою ІТ-проєкту.

## **1.5 Висновки до першого розділу**

1. Для формування області дослідження та більш чіткого розуміння предмету дослідження у першому розділі було проаналізовано особливості сучасних ІТ-проєктів, досліджено специфіку управління ІТ-проєктами з

врахуванням їхньої інноваційності та відповідної участі членів команди проекту, проаналізовано засоби управління такими проектами.

2. Проведено огляд підходів до управління командами ІТ-проектів та подано результати аналізу управління командами, що використовують гнучкі технології розробки програмного продукту.

3. Подано огляд сучасних креативних технологій управління ІТ-проектами, що мають у собі інноваційну складову та застосовують гнучкі технології. Це, насамперед, технології дизайн-мислення, технології створення MVP, та технологія Proof of Concept.

4. Проведено аналіз інформаційних технологій управління проектами. Слід окремо зазначити, що використання автоматизованих засобів управління у будь-яких сферах діяльності, призводить до знеособлення та віддалення учасників процесів, і деяких випадках це може дати позитивних ефект. Але у клієнт-орієнтованому управлінні проектами, ціннісно-орієнтованому управлінні проектами, коли задоволення стейкхолдерів є головним критерієм успішного завершення проекту, без проявленої емпатії команди до проблем замовника неможливо досягти того успіху. Тому розробка засобів управління командою ІТ-проекту з використанням технологій дизайн-мислення та MVP є актуальною.

5. Виконана постановка актуального науково-прикладного завдання дослідження, що полягає в розробці та вдосконаленні існуючих моделей, методів та інформаційних засобів управління командами ІТ-проектів з інноваційною складовою.

6. За результатами дослідження опубліковано 2 статті [32, 42], та 2 тези доповідей на конференціях [2, 18].



## Список використаних джерел до розділу 1

1. Danchenko O., Amalahu V., Zarutsky S., Bielova O. IT projects: features, characteristics and classification. *Управління проектами: стан та перспективи : Матеріали XVII Міжнародної науково-практичної конференції*. Миколаїв : Видавець Торубара В.В., 2021. 130 с., с.107-108.

2. Близнюкова І.О., Данченко О.Б., Тесленко П.О. Аналіз сучасних визначень ІТ-проектів. *Управління проектами у розвитку суспільства. Тема: «Управління проектами в умовах пандемії COVID-19»: тези доповідей*. Київ : КНУБА, 2021. – С.100-104.

URL:<http://eprints.kname.edu.ua/58522/1/%D0%A2%D0%B5%D0%B7%D0%B82021.pdf>.

3. Коваленко О.О., Денисюк А.В., Бажан В.М. Порівняльний аналіз методологій розробки ІТ продукту. *Вінницький національний технічний університет*. URL: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2021/paper/download/12199/10302>.

4. Bedrii D.I., Semko I., Krylov V. IT-projects in power engineering. Project, Program, Portfolio Management. *Proceccing of the Fourth International Scientific and Practical Conference 06-07 December 2019*. Book 2. Odesa, ONPU, 2019. p. 16-20. ISSN 2522-9435. URL: <https://www.scopus.com/record/display.uri?eid=2-s2.0-85105565728&origin=inward&txGid=f2b8fd2b28f784abcaec6504d3b8ca16>.

5. Богославец А.А. Классификация IT-проектів. *Комунальне господарство міст*. Х.: 2014. Випуск 118. С. 56–59.

6. Глушенкова А.А. Особливості управління інноваційними проектами у сфері телекомунікацій та інформатизації. *Економіка. Менеджмент. Бізнес*, 2015. №4 (14). С. 72 – 77.

7. Катренко А.В. Управління ІТ-проектами. [Книга 1. Стандарти, моделі та методи управління проектами] : [підручник]. Львів: «Новий Світ-2000», 2011. 550 с.

8. Robert T. Futrell. Quality Software Project Management. Prentice Hall, 2002, 1639 p. ISBN 8177587536, 9788177587531.
9. Murray Cantor. Object-Oriented Project Management with UML. Publisher: John Wiley & Sons, Inc, 1998. – 388 p. ISBN: 0471253030.
10. Walker Royce. (2000) Software project management: a unified framework. The Addison-Wesley object technology series. Includes bibliographical references and index. ISBN 0-201-30958-0.
11. Dr. Winston W. Royce. Managing the development of large software systems. Proc. IEEE WESCON, Aug 1970. URL: <http://www-scf.usc.edu/~csci201/lectures/Lecture11/royce1970.pdf>.
12. Прокопенко Т.О., Поволоцький Я.О. Система критеріїв оцінювання ефективності проєктів галузі інформаційних технологій. *Вісник Черкаського державного технологічного університету*, 2022. № 4. С.23–30. DOI: <https://doi.org/10.24025/2306-4412.4.2022.271448>.
13. Підходи до управління програмними проєктами у SWEBOK V3.: URL: <https://www.researchgate.net/publication/316493834>.
14. SWEBOK V3. Guide to the Software Engineering Body of Knowledge. Version 3.0. URL: <https://cs.fit.edu/~kgallagher/Schtick/Serious/SWEBOKv3.pdf>.
15. Software Extension to the PMBOK® Guide Fifth Edition. Project Management Institute. Publ., 2013. [www.PMI.org](http://www.PMI.org). 240 p.
16. Sarah K. White and Lynn Greiner. (2022). What is ITIL? Your guide to the IT Infrastructure Library URL: <https://www.cio.com/article/272361/infrastructure-it-infrastructure-library-til-definition-and-solutions.html>.
17. Scott Berkun (2005). The Art of Project Management. Released April 2005. Publisher: O'Reilly Media, Inc. ISBN: 9780596007867. URL: <https://www.oreilly.com/library/view/the-art-of/0596007868>.
18. Близнюкова І.О., Тесленко П.О., Данченко О.Б. Інструменти управління ІТ-проєктами з інноваціями. *Збірка тез VIII Міжнародної НТК «Інформатика. Культура. Технології» ІКТ-2021*. Одеса : ІКС, 2021. С. 84-86. URL: [http://ics\\_conf.tilda.ws/ict\\_eng](http://ics_conf.tilda.ws/ict_eng).

19. Robert C. Martin (2019) Clean Agile. Back to Basics. Boston. 2020. Pearson Education, Inc. 1st edition, 240 p. ISBN 0135781868.
20. Складові системи планування проекту. URL : <https://buklib.net/books/23851/>.
21. Software Engineering Body of Knowledge (SWEBOK). URL : <https://www.computer.org/education/bodies-of-knowledge/software-engineering>.
22. A Guide to the Project Management Body of Knowledge (PMBOK® Guide). Sixth Edition. USA. PMI, 2017. 756 p.
23. Управління IT проектами. URL : <http://dspace.wunu.edu.ua /retrieve/19638/%D0%9B%D0% B5%D0%BA%D1%86%D1 %96% D1%97.pdf>.
24. Дослідження методів підтримки прийняття рішень при удосконаленні процесу розробки IT-проекту. URL : [https://openarchive.nure.ua/bitstream/document/19072/1/2021\\_M\\_IUS\\_Potehin\\_SV.pdf](https://openarchive.nure.ua/bitstream/document/19072/1/2021_M_IUS_Potehin_SV.pdf) 6.
25. A Guide to the Project Management Body of Knowledge (PMBOK® Guide- Sixth Edition / Agile Practice Guide Bundle (HINDI). Project Management Institute. Publ., 2017. www.PMI.org. 115 p.
26. Crystal Clear методологія — простота и совершенство для бизнес-приложений <https://worksection.com/blog/crystal-clear.html>.
27. Schwaber K. and J. Sutherland The Scrum Guide. The definitive Guide to Scrum: The Rules of the Game. 2017, 19 p. URL : <https://scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>.
28. Schwaber, Ken; Beedle, Mike (2002). Agile software development with Scrum // Prentice HallPTR Upper Saddle River, NJ, USA, 2001.
29. J. Sutherland. (2019) Scrum. A revolutionary approach to building teams, beating deadlines and boosting productivity. Random House Business ISBN: 9781847941107.
30. Real Life Examples of Agile Methodology (2018) URL : [https://www.growthaccelerationpartners.com/blog/real-life-examples-of-agilemethodology#content\\_upgrade\\_popup\\_9676](https://www.growthaccelerationpartners.com/blog/real-life-examples-of-agilemethodology#content_upgrade_popup_9676).

31. Прокопенко Т.О., Підкуйко О.І. Розробка графоаналітичної моделі ситуаційного управління проектом в умовах SCRUM у сфері інформаційних технологій. *Вісник Черкаського державного технологічного університету*, 2022. №2. С 4–10. DOI: <https://doi.org/10.24025/2306-4412.2.2022.261704>.
32. Близнюкова І.О., Семко І.Б., Кійко С.Г. Огляд сучасних методологій управління командами ІТ-проектів. *Управління розвитком складних систем*. К.: КНУБА, 2020. №43. С. 100-104. DOI: 10.32347/2412-9933.2020.43.60-66. URL: <http://mdcs.knuba.edu.ua/article/view/219835>.
33. Hamilton T. Scrum vs Kanban – Difference Between Them. URL: <https://www.guru99.com/scrum-vs-kanban.html>.
34. Джеймс Вумек, Деніел Джонс. Ощадливе виробництво. Фабула, 2018. 448 с. ISBN978-617-09-3892-3.
35. Zanora V., Momot S., Bedrii D., Fonar L. Conflict management in enterprise development project teams. *Academy review*. Dnipro: Alfred Nobel Univ, 2023. Is. 1(58). P. 187-204. ISSN 2074-5354 (print), ISSN 2522-9745 (online). DOI: <https://doi.org/10.32342/2074-5354-2023-1-58-14>. URL: <https://acadrev.duan.edu.ua/images/PDF/2023/1/15.pdf>. WOS: 000982340500014.
36. Новохацька Д.В. Особливості та проблеми реалізації ІТ-проектів в Україні. *Вісник ЧДТУ*, 2016. № 2. С.72-77.
37. Laing Samantha and Hryvs Karen (2013). *Growing agile: a coach's guide to training Scrum*. Publisher: Growing Agile; 1st edition, 195 p. ASIN : B00E8DFMT4.
38. A Guide to the Project Management Body of Knowledge (PMBOK® Guide) and the Standard for Project Management. Seventh Edition. USA. PMI, 2021. 274 p.
39. Івченкова О. Ю., Лях А. О. Аналіз моделей і методів розподілу трудових ресурсів в управлінні реалізацією портфеля ІТ-проектів. *Вісник економічної науки України*, 2016. № 2. С.87-91.
40. Шашкова, Н., Фадєєва, І., Казакова, Т. Управління проектами в ІТ сфері: застосування гнучких методологій. *Scientific Notes of Lviv University of Business and Law*, 2021. 28. С.166-172. Retrieved from URL: <https://nzlubp.org.ua/index.php/journal/article/view/402>.

41. Bushuyeva N., Bushuiev D., Bushuieva V. Agile leadership of managing innovation projects. *Сучасний стан наукових досліджень та технологій в промисловості*, 2019. № 4. С. 77-84. DOI: <https://doi.org/10.30837/2522-9818.2019.10.077>. URL: <https://journals.urau.ua/itssi/article/view/2522-9818.2019.10.077>.

42. Близнюкова І.О. Концепція створення мінімально життєздатного продукту та дизайн-мислення в управлінні командою ІТ-проєкту / Близнюкова І.О., Тесленко П.О., Данченко О.Б., Меленчук В.М. *Вісник національного технічного університету «ХПІ» : Зб.наук.пр. Серія : Стратегічне управління, управління портфелями програмами та проєктами*. Харків: НТУ «ХПІ», 2021. №2(4). С. 11 – 17. DOI: 10.20998/2413-3000.2021.4.2. URL: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/52329>.

43. Jalote Pankaj (2002) Software project management in practice. Boston : Addison-Wesley. ISBN 9780201737219.

44. Добровська Л.М., Аверьянова О.В. Управління ІТ-проєктами в Microsoft Project: Комп'ютерний практикум URL: навчальний посібник для студентів спеціальності 122 “Комп'ютерні науки” для всіх спеціалізацій. Київ: КПІ ім. Ігоря Сікорського, 2020 – 152 с.

45. Використання програми Microsoft Project, Planner, To Do або програми "Завдання" в Teams. URL: <https://support.microsoft.com/uk-ua/office>.

46. ProjectLibre: програма управління проєктами з відкритим кодом. URL: <https://www.linuxadictos.com/uk/projectlibre-un-programa-para-la-gestion-de-proyectos-de-codigo-abierto.html>.

47. Бодненко Д. М., Котик В. В., Мишковець С. С., Соколовська Т. В. Використання сервісу trello в професійній діяльності. URL : [https://elibrary.kubg.edu.ua/id/eprint/32911/1/1\\_Bodnwnko\\_Kotyky\\_Myshkovec\\_Sokolovska\\_IT-20.pdf](https://elibrary.kubg.edu.ua/id/eprint/32911/1/1_Bodnwnko_Kotyky_Myshkovec_Sokolovska_IT-20.pdf).

48. Pivotal Tracker is changing how teams build software—one story at a time. URL : <https://www.pivotaltracker.com>.

49. Управління проектом Gantt URL : <https://chrome.google.com/webstore/detail/gantt-project-management/himomacmcpodhkahelbnmaddladgjo?hl=uk>.

50. Worksection. URL : <https://worksection.com/ua>.

51. Jira: що це таке і навіщо потрібна ця система? URL : <http://hi-news.pp.ua/kompyuteri/9918-jira-scho-ce-take-navscho-potrjna-cya-sistema.html>.

52. Jira, Youtrack, Worksection. Детальний розбір. URL : <https://worksection.com/ua/blog/jira-youtrack-worksection.html>.

## РОЗДІЛ 2

### МОДЕЛІ КРЕАТИВНОГО УПРАВЛІННЯ КОМАНДАМИ ІТ-ПРОЄКТІВ

#### 2.1 Методологія та архітектура дослідження

##### 2.1.1 Архітектура дисертаційного дослідження

Реалії сучасного цифрового суспільства та цифрової економіки формують певні обмеження та вимоги до процесів управління проєктами та їхніх результатів, тобто до продуктів проєкту.

Головним, на думку автора, є «розворот до споживача», коли забезпечення якості продукту чи послуги розглядається через призму затребуваності кінцевим користувачем (так зване *customer experience*) [1].

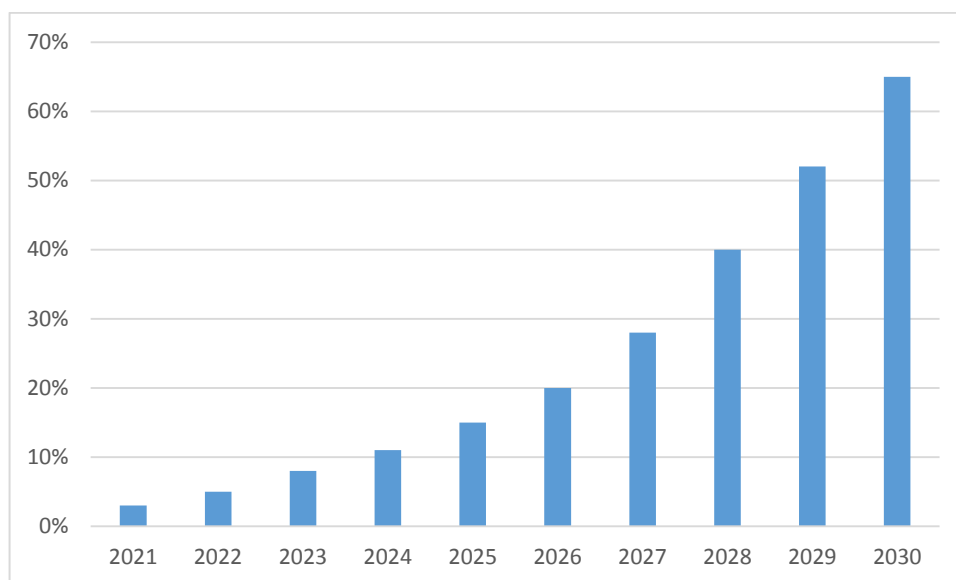
Наступним є суттєве скорочення часу від формування ідеї до виводу продукту на ринок (T2M). Такі вимоги не в змозі забезпечити важкі ієрархічні організаційні структури, що часто зустрічалися в організаціях останні 30 років, які відповідали аграрній, промисловій економіці, сфері послуг [2]. Їм на зміну прийшла економіка знань (або економіка даних) [2], яка вимагає невеликі Agile-команди, що застосовують ітеративний підхід, швидке прототипування у межах гнучких технологій розробки програмних продуктів [3].

Крім того, разом із клієнт-орієнтованою парадигмою створення продукту ІТ-проєкту, все більше уваги приділяють дослідженню працівників розумової діяльності (*Knowledge-intensive work*), здатних генерувати цінність для споживачів [4]. Вивчення клієнтського досвіду, так званого *customer experience*, сформувало нову парадигму «*experience economy*». Термін «*experience economy*» або «економіка вражень» був запропонований Б. Дж. Пайном II та Дж. Х. Гілмором [2].

Споживачі продукту проєкту (ППП) прагнуть отримати найкраще враження, емоцію від нього на всьому життєвому шляху PPP.

На основі цього можна припустити, щоб розглядати customer experience як додаткову вартість проектної діяльності. Тоді, наступним кроком формалізується необхідність у формуванні засобів управління командою проекту в термінах «employee experience». Розробка бізнес-процесів проекту повинна спиратися на потреби людини та людської залученості. Зазначене може бути сформовано на основі інструментів «дизайн-мислення» (design thinking), та «людино-центрованого дизайну» (HCD).

За даними сучасної концепції цифрової економіки України, що була прийнята урядом ще у 2018 році, доля цифрової економіки складає 3% [5] (рис.2.1). Але, за прогнозними даними, до 2030 року доля повинна бути не менш ніж 65%.



**Рис. 2.1. Частка цифрової економіки у загальному ВВП України, %**

Цифровою вважається тип економіки, де ключовими факторами та засобами виробництва є цифрові дані (бінарні, інформаційні тощо) та мережеві транзакції, а також їх використання як ресурсу, що дає змогу істотно збільшити ефективність та продуктивність діяльності та цінність для отриманих продуктів та послуг.

Тому вважаємо, що дизайн-мислення, як інструмент управління проектами, а саме процесами управління командою проекту, є вкрай необхідним в реальних вимогах сьогодення. Такі бізнес-процеси будуть орієнтовані на людину, на члена



команди проєкту, та створювати позитивний психологічний клімат, який забезпечить можливість співробітнику вирішувати завдання проєкту та формувати його продукт через використання та розвиток власних здібностей [1].

Таким чином, впровадження дизайн-мислення в методологію управління проєктами ІТ-компанії забезпечить формування таких відносин з членами команди, що вони стають частиною цієї компанії, при цьому акцент управління змінюється з «ролі співробітника в житті компанії» на «роль компанії в житті співробітника» [6].

Підхід «дизайн-мислення» (Design thinking) активно просувається як практичний інструмент і наукова дисципліна Hasso Plattner Institute і D.school (Стенфорд), спрямований на створення продукту або послуги, затребуваного споживачем [7, 8, 9, 10]. Технологія дизайн-мислення включає в себе п'ять етапів [1, 8, 11].

Перший етап дизайн-мислення є основним, бо саме в ньому повинна з'явитися інновація. Але, фахівцям технічних спеціальностей доволі складно організувати комунікацію з користувачами продукту та замовником проєкту, яка виокремить проблему для формування завдання на інновацію. Як показує практика, практикуючому інженеру досить складно відпрацювати в максимальній повноті саме емпатію [1]. Ще однією ключовою позицією дизайн-мислення є структурування інтелектуальної особистої та колективної роботи. Дизайн-мислення повинно формувати проєктну свідомість робочої групи [4].

У зв'язку з цим виникає важлива та актуальна задача з виокремлення місця дизайн-мислення в структурі проєктного управління, та розробки ефективних процесів управління командою ІТ-проєкту, задля забезпечення очікуваного замовником результату.

Одночасно з методикою дизайн-мислення пропонується в процесі управління командою інноваційного ІТ-проєкту застосовувати концепцію мінімально-життєздатного продукту, що часто використовується в сучасних ІТ-проєктах, особливо в тих, які реалізуються з гнучкими методологіями управління.

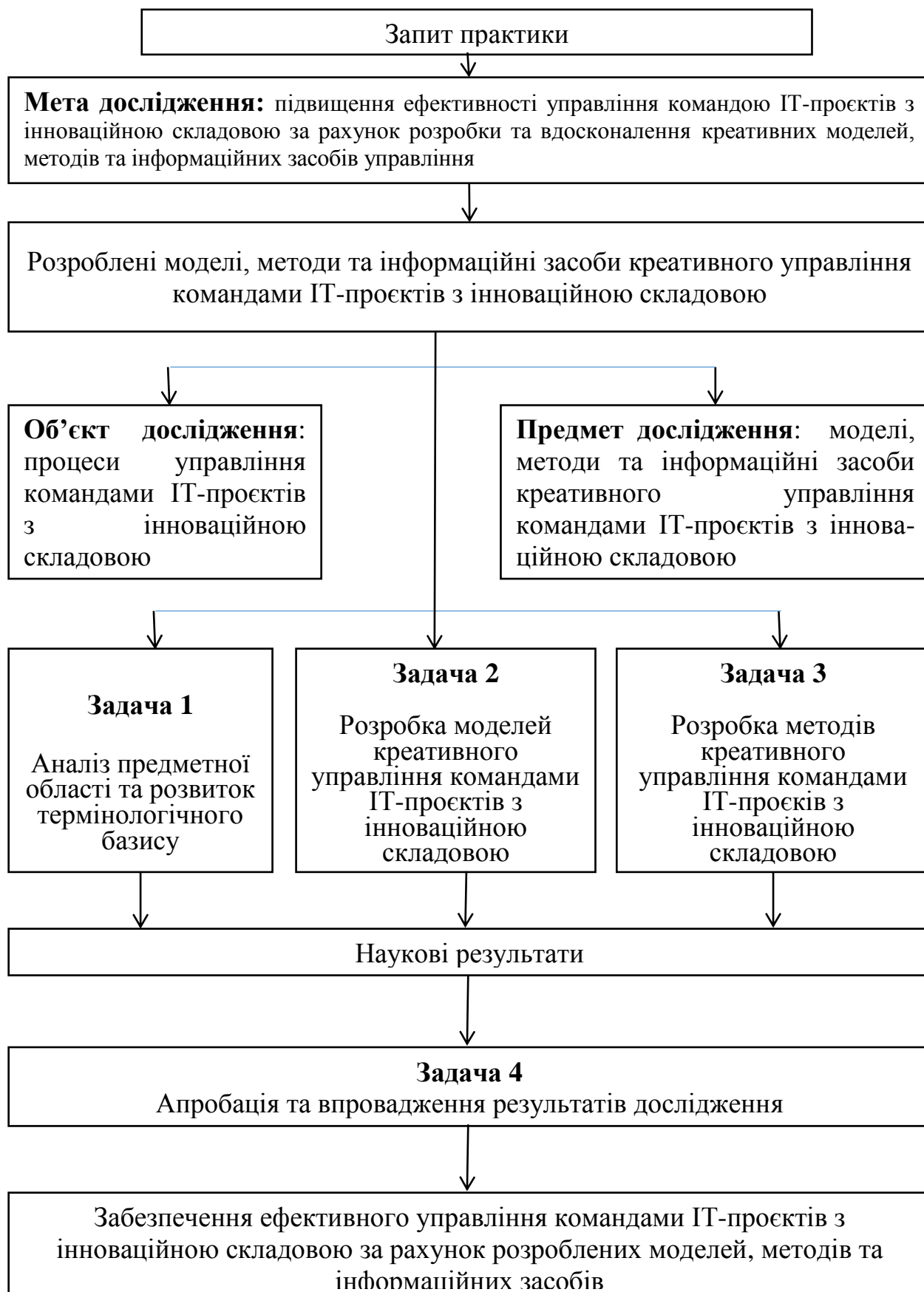
Виходячи з аналізу, проведеного в розділі 1 дисертаційного дослідження, можна стверджувати що існуючі підходи до управління командами ІТ-проектів не завжди забезпечують ефективне управління саме в проектах з інноваційною складовою.

У зв'язку з цим, виникає важливе та актуальне **науково-прикладне завдання**, обумовлене запитом практики, і направлене на підвищення ефективності управління командами ІТ-проектів з інноваційною складовою, що полягає в розробці нових, та вдосконаленні існуючих сучасних креативних моделей, методів та інформаційних засобів управління командами ІТ-проектів (а саме з використанням технології дизайн-мислення та концепції МЖП), відповідно до якого сформована наступна архітектура дослідження (рис. 2.2).

**Метою дисертаційної роботи** є підвищення ефективності управління командою ІТ-проектів з інноваційною складовою за рахунок розробки та вдосконалення креативних моделей, методів та інформаційних засобів управління з використанням техніки дизайн-мислення та концепції створення МЖП.

Для досягнення вказаної мети в дисертаційній роботі виділено наступні **задачі дослідження**:

- аналіз предметної області, а саме особливостей ІТ-проектів з інноваційною складовою та особливостей управління ними, огляд сучасних моделей, методів та інформаційних засобів управління командами таких проектів (задача 1);
- розробка концептуальної моделі креативного управління командою ІТ-проекту (підзадача задачі 2);
- розробка методу управління креативною командою ІТ-проекту (підзадача задачі 3);
- розробка семіотичної моделі управління командою ІТ-проекту (підзадача задачі 2);
- удосконалення методу формування креативної команди ІТ-проекту (підзадача задачі 3);



**Рис. 2.2. Архітектура дисертаційного дослідження**

- вдосконалення термінологічної основи управління креативною командою ІТ-проектів (підзадача задачі 1);
- розробка інформаційної технології креативного управління командою ІТ-проекту (підзадача задачі 4);
- застосування на практиці розроблених інструментів креативного управління командою ІТ-проекту (підзадача задачі 4).

Задля досягнення поставленої мети було використано такі методи та техніки управління: наукові основи управління ІТ-проектами, фахова література, наукові праці вчених, практики управління командами, системний підхід для дослідження та визначення характеристик процесів управління командою, порівняльний аналіз для формалізації показників системи управління командою ІТ-проектів, методи управління проектами для ідентифікації та оцінки необхідних компетентностей претендентів до команди, прогнозування для визначення мінімально-життєздатної команди проекту, оптимізаційні методи, генетичний алгоритм та технологія дизайн-мислення для розробки підходів та методів формування ефективної команди ІТ-проекту, ситуаційне та семіотичне моделювання для побудови моделі креативного управління командою ІТ-проекту. Інформаційним підґрунтям роботи є дані статистики ІТ-галузі.

### **2.1.2 Технологія дизайн-мислення**

Однією з особливостей ІТ-проектів можна вважати те, що більшість з них розробляються «на замовлення», тобто ІТ-проекти з інноваційною складовою є суто унікальні [12].

Це означає, що рівень новизни продукту достатньо високий, він перевищує середнє значення. Мається на увазі, що продукт проекту є новим, або порівняно з діючими аналогами, в ньому суттєво змінені його властивості та/або функціонал, суттєво змінена сфера застосування, тощо. Мова йде про те, що користувачам пропонується новий продукт, з яким вони не знайомі, не використовували, не припускали навіть можливості його існування.

Для виокремлення таких продуктів проєктів можна запропонувати для них назву «Три «НІ»» = «не знайомі, не використовували, не припускали навіть можливості його існування» [11].

Просуненням подібних продуктів на ринок збуту зараз займається так званий агресивний маркетинг, який нав'язує користувачеві продукти, які йому в принципі і не потрібні. Це виконується на основі різних психологічних технік.

Але, серед продуктів «3-НІ» обов'язково існують продукти, що будуть корисні для споживачів, та змінять їх життя на краще. Для того, щоб з'ясувати структуру такого продукту, використовують технології дизайн-мислення. Спочатку технологія дизайн-мислення застосовувалася лише для start-up проєктів. Але пізніше, цю технологію стали застосовувати й для загального управління проєктами. В [1] автори розглядали можливість застосування дизайн-мислення для управління ІТ-проєктами. Було вказано на те, що клієнт-орієнтована парадигма створення продукту ІТ-проєкту вимагатиме й відповідних змін в технологіях управління такими проєктами.

В роботі [1] показано, що впровадження дизайн-мислення в методологію управління проєктами ІТ-компанії створить такі умови, що члени команди стануть частиною цієї компанії, а такі співробітники вже будуть перейматися потребами замовників і користувачів [11].

Технологія дизайн-мислення структурована (рис. 2.3), має властивість системності і має взаємопов'язані етапи [1, 13, 14]. Перший етап - збір і аналіз вимог від стейкхолдерів, які прямо або опосередковано будуть втілені у майбутньому продукті проєкту. У класичних термінах дизайн-мислення, перший етап то є емпатія.

Другий етап - формулювання концепції майбутнього продукту з точки зору його призначення, або фокусування. Третій - генерація можливих рішень того, яким може бути майбутній продукт та вибір одного з них для подальшого створення прототипу. Четвертий етап - прототипування одного з відібраних варіантів для його перевірки, п'ятий - тестування, отримання зворотного зв'язку

від користувачів та замовника, внесення змін (добробка) прототипу, або повернення до фокусування та генерації ідей, якщо користувач незадоволений продуктом.



**Рис. 2.3. Технологія дизайн-мислення**

Однак, такий розподіл на етапи дещо умовний, і дана технологія — це щось більше, ніж процес з п'яти або семи кроків. Вказані на рисунку 2.3 лінії зворотного зв'язку передбачають повернення від процедури розробки продукту на попередні етапи, за умови незадоволеності замовником поточної версії продукту.

Проте такий схематизм дозволяє виділити деякі ролі, правила умовності для членів команди, моделі поведінки і формальні очікування, які у підсумку повинні сформувати принципи і концепцію управління командою проєкту [11].

Дизайн-мислення починається з емпатії — глибокого розуміння людей, яким вона призначена [15]. Команда проєкту, яка думає як дизайнери, ставить себе на місце замовників та користувачів. Передбачається, що управління проєктом вже клієнто-орієнтоване, але тут ще необхідно розуміння стейкхолдерів, з їх реальними проблемами, а не враховувати їх тільки як джерело власного доходу, або як набір демографічних даних про вік, рівень доходу та сімейного статусу і т.і. Використання дизайн-мислення передбачає глибоке осмислення їх емоційного статусу, потреб і побажань щодо майбутнього продукту проєкту.

Оскільки дизайн - це ще й процес винаходу, команда проєкту повинна мислити як дизайнери, і вважати себе творцями [16].

Різниця в тому, що класичний менеджмент знаходить рішення з «вже існуючого сьогодні», а дизайнери по суті «винаходять завтрашній день», тобто створюють те, чого немає сьогодні. І проблема в тому, що інструменти завтрашнього дня дуже рідко можуть бути отримані шляхом логічного висновку.

Дизайн-мислення передбачає ітераційний процес. Ухвалення рішення про те, «що рішення знайдено», буде прийматися на основі зворотного зв'язку від замовника, а саме про те, що поточна вдосконалена версія продукту проєкту задовольняє очікування замовника. Це висуває додаткові вимоги до команди проєкту, а саме необхідність постійного навчання або підвищення своїх компетентностей і навичок.

Класичний менеджмент передбачає прямий, лінійний метод вирішення: визначення завдання/проблеми, пошук декількох варіантів рішень і вибір одного з них.

Дизайн-мислення для успішних рішень пропонує експериментувати, причому експериментувати з емпатією, тобто експеримент проводиться виключно з метою задоволення очікувань замовника. А для цього знову потрібно постійно вчитися.

Ще одна відмінність між класичним менеджментом і дизайн-мисленням, яке повинно бути враховано в інструментах управління командою ІТ-проєкту, - це відмінності в базових передумовах і факторах, на основі яких приймаються рішення. Класичний менеджмент передбачає раціональність і об'єктивність. Рішення приймаються на основі економічної або технологічної логіки. Реальність точна і вимірювана кількісно. Існує «істина», і відповіді можуть бути «правильними» і «неправильними» [16]. З використанням технології дизайн-мислення рішення приймаються на основі почуттів і вражень людини. Реальність створюється людьми, які її проживають. Проєктні рішення пояснюються швидше емоціями, ніж логікою, і бажання вважається більш потужною мотивацією, ніж здоровий глузд.

У практиці дизайн-мислення постійно відбуваються ітерації — не тільки в часі, але і між рівнями абстракції, між загальною картиною і конкретними елементами. Команда проєкту, що викорисатвує технологію дизайн-мислення виготовляє моделі і прототипи, для візуалізації і матеріалізації ідей для замовника.

### **2.1.3 Технологія створення мінімально життєздатного продукту**

В комбінації з технологією дизайн-мислення доречним є використання концепції створення мінімально життєздатного продукту, як засобу для створення «прототипу» продукту ІТ-проєкту.

Тому наступним кроком необхідно формалізувати термін «прототип» – як поточну версію продукту проєкту, яка існує на поточному етапі життєвого циклу проєкту. В якості такого терміну пропонується використовувати «Мінімально життєздатний продукт» (МЖП). Цей термін найбільш повно та ємно розкриває результат процесів управління за технологією дизайн-мислення. Коротко його суть можна викласти наступним чином: команда ІТ-проєкту повинна у найкоротші строки подати замовникові прототип продукту проєкту. Зрозуміло, якщо строки мінімальні, то він володітиме мінімальним функціоналом, але ідея його така, щоб він продемонстрував працездатність майбутнього продукту проєкту [11].

В англійських дослідженнях МЖП називають Minimum Viable Product (MVP) - перша пропозиція клієнтам з мінімальним набором функцій, яка представляє максимальну цінність [17].

Як можна помітити з назви МЖП, існує дилема між "величиною мінімальності" в продукті проєкту і "кількістю його готовності" (життєздатністю). Засновники стартапів бояться, що користувачі зненавидять такий мало функціональний продукт. Однак Ерік Райс (Eric Ries), автор «Lean Startup», закликає не перейматись з цього приводу [17]. Він каже, що якщо реакція буде негативною, завжди можна все виправити в наступному релізі. Ітеративний



процес створення MVP передбачає, що реально повернутися у вихідну точку або на певний етап розробки.

Слово «viable» (життєздатний) в аббревіатурі MVP можна трактувати по-різному у різний спосіб [17]:

- придатний для тестування;
- придатний для використання;
- той, що подобається користувачам.

Для створення МЖП застосовують алгоритм з 4 етапів:

- Think it - поміркуй;
- Build it - побудуй;
- Ship it - розповсюджуй;
- Tweak it – налаштуй те, що побудував.

Етап обмірковування передбачає висування гіпотези, створювання опису та начерки прототипу, які стануть в нагоді для розробки. По завершенні гіпотез команда будує МЖП - створює простий функціонал, щоб визначитися, чи відповідають його функції потребам замовника та користувачів, чи ні. Якщо стейкхолдери не зацікавлені, від ідеї відмовляються [18, 19].

Коли задумка знаходить відгук у майбутніх користувачів продукту проєкту – її тестують і переробляють, поки не буде отриманий готовий продукт, який можна поширити на всіх користувачів [20].

Такий підхід дозволяє знизити ризики і економно витратити гроші на розробку продукту проєкту. Тобто, головною задачею використання ідеї створення МЖП є якомога раніше визначити, чи потрібен продукт проєкту замовнику, та чи представляє він собою цінність, чи буде той проєкт успішним [21, 22].

#### **2.1.4 Методика Proof of concept**

Перевірка концепції (PoC) — це демонстрація практичної здійсненності будь-якого методу, ідеї, технології, з метою доведення факту, що метод, ідея або

технологія працює [23]. Також помилково приймати PoC як "чернетку" проєкту, яку перед завершенням треба доопрацювати. Вважається, що PoC набагато ближче до дослідження, ніж до розробки працюючого продукту [23].

«Прототип» з етапу технології дизайн-мислення потрібен для перевірки ідеї у «фізичному розумінні», тобто у розумінні продукту проєкту, у розумінні MVP. У випадку інноваційних проєктів, стає за потребу перевірка ідеї у сенсі технологічного здійснення певних розрахунків, спроектованих систем тощо. Якщо ключовою функцією планованого продукту проєкту є, наприклад, розпізнавання образів з похибкою менш 1%, то в межах PoC ми перевіримо технічну можливість таких обчислень, а при створенні прототипу змовнику буде продемонстровано розташовування відповідних полів введення, та інтерфейсу роботи із зовнішніми камерами стеження.

Для неінноваційних проєктів перевірка концепції є не обов'язковим етапом проєкту, але для інноваційних проєктів відсутність PoC у разі збільшує невизначеність вже на початку проєкту, бо вона визначає загрозу безпеки, непередбачувану складність проєктів, дефіцит кваліфікованих кадрів. Тому застосування PoC з'ясовує не тільки працездатність майбутніх рішень, але й пропорційність витрат та ризиків проєкту [24].

В [25] проблему оцінки реалізуємості новітніх ідей розкрито через аналіз «непередбачуваних» проєктів, які мають заздалегідь невідомі події. Для управління такими проєктами автори [25] пропонують проактивне формування компетентностей проектної команди через методику визначення адаптаційного потенціалу членів команди, яка базується на прогнозуванні рівня компетентності та стійкості знань учасників проектної групи.

### **2.1.5 Креативне управління IT-проєктами**

Управління інноваційними проєктами приймає невизначеність, зміни та непередбачуваність як норму [26]. В цьому випадку застосування традиційних

методів управління проектами, які спираються на ретельне планування, майже неможливе [27].

Збільшення прогнозованості інноваційних проєктів може забезпечити використання креативного потенціалу команди [28]. Креативність має стає ключовою навичкою команди для забезпечення успішного завершення ІТ-проєкту.

Креативний (від англ. «creative») означає «творчий», як пошук неординарних підходів, розкриття інтелектуальної складової особистості. Під креативністю розуміється універсальна творча здатність, рівень творчої обдарованості, який є однією з характеристик особистості [29].

Визначення креативності має різне забарвлення та наголоси в залежності від сфери застосування [30-33].

У [32] креативність позиціонується як «здатність до конструктивного, нестандартного мислення та поведінки, усвідомлення і розвитку свого досвіду». Автор описує прояв креативності як швидкість та оригінальність мислення, схильності до естетичних цінностей, що дає можливість знаходити ефективні рішення та переборювати стереотипність мислення та поведінку у звичних ситуаціях, приймати нові, оригінальні рішення.

В креативному управлінні творчість виступає як засіб підвищення ефективності та джерело інновацій. Необхідність креативного управління виникає тоді, коли стандартні принципи та засоби управління не дають результату. Креативне управління - це сукупність засобів впливу задля забезпечення творчого підходу, формування індивідуальних здібностей та підвищення імовірності успішного завершення ІТ-проєктів з інноваційною складовою [34].

Креативне управління містить у собі [35]:

- оцінку творчого потенціалу особистості;
- створення творчої атмосфери в команді ІТ-проєкту, яка працює певний час разом, має свої традиції, неформальних лідерів та розподіл функціональних ролей;

- формування тимчасових творчих колективів задля участі в груповій динаміці;
- оцінка мотиваційних установок у найближчій перспективі завдань, що вимагають застосування творчих здібностей членів команди.

Окремо виділяють «продуктивну креативність», яка заснована на позитивній мотивації, заохоченні спілкування, на тренінгах та позитивній взаємодії членів команди. Така креативність проявляється через самореалізацію, підвищення потреб особистостей, надбання знань та навичок, появу нових ідей. Продуктивну креативність та позитивну мотивацію пов'язують із самореалізацією та підвищенням рівня потреб згідно з пірамідою А. Маслоу [27].

В [36] проаналізовано принципи формування креативного потенціалу команди проекту. Автори провели аналіз факторів успіху проекту, спираючись на креативний потенціал команди. В [36] показано, що високий рівень креативності команди може супроводжуватися великою кількістю організаційних ризиків. Причому в умовах невизначеності креативна команда ефективно здійснює структурне управління, забезпечує народження нових ідей, стратегій через обмін думок та комунікації [37]. Креативні команди оперативно аналізують складні проблеми та розробляють варіанти їх вирішення. І як висновок, автори [36] зазначають, що креативний потенціал команди є одним з вирішальних чинників успішного завершення проекту.

## **2.2 Концептуальна модель креативного управління командою ІТ-проекту**

Сформулювавши загальне бачення напряму дисертаційного дослідження у підрозділі 2.1, продовжимо формалізацію наукового пошуку у вирішенні завдання розробки нових моделей, методів та інформаційних засобів креативного управління командою ІТ-проекту задля підвищення ефективності управління ним.

В першому та другому розділах, в результаті аналізу сучасних підходів до управління розробкою ІТ-продукту було виокремлено дві техніки, які

привалюють на цьому етапі, технологія дизайн-мислення та концепція створення мінімально-життєздатного продукту. Автором запропоновано застосування згаданих технік до управління командами ІТ-проектів з інноваційною складовою, що є сьогодні актуальним та ефективним задля підвищення імовірності успішної реалізації проекту.

Головна ідея дослідження полягає у тому, що застосування креативних підходів до управління, а саме, технології дизайн-мислення, концепції МЖП та РоС, при розробці та реалізації ІТ-проектів потребує відповідних знань та навичок, носіями яких є безпосередньо члени згаданих проектних команд. Тому здається логічним застосувати ті ж самі принципи до створення та управління такими командами, тим паче, що відповідні знання вже будуть присутніми в таких командах на початковій фазі проекту.

Ця ідея дала змогу сформулювати концептуальну модель креативного управління командою ІТ-проекту з інноваційною складовою [38], графічне представлення якої подано на рис. 2.4.

Початковими даними в моделі є концепція ІТ-проекту, яка затверджена його замовником. У часовому просторі це відповідає ситуації за РМВоК [39], коли вже затверджений устав, сформовано бачення продукту проекту та очікування його користувачів (або їх проблеми, які має вирішити цей продукт). Початковими умовами до моделі є сформовані вимоги до продукту, на основі яких мають бути висунуті вимоги до виконавців та ролей команди проекту. Початкові умови формуються у блоках 1-3, але ці дії не входять безпосереднє до управління командою.

Слід зауважити, що розроблена концептуальна модель повністю відповідає загальному підходу управління командою проекту, а саме, складається з наступних етапів: формування команди, розвиток команди, моніторинг та контроль команди.

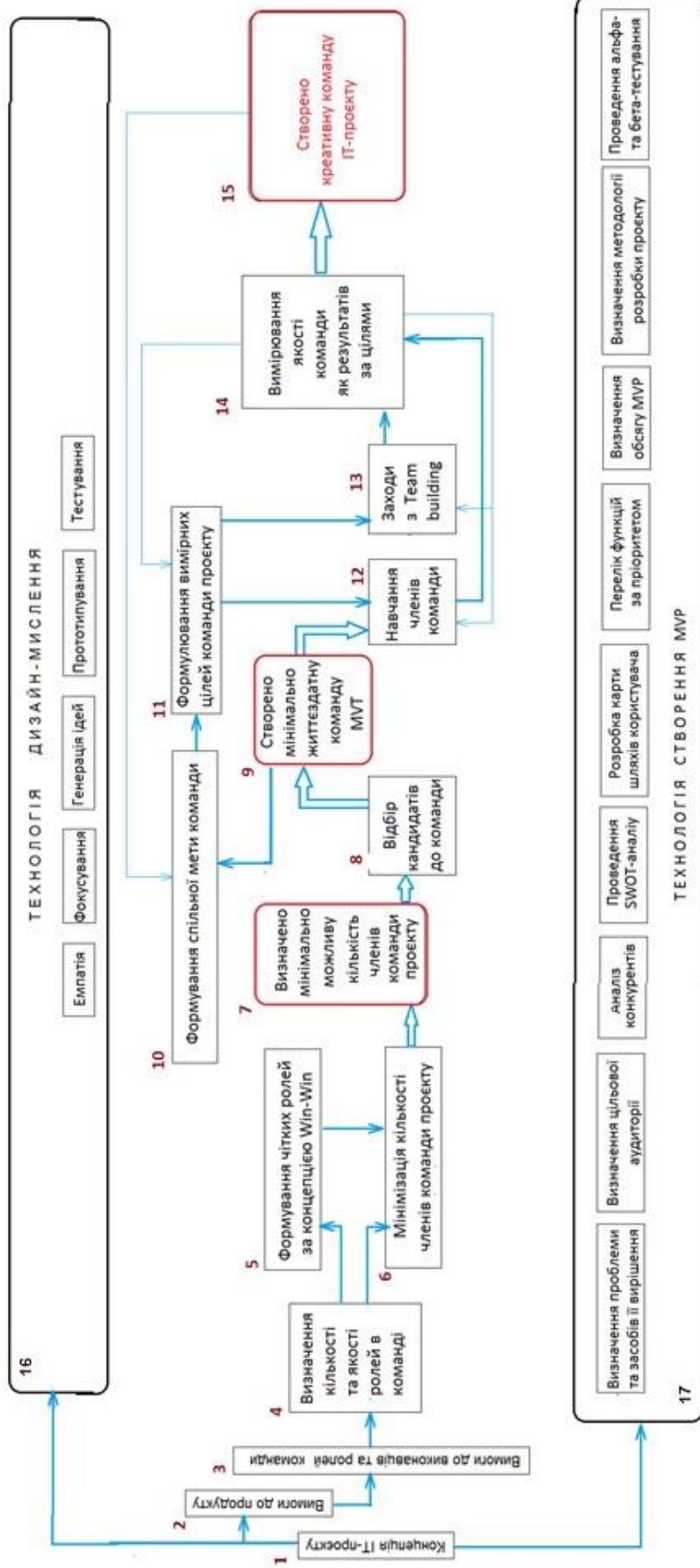


Рис. 2.4. Концептуальна модель управління командою IT-проекту з інноваційною складовою

Застосування креативних інструментів до управління командою, обумовлено вимогами сучасних ІТ-проектів та рекомендацій, що подані у новітній редакції РМВoK 7 [40].

Результатом моделі є сформована креативна команда ІТ-проекту (блок 15).

Таким чином, запропонована модель є відображенням процесу створення креативної команди ІТ-проекту з інноваційною складовою. Тут слід зауважити, що «створення креативної команди» не є першим етапом управління командою, за РМВoK, а саме її «формуванням», мається на увазі, що в результаті креативного управління командою ІТ-проекту за час його життєвого циклу буде створена креативна за визначенням команда.

Для подальшого обґрунтування положень концептуальної моделі необхідно сформулювати визначення: 1) креативної команди ІТ-проекту (ККІТП) та 2) процесу створення ККІТП.

**Визначення 2.1.** Креативна команда ІТ-проекту - це самокерована команда, що сама зорганізується, в якій відсутні централізовані (зовнішні) засоби управління.

Така команда сама бере на себе відповідальність, сама визначає проблеми/завдання та приймає рішення щодо їх вирішення. ККІТП формує загальні власні цілі (які не є цілями ІТ-проекту) з урахуванням кожного учасника (члена команди) та досягає їх відповідними шляхами впродовж проекту.

Перш за все, обмежимо сферу застосування такої команди. Будемо вважати, що ККІТП буде застосовуватися лише для ІТ-проектів, що мають у собі інноваційну складову та використовують гнучкі технології розробки програмного продукту.

Відмінністю від загальноновживаних визначень та сенсу команд ІТ-проекту, будемо вважати те, що ККІТП визначає для себе стан креативного/інтелектуального розвитку та досягає показників цього стану впродовж всього життєвого циклу ІТ-проекту.

**Визначення 2.2. Процес створення ККІТП** - це процес перетворення відібраних кандидатів у згуртовану команду, яка має високий ступінь креативності, а сам процес формування базується на технології дизайн-мислення.

*Обмеження процесу:* до команди не можуть бути відібрані особи, що не мають досвіду командної розробки ІТ-проектів, які мають інноваційну складову.

*Допущення процесу:* вважається, що кандидати в ККІТП мають навички розробки програмного продукту вище середнього рівня; приймали участь в ІТ-командах, що самокерувалися; мають здатність до емпатії, або мають властивості емпатії.

Емпатія має бути направлена у бік команди, тобто керівник проекту має визначити усі проблеми та цілі команди та забезпечити їх досягнення, через формування ідей, фокусування, створення прототипу ККІТП, який задовольнить очікування її членів. Крім того, креативна команда ІТ-проекту має опанувати феномен групової емпатії задля компенсації низького рівня емпатії у тих її членів, чия присутність в команді необхідна завдяки їхнім високим професійним здібностям.

Таким чином, першим елементом концептуальної моделі є визначення кількості та якості ролей у команді (блок 4). Як було зазначено вище, даний елемент спирається на вимоги до виконавців та ролей команди. Відмінністю зазначених сутностей є те, що перша, тобто вимоги, формується за технологією створення програмного продукту, а друга — приймається рішення про кількість з урахуванням наявних ресурсів та корпоративної культури ІТ-компанії, що реалізує проект.

Наступний елемент моделі направлений на мінімізацію кількості команди з урахуванням техніки MVP - мінімально життєздатного продукту, а саме направлений на створення мінімально життєздатної команди (блок 6). Але ця процедура доповнена використанням стратегії Win-Win, тобто вона направлена не на зменшення команди будь-якою ціною, можливо за рахунок якості розробки, або виснаження сил та особистого часу членів команди, а з урахуванням тих



критеріїв, коли кожен з членів команди, має залишитися «у виграшу». Ця процедура відбувається у блоці 5.

В результаті кроків 5 та 6 отримано перший проміжний результат, а саме, визначено мінімально можливу кількість членів команди проєкту з урахуванням елементів корпоративної культури організації та поточного стану трудових ресурсів в проєкті на даний момент. Цей результат на моделі зображено елементом червоного кольору (блок 7).

Після того, як мінімально можливу кількість визначено, слід переходити до безпосереднього відбору кандидатів до команди (блок 8). Метод відбору буде подано пізніше.

Після відбору отримаємо другий проміжний результат управління командою ІТ-проєкту, а саме, мінімально життєздатна команда (MVT) створена (блок 9).

Цей факт ініціює дві групи процесів: перший - починається формування спільної мети та вимірних цілей безпосередньо для обраних членів команди (блоки 10-11), а другий - навчання команди (блок 12).

На відміну від існуючих алгоритмів управління командою ІТ-проєктів, розроблена модель позиціонує членів команди як особистостей, які саме як особистості повинні зростати під час свого життя, і проєкт не має цьому заважати, а навпаки, сприяти цьому. Це означає, що застосована вище стратегія Win-Win залишається активованою впродовж усього життєвого циклу проєкту, тобто у блоці 10 реалізується алгоритм створення спільної мети команди на основі стратегії Win-Win.

Далі, спільна мета має бути трансформована у вимірні цілі команди проєкту (блок 11).

Друга група процесів, а саме навчання команди, розподіляється на два вектори. Перший - це професійне зростання (професійне включає в себе три складові: 1) вдосконалення навичок з програмування, 2) навчання технікам дизайн-мислення та MVP, 3) набуття та розширення властивостей емпатії. Другий вектор - соціально-психологічний розвиток, що пов'язаний із створенням

позитивного внутрішнього середовища проєкту, побудови команди, її згуртованості та ін. Ці процеси реалізовано у блоках 12 та 13, відповідно.

Заходи з навчання постійно перевіряються шляхом вимірювання якості команди (блок 14) [41].

Одним з показників якості є перелік компетентностей членів команди, які можуть бути виміряні стандартними, або спеціально розробленими засобами, наприклад як у [42].

Ці компетентності набуваються членами команди за раніше сформованими цілями (блок 11). Результати вимірювання формують петлі зворотного зв'язку до блоків 11 та 12, які дозволяють вносити корективи в перелік вимірних цілей команди проєкту (блок 11), або безпосередньо у заходи навчання (блок 12).

Якщо в результаті вимірювання показники якості відповідатимуть очікуванням команди, то можна вважати, що процес управління командою ІТ-проєкту завершено, і креативну команду - створено (блок 15).

Але зауважимо, що цей процес динамічний і необхідно перевіряти, чи не змінилися очікування та цілі членів команди впродовж ЖЦП, що реалізовано стрілкою зворотного зв'язку до блоку 10.

Крім того, зауважимо, що існують випадки, коли необхідно перевіряти спільну мету та вимірні цілі команди проєкту, які можуть змінюватись в зв'язку із зміною складу команди або із зміною або додаванням нової предметної галузі у проєктне середовище, яке потребує вивчення.

Таким чином, процес створення креативної команди відтворюється не один раз впродовж проєкту, а має відбуватися постійно, або із заданою циклічністю, в залежності від особливостей конкретного ІТ-проєкту.

Впродовж всього процесу управління ККІТП згідно концептуальної моделі застосовується технологія дизайн-мислення (блок 16) та концепція МЖП (блок 17).

### **2.3 Обґрунтування гомоморфного перетворення моделі креативного управління командою ІТ-проєкту**

Дослідження системи управління проєктами полягає у виділенні тих процесів, які відображають сутнісні характеристики всієї системи. Вибір таких процесів та його опис є основою моделювання. Під моделлю будемо розуміти ідеальний або матеріальний об'єкт, який в умовах експерименту замінює об'єкт-оригінал. Така заміна проявляється у спрощеному поданні поведінки самої системи.

Таке «спрощення» має ґрунтуватися на адекватності між процесами, що відбуваються в реальній системі та в моделі. Для складних систем, якою є система управління проєктами, використовують спрощені моделі, в яких залишають лише суттєві властивості оригіналу. У зв'язку з таким спрощенням, моделі поділяють на: ізоморфні та гомоморфні; ідеальні та матеріальні; описові, функціонуючі, зовнішньої подоби, семіотичні та моделі-аналоги.

В управлінні проєктами, на етапі планування, з метою зниження невизначеності традиційно використовують принцип декомпозиції (breakdown), коли весь ЖЦП поділяється на етапи, а ті, у свою чергу, — на процеси і т.п. Результатом таких процесів декомпозиції є ієрархічні структури проєкту: WBS, OBS, RBS тощо.

Побудова моделі процесів внаслідок декомпозиції є неформальною процедурою. Якість моделі залежить від кваліфікації дослідника, і навіть від мети моделювання. З іншого боку, точність моделі необхідна для зручності її використання під час прийняття проєктних рішень впродовж реалізації реальних проєктів. Отже, модель, що формується, повинна бути спрощеним поданням процесу управління командою ІТ-проєкту. Оцінку подібності моделі та реальних процесів необхідно проводити за однакових умов зовнішніх впливів, керуючих сигналів, тощо.

В результаті порівняння реальних та «модельних» процесів модель називають ізоморфною, якщо вона повністю відповідає реальній системі. У

такому разі ізоморфна модель виявляється досить складною та незручною як для дослідження, так і для практичного застосування. Спрощення моделі може бути здійснено через зменшення числа параметрів процесу, їх поєднання, застосування аналогій з інших галузей знань.

В цьому випадку такі моделі називають гомоморфними. Для дослідження складних систем зазвичай використовують такі моделі [43]:

- графи систем чи іконографічні моделі;
- математичні моделі;
- фізичні моделі.

Перший тип використовують під час аналізу функціональних взаємозв'язків. Прикладом іконографічної моделі системи управління проєктами є структура сукупності контурів управління: завдання – відповідальні – виконавці – ресурси – фінанси – ризики. Така модель дає лише загальне уявлення про систему, іноді це ще називають «баченням проєкту». Більше повне уявлення про проєкт дадуть фізичні або математичні моделі.

При побудові фізичних моделей використовуються результати формального порівняння деяких компонентів, що характеризують процеси, що відбуваються в реальних системах та гомоморфних моделях, що відрізняються від оригіналу за своєю природою та пристроєм. Прикладом таких моделей в управлінні проєктами є модель «Опір» – рушійні сили проєкту [44], модель «Вітрильник» [45] та багато інших. В них гомоморфність присутня через використання подібних та інших за своєю природою процесів.

Найбільш повне уявлення про залежність управляючих сигналів системи від вхідних параметрів та процесів можна отримати через математичне моделювання процесів.

Воно може бути представлене в аналітичному вигляді, а також у графічній та табличній формах, у вигляді програмного забезпечення тощо. Математична модель стане ефективним засобом аналізу процесів управління, у разі можливості формалізації її поведінки для процесів, що цікавлять.

Через неформалізованість або складність формалізації деяких процесів управління проектами необхідно використовувати різні моделі. Усі вони умовно поділені на два напрямки. У першому створюють ефективну модель кожного завдання, далі – моделі їхньої сукупності, агрегуючи які отримують модель процесу в цілому.

До другого напрямку належать підходи, що полягають у моделюванні процесів окремих підсистем, наприклад: управління командою проекту, ризиками тощо. Обидва напрями містять підходи, що враховують властивості системи управління проектами як організаційно-технічної системи, включаючи процеси планування, реалізації та контролю [46].

Найчастіше при вирішенні поставленого завдання використовують метод імітаційного або напівнатурального моделювання, коли менеджери проектів програють різні варіанти траєкторії розвитку проекту. Проте, застосування обчислювальних потужностей, як і процедури логічного висновку, обмежується відкритістю «системи проект», що проявляється нелінійністю поведінки системи, нелінійною залежністю вихідних змінних управління від вхідних, управляючих впливів.

Тому у зв'язку з нелінійністю системи процедуру оптимізації функціонування системи управління командою проекту в дисертаційній роботі розглядати не будемо. Зупинимось на пошуку та обґрунтуванні моделі, яка дозволить забезпечити такі умови (значення параметрів процесу управління), які найбільш повно розкриють шукану властивість системи. Це дозволяє абстрагуватися від несуттєвих деталей, що ускладнюють рішення, і призводять до складності через багатокритеріальність.

Отже, оскільки ізоморфізм передбачає як тотожність структури системи, так і тотожність функціонування самої системи та її моделі, то побудова моделей складних систем дуже рідко призводить до повного ізоморфізму [47, 48].

В даному випадку систему управління ІТ-проектом можна віднести до дуже складної системи, спираючись на класифікацію Біра [43]. Тоді ступінь ізоморфності моделі реальної системи неможливо перевірити жодними відомими

способами. У цьому випадку аналізовану систему рекомендують спростити, застосувавши до неї перетворення, яке буде однозначним лише в один бік. Тому спрощення повинно проводитися за допомогою гомоморфної згортки, коректність якої повинна бути підтверджена.

В [45] показано, що «... повнота і точність відтворення зовнішнього світу у людській свідомості завжди відносні і ... неможливо визначити ізоморфними». Тому завдання гомоморфного перетворення моделі управління командою ІТ-проєкту буде зводитися до логічно несуперечливого перетворення всієї доступної інформації про систему управління ІТ-проєктом в ємну і компактну модель, використання якої буде зручним для прийняття проєктних рішень. Таким чином, задачу слабоформалізованого управління командою інноваційного ІТ-проєкту конкретизуємо через гомоморфне перетворення базових процесів дизайн-мислення. Логічно несуперечливим обґрунтуванням є подібність процесів управління командою процесам дизайн-мислення (табл.2.1).

*Таблиця 2.1.*

**Визначення подібності процесів**

Процеси управління командою ІТ-проєкту	Процеси дизайн-мислення
Формування контактів між членами команди та замовником, користувачами проєкту	Емпатія
Формування очікувань команди проєкту	Фокусування
Формування вимог членів команди проєкту до особистісного розвитку та зростанню	Генерація ідей
Реалізація заходів з формування креативної мінімально життєздатної команди проєкту	Прототипування
Аналіз здобутків у особистісному зростанні членів команди під час життєвого циклу проєкту	Тестування

Таким чином, за результатами аналізу процесів ІТ-проєктів, що подані у табл.2.1, можемо зробити висновок, що гомоморфна згортка у бік обмеження

параметрів [50], що характеризують ці процеси, може бути застосована задля розробки компактної семіотичної моделі управління командою ІТ-проектів.

## **2.4 Семіотична модель управління командою ІТ-проекту**

Семіотичне управління найчастіше використовують як узагальнене управління. "Узагальнення" необхідне при складних ситуаціях, коли має місце великий обсяг вихідних даних і велика кількість залежностей та зв'язків [51]. Прийняття рішень у складних ситуаціях потребує узагальнення та редукції, проте при цьому значні параметри об'єкту управління та поточної ситуації вважаються незмінними. Тому, у складних ситуаціях семіотичне управління є альтернативно-значимим рішенням [52].

Вважається, що семіотичні моделі — це моделі верхнього рівня абстракції по відношенню до ситуаційних моделей. Які, в свою чергу, є моделями верхнього рівня по відношенню до технологічних моделей, які зазвичай і використовують в управлінні проектами, спираючись на технологію розробки продукту проекту.

Такий «верхній» рівень семіотичних моделей значно скорочує час та зменшує складність аналізу проекту та логіку проектних рішень, а застосування семіотичного управління підвищує швидкість та якість управління проектами. У [52] показано, що семіотичне управління є ключовим при створенні систем управління складними ситуаціями, якими, безумовно, є управління командою ІТ-проекту з інноваційною складовою. В даному випадку складність проявляється у вигляді відсутності однозначної залежності результатів від вхідних параметрів.

Наприклад, якщо замовник (керівник ІТ-компанії/команди/проекту) хоче створити креативну команду ІТ-проекту, то як система управління реагуватиме на таку команду на вході? Можливо, йому спочатку необхідно просто створити команду, яка буде зацікавлена в управлінні та реалізації даного проекту. Однак, така інформація на вході відсутня, і система управління ухвалить неправильне рішення.

Ще однією проблемою є те, що креативна команда управління інноваційними ІТ-проєктами не є статичною системою. Протягом життєвого циклу команди можуть змінюватися як набір вхідних вимог (мається на увазі вимоги до компетентностей членів команди з боку проєктних задач, які вони мають виконувати), так і вихідні очікування самої команди та/або її керівника, або «служуючого лідера», як прийнято в дисертації. Це потребує зміни критеріїв управління впродовж ЖЦП.

В таких випадках при управлінні об'єктом з динамічною та унікальною структурою застосовують метод ситуаційного управління [53, 54]. Цей метод заснований на понятті ситуації, її класифікації та зміні.

Зазначимо спільність ситуаційного та семіотичного управління [51]. Обидва підходи спираються на використанні великої кількості правил, що описують поведінку об'єкта управління, в даному випадку команди управління як частини інноваційних ІТ-проєктів, а також описують множину правил, які пов'язують ситуації, пов'язані з об'єктом, і правило прийняття рішення.

Завданням управління в обох випадках є вибір правила із заданого переліку для використання у конкретній ситуації. При цьому модифікація, тобто зміна семіотичної моделі управління - допускається, тобто, правила змінюються на основі аналізу турбулентного оточення проєкту, очікування команди та результатів попереднього управління.

В ситуаційному управлінні сукупність відомостей про поточну структуру та стан об'єкта називають поточною ситуацією.

Крім того, вводять поняття повної ситуації *SCi (Complete situation)*, яка включає: поточну ситуацію *SRi (Current situation)*, знання про стан системи управління, а також знання про технологію управління.

І третьої значної сутністю системи управління називають обмежену множину можливих керуючих впливів на об'єкт, які називають однокрокові рішення *OSSi (One step solution)*.



Тоді дію управління, який переводить систему з поточної ситуації  $SR_i$  у поточну ситуацію  $SR_m$  з урахуванням повної ситуації  $SC_i$ , можна записати в такому вигляді [51]:

$$\{SC_i; SR_i\} \xrightarrow{OSS_i} SR_m \quad (2.1)$$

Наведене правило означає наступне: якщо поточна ситуація і повна ситуація в команді проєкту і в системі управління проєктом дозволяють застосувати якийсь однокроковий вплив, і він застосовується, то це призводить до появи нової поточної ситуації  $SR_m$ .

Можливість системи управління впливати на об'єкт визначається набором логіко-трансформаційних правил.

На рисунку 2.5 наведено структуру управління командою ІТ-проєкту на основі ситуаційної моделі.



**Рис. 2.5. Ситуаційна модель управління командою ІТ-проєкту**

Процес креативного управління командою ІТ-проєкту з урахуванням поданої ситуаційної моделі представимо у вигляді наступного алгоритму:

1. У блоці 1 відбувається формалізація поточної ситуації, в якій знаходиться команда ІТ-проєкту у теперішній час. Сформований опис поточної ситуації команди, як об'єкта управління, передається на блок 2.

2. В блоці 2 оцінюється необхідність втручання системи управління у процес. Якщо необхідність втручання існує, формалізований опис поточної ситуації передається у класифікатор (блок 3).

3. У блоці 3 виконується порівняння поточної ситуації з одним або декількома класами, яким відповідає одне однокрокове рішення.

Оскільки кількість можливих керуючих впливів на об'єкт завжди обмежено, а кількість існуючих повних ситуацій значно більша за кількість можливих рішень, то створюється класифікатор, який розділяє згадані повні ситуації на класи [52].

4. У розв'язувачі (блок 4) на основі логіко-трансформаційних правил, обирається правило, яке має використовуватися в даній ситуації. Якщо таке правило одне, то для об'єкту управління формується відповідний керуючий вплив (блок 7).

5. Якщо в розв'язувачі виявлено кілька можливих до застосування правил, то ці попередні рішення обробляються і серед них обирається найкраще (блок 5), на підставі якого формується керуючий вплив (крок 4).

6. Якщо ж в результаті проведеного аналізу, кращого рішення знайдено не було (тобто воно відсутнє для даної ситуації, блок 6), то обирається одне з рішень, яке відповідно до теорії ситуаційного управління [51, 56] надасть команді ІТ-проєкту, як об'єкту управління, незначний вплив (блок 7).

Або, через недостатність даних, система прийняття рішення відмовиться від будь-якого впливу. Тобто, у будь-якому випадку, система має рекомендувати рішення про ситуативний вплив на команду: ординарне рішення; найкраще рішення з можливих; незначне рішення із мінімальним впливом; рішення про відсутність дії.

Формальний опис моделі креативного управління командою ІТ-проєкту, або формальна модель, має ґрунтуватися на чотирьох множинах [55]\

$$Y = \langle BE, R, A, U \rangle \quad (2.2)$$

Множина *BE* формальної моделі містить усі базові елементи, які є основою інших елементів системи. В цій множині відсутні будь-які обмеження, при цьому кожен з елементів повинен відрізнятися від інших. Базовими елементами команди ІТ-проєкту можна вважати: імена членів команди; назви їх ролей; їхні компетентності і т.і.

Множина *R* — це множина синтаксичних правил. На їх основі формуються сукупності з базових елементів, які вважатимуться синтаксично правильними. На цю множину також не накладається спеціальних обмежень, за винятком цільового призначення: «чи є побудована на основі правил сукупність базових елементів синтаксично правильною». У термінах проєктної команди синтаксично правильними будуть вважатися всі поєднання, наприклад, імен членів команди та назв ролей, імен членів команди та їхніх особистих компетентностей тощо.

Множина аксіом, або множина *A*, утворює будь-яку множину синтаксично правильних сукупностей. У нашому випадку це можуть бути поєднання, наприклад, імені члена команди, назви його ролі та його компетентності або результати навчання, при цьому вони повинні утворювати повний стан системи. Однак, якщо після цього до нього додати, наприклад, номер телефону або будь-який інший ідентифікатор, отримана сукупність вже не буде вважатиметься аксіомою.

Множина *U* складається із семантичних правил, які розширюють множину аксіом, через додавання нових синтаксично правильних сукупностей.

У термінах проєктної команди будь-яке поєднання з іменем члена команди та його обов'язкових параметрів є семантичною правильною сукупністю. Однак, при додаванні будь-яких параметрів, які не належать об'єкту, створюється прецедент, в якому така сукупність вже не буде семантично правильною.

Однак, як було сказано вище, команда проєкту, як і сама система «Проєкт», є відкритою, нелінійною, динамічною. Для таких систем ефективніше використовувати семіотичні моделі, які беруть за основу, описану вище, формальну модель [53, 55].

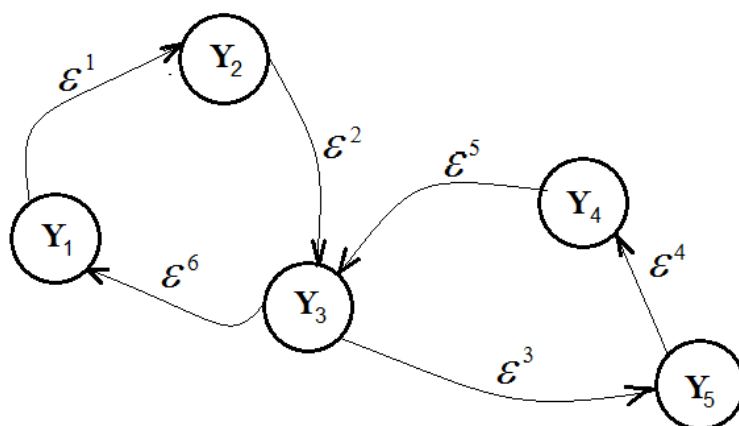
Суть семіотичної моделі полягає в тому, що для кожної з множин формальної моделі системи (2.2) застосовується набір правил, якими вона змінюється, тобто можуть змінюватись базові елементи, семантичні та синтаксичні правила, аксіоми. Семіотичну модель представимо у вигляді залежності (2.3):

$$S = \langle Y, \varepsilon^{BE}, \varepsilon^R, \varepsilon^A, \varepsilon^U \rangle, \quad (2.3)$$

де  $\varepsilon$  — правило зміни множин:  $BE, R, A, U$ .

В семіотичній моделі можливо змінювати прагматику базових елементів, їхню семантику та синтаксис.

На рисунку 2.6 показана семіотична модель управління командою ІТ-проєкту у вигляді мережі.



**Рис. 2.6. Семіотична модель управління командою ІТ-проєкту**

На рис. 2.6 показано, що модель містить в собі 5 варіантів формальної системи  $Y_i$  та правила зміни базових елементів з (2.2)  $\varepsilon^i$ :

$Y_1$  — команда формується;

$Y_2$  — команда визначає індивідуальні та групові цілі;

$Y_3$  — тестування команди;

$Y_4$  — команда формує технічні компетентності;

$Y_5$  — команда формує психологічні навички.

Але команда проєкту є динамічною системою і формальні системи можуть змінюватися. Наприклад, по завершенню етапу формування команди та адаптування один до одного, почнеться етап її становлення та набуття креативних компетентностей. Варіанти формальних систем можуть додаватися нові, а можуть і зникати попередні. Саме ці динамічні перетворення відображаються у змінах до правил  $\varepsilon^i$ . У ситуативній моделі правила не змінюються, а залишаються впродовж усього життєвого циклу проєкту.

З рисунку 2.6 видно, що за рахунок введення правил став можливий перехід від однієї формальної системи до іншої і це пов'язано із зміною  $\varepsilon^i$ , яке може збігатися з  $\varepsilon^{BE}$ ,  $\varepsilon^R$ ,  $\varepsilon^A$ ,  $\varepsilon^U$ , або з будь-яким їх поєднанням. Тобто, на відміну від знакової семіотичної системи у випадку складної системи, до типу яких належить проєкт, зміну правил прив'яжемо до поточного стану системи. Тоді зміна правил, наприклад, для базових елементів (повноти даних про членів команди проєкту) може мати такий вигляд  $\varepsilon^{BE}$ :

- 1) визначати лише роль за функціональним обов'язком;
- 2) визначати прізвище, ім'я, по батькові;
- 3) визначати прізвисько членів команди.

Другим прикладом аксіоми може бути визначення мети, яку ставлять перед собою претенденти до команди, а потім вже і члени команди  $\varepsilon^A$ :

- 1) збільшення заробітної плати та зменшення часового навантаження;
- 2) підвищення рівня власних технічних компетентностей;
- 3) підвищення рівня власної емпатії;
- 4) формування загальнокомандних цілей, тощо.

Як було зазначено вище, команда, як система, є динамічною. Будь-якого ідеального стану за досяжності якого робота над командою буде припинена - не існує [56], тільки закриття проєкту. Будь-які зміни, наприклад, додавання нового стану команди, потребує внесення змін у множину базових елементів  $\{BE\}$ . Тоді, набір правил  $\varepsilon^{BE}$ , по-перше, має дозволяти робити такі зміни, а подруге - має існувати процедура оновлення множини базових елементів.

Правила зміни аксіом має відображати поточний стан речей в системі управління для забезпечення повної ситуації. Наприклад, може скластися ситуація, коли до одного члена команди у конкретний момент може бути застосоване більш одного управлінського впливу: навчання для підвищення технічних компетентностей та психологічних (емпатичних) навичок. Така присутність декількох неузгоджених керуючих впливів призведе до небажаного результату.

Для моделі, що приведена на рис. 2.6, інтерпретація змін правил може бути наступною:

- $\varepsilon^1$  – система дозволяє визначати та збирати цілі команди;
- $\varepsilon^2$  – система дозволяє вносити зміни до тестів в залежності від встановлених цілей команди;
- $\varepsilon^3$  – система дозволяє вносити нові аксіоми задля формування психологічних навичок;
- $\varepsilon^4$  – система дозволяє вносити нові заходи із формування технічних компетентностей;
- $\varepsilon^5$  – правило обмеження одночасного впливу на суб'єкт управління (наприклад одночасного навчання за технічними та психологічними компетентностями);
- $\varepsilon^6$  – система дозволяє змінити загальну кількість членів команди.

## 2.5 Висновки до другого розділу

1. У другому розділі проведено аналіз ефективних методологій, на які буде спиратися усе подальше дослідження та розробка заходів з досягнення мети дослідження. У якості значущих прийняті клієнт-орієнтовні техніки гнучкого управління ІТ-проєктами: технологія дизайн-мислення, технологія створення мінімально-життєздатного продукту MVP, методика Proof of concept. Показано, що саме ці техніки є найбільш ефективним середовищем для розробки ІТ-проєктів із інноваційною складовою [57].

2. За результатами аналізу розроблено архітектуру дослідження, яку подано на рисунку 2.2.

3. В розділі показано, що головним критерієм та складовою успішного завершення ІТ-проєкту з інноваційною складовою є команда проєкту. Тому її стан та розвиток впродовж проєкту напряду впливатиме на показники успішності. За результатами аналізу властивостей подібних команд було запропоновано концептуальну модель креативного управління командами ІТ-проєктів з інноваційною складовою, яка подана на рисунку 2.4 у підрозділі 2.2.

4. В роботі запропоновано використовувати в якості інструментів управління та розвитку подібних команд наступні креативні інструменти: технологію дизайн-мислення, концепцію мінімально життєздатного продукту, методику PoC.

5. Оскільки інструментом наукового пошуку є моделі та методи, було проаналізовано, наскільки створювані моделі будуть адекватними оригіналу, тобто об'єкту дослідження, та чи правомірними будуть теоретичні висновки, щодо застосування їх на практиці. Саме для цього було показано, що моделі будуть будуватися за рахунок гомоморфного перетворення, через гомоморфну згортку. Таким чином, отримані у роботі гомоморфні моделі будуть адекватними при дослідженні процесів управління командою управління інноваційними ІТ-проєктами.

6. Розроблена у другому розділі модель управління командою ІТ-проєкту має властивість семіотичності, оскільки оточуючі процеси мають яскраво виражений ситуативний характер.

7. У підрозділі 2.4 подано розроблені: семіотичну модель управління командою ІТ-проєкту (рис. 2.6) та формальну ситуаційну модель (рис. 2.5), яка слугує їй базисом.

8. За результатами дослідження опубліковано 1 статтю [11], та 2 тези доповідей на конференціях [1, 38].

## Список використаних джерел до розділу 2

1. Близнюкова І.О., Данченко О.Б., Тесленко П.О. Підвищення рівня задоволеності замовника проєкту з використанням методології дизайн-мислення. *Project, Program, Portfolio Management. P3M-2020: Тези доповідей V Міжнародної науково-практичної конференції* : [у 2т.]. // Відповідальний за випуск П.О. Тесленко. Том 1. Одеса. : Балан В. О., 2020. С. 22–26. URL : [URL:https://drive.google.com/drive/u/2/folders/1ubmCVXUPHkEeIczv52D3NFPQL\\_14Kr31](https://drive.google.com/drive/u/2/folders/1ubmCVXUPHkEeIczv52D3NFPQL_14Kr31).
2. Pine II, B. J., Gilmore, J. H. Welcome to the Experience Economy. *Harvard Business Review*. URL: <https://hbr.org/1998/07/welcome-to-the-experience-economy>.
3. Krigsman. M. CIO Playbook: IT value and the digital mindset URL : <https://www.zdnet.com/article/cio-playbook-it-value-and-the-digital-mindset>.
4. Hasso Plattner Design Institute at Stanford University. URL : [dschool.stanford.edu](https://dschool.stanford.edu).
5. Україна 2030Е — країна з розвинутою цифровою економікою. Український інститут майбутнього. URL : <https://strategy.uifuture.org/kraina-z-rozvinutoyu-cifrovoyu-ekonomikoyu.html>.
6. Josh Plaskoff. Employee experience: the new human resource management approach. *Strategic HR Review*, 2017. ISSN: 1475-4398 URL : [https://www.emerald.com/insight/content/doi/10.1108/SHR-12-2016-0108/full/html?fullSc=1&utm\\_source=TrendMD&utm\\_medium=cpc&utm\\_campaign=Strategic\\_HR\\_Review\\_TrendMD\\_1&WT.mc\\_id=Emerald\\_TrendMD\\_1](https://www.emerald.com/insight/content/doi/10.1108/SHR-12-2016-0108/full/html?fullSc=1&utm_source=TrendMD&utm_medium=cpc&utm_campaign=Strategic_HR_Review_TrendMD_1&WT.mc_id=Emerald_TrendMD_1).
7. Ertel, C., Solomon, L. K. Moments of impact: how to design strategic conversations that accelerate change. New York: Simon & Schuster, 2014. 273 p.
8. Human-Centered Systems Thinking. URL : <https://www.ideo.com>.
9. Liedtka, J., Ogilvie, T. Designing for growth: A design thinking toolkit for managers. New York: Columbia University Press, 2011. 256 p.



10. Vasilieva, E. Developing the creative abilities and competencies of future digital professionals. *Automatic Documentation and Mathematical Linguistics*, 2018. No. 52 (5). P. 248–256.

11. Близнюкова І.О. Концепція створення мінімально життєздатного продукту та дизайн-мислення в управлінні командою ІТ-проєкту / І.О. Близнюкова, П.О. Тесленко, О.Б. Данченко, В.М. Меленчук. *Вісник національного технічного університету «ХПІ» : Зб.наук.пр. Серія : Стратегічне управління, управління портфелями програмами та проєктами*. Харків: НТУ «ХПІ», 2021. №2(4). С. 11 – 17. DOI: 10.20998/2413-3000.2021.4.2. URL: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/52329>.

12. A Guide to the Project Management Body of Knowledge (PMBOK® Guide). Sixth Edition. USA. PMI, 2017. 756 p.

13. Brown, T. (2008). Design Thinking. *Harvard Business Review*, 86, 84-92.

[https://www.researchgate.net/publication/5248069\\_Design\\_Thinking](https://www.researchgate.net/publication/5248069_Design_Thinking).

14. Jeanne Liedtka, Tim Ogilvie. Designing for Growth: A Design Thinking Tool Kit for Managers. *Columbia Business School Publishing*, 2011. 248 p. ISBN: 9780231158381.

15. Stinson L. IBM's Got a Plan to Bring Design Thinking to Big Business, 2016, Jan 21. URL : <https://www.wired.com/2016/01/ibms-got-a-plan-to-bring-design-thinking-to-big-business>.

16. Naiman L. Design Thinking as a Strategy for Innovation. *The European Business Review*, May 2019. URL: <https://www.europeanbusinessreview.com/design-thinking-as-a-strategy-forinnovation>.

17. Anh Nguyen Duc & Pekka Abrahamsson. Minimum Viable Product or Multiple Facet Product? The Role of MVP in Software Startups. *Lecture Notes in Business Information Processing*, 2016. URL: [https://link.springer.com/chapter/10.1007/978-3-319-33515-5\\_10](https://link.springer.com/chapter/10.1007/978-3-319-33515-5_10).

18. Данченко О.Б., Сепеда Гуаман Д.Ф. Інформаційна модель взаємодії стейкхолдерів організаційних проєктів у сфері обслуговування літаків. *Вісник*

Національного технічного університету «ХПІ». Серія: Стратегічне управління, управління портфелями, програмами та проєктами, 2019. №1 (1326). С. 24 – 29.

19. Журан О. А., Глава М. Г. Сучасний інструментарій управління портфелем проєктів. *Project, Program, Portfolio Management. РЗМ: Тези доповідей IV Міжнародної науково-практичної конференції* (06-07 грудня 2019 р.): [у 2 т.]. Т. 1. Одеса: Балан В.О., 2019. С. 32–34.

20. Dunets O., Wolff C., Sachenko A., Hladiy G., Dobrotvor I. Multi-agent System of IT Project Planning. *Proceedings of the 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications* (IDAACS'2017), Bucharest, Romania, 21-23 September, 2017. P. 548-552.

21. Шерстюк О.І., Тесленко П.О. Аналіз компетенцій команди проєкту при її взаємодії із зацікавленими сторонами. *Тези доповідей XVI міжнародної конференції "Управління проєктами у розвитку суспільства"*. К.: КНУБА, 2019. С. 248 – 249.

22. Teslenko P., Antoshchuk S., Krylov V. Increasing probability of successful projects complete. *Proceedings of the International Research Conference at the Dortmund University of Applied Sciences and Arts took place on June 30th - July 1st. 2017. Dortmund* : the Dortmund University. P. 28-30.

23. Lednor, P. W. (2019). *How To Be Innovative: Early-stage Innovation For Scientists, Technologists And Others-From Idea To Proof-of-concept*. World Scientific Publishing Company. 204 p. ISBN 9813222034.

24. Mary K. Pratt. Proof of concept (POC). URL : <https://searchcio.techtarget.com/definition/proof-of-concept-POC>.

25. Olha Sherstiuk, Oleksii Kolesnikov, Viktor Gogunskii, Kateryna Kolesnikova. Developing the Adaptive Knowledge Management in Context of Engineering Company Project Activities, *International Journal of Computing*, 2020, 19(4), 590-598, Scopus/. <https://doi.org/10.47839/ijc.19.4.1993>.

26. Doug De Carl. eXtreme Project Management: Using Leadership, Principles, and Tools to Deliver Value in the Face of Volatility, Jossey Bass, A Wiley Imprint, www.josseybass.com 2004, 516 p. ISBN 0-7879-7409-9.

27. Польшаков І.В. Креативний підхід до управління проектно-орієнтованими організаціями. *Управління розвитком складних систем*, 2016. № 26. С. 91-97. URL : <http://urss.knuba.edu.ua/files/zbirnyk-26/12.pdf>.

28. Бушуєв С.Д. Дорош М.С., Шакур Н.В. Інноваційне мислення при формуванні нових методологій управління проектами. *Управління розвитком складних систем*, 2016. №26 С. 49 – 57. URL: <https://repository.knuba.edu.ua/server/api/core/bitstreams/5aaa376b-3a97-45ea-9227-6e799dd5e9d8/content>.

29. K. K. Trier and T. Treffers, "Agile Project Management in Creative Industries: A systematic literature review and future research directions," *2021 IEEE Technology & Engineering Management Conference - Europe (TEMSCON-EUR)*, Dubrovnik, Croatia, 2021, pp. 1-8, doi: 10.1109/TEMSCON-EUR52034.2021.9488611.

30. Войтушенко А.А. Поняття креативного потенціалу у сфері управління проектами. *Управління розвитком складних систем*, 2019. №37. С. 13 – 17, [dx.doi.org\10.6084/m9.figshare.9783152](https://doi.org/10.6084/m9.figshare.9783152).

31. Ярошенко Ф.О., Бушуєв С.Д. Формування креативних моделей управління проектами розвитку системи державних фінансів України. *Фінанси України*, 2011. № (2). С. 4 – 10. URL: [http://www.irbis-nbuv.gov.ua/cgi-bin/irbis\\_nbuv/cgiirbis\\_64.exe?C21COM=2&I21DBN=UJRN&P21DBN=UJRN&IMAGE\\_FILE\\_DOWNLOAD=1&Image\\_file\\_name=PDF/Fu\\_2011\\_2\\_3.pdf](http://www.irbis-nbuv.gov.ua/cgi-bin/irbis_nbuv/cgiirbis_64.exe?C21COM=2&I21DBN=UJRN&P21DBN=UJRN&IMAGE_FILE_DOWNLOAD=1&Image_file_name=PDF/Fu_2011_2_3.pdf).

32. Харцій О.М. Розвиток креативного потенціалу у майбутніх менеджерів організацій: автореф. дис... канд. психол. наук: 19.00.10. «Організаційна психологія, економічна психологія». – К., 2008. – 22 с.

33. Кузьмінська Ю. М. Практичне застосування моделей та методів управління креативністю і ризиками команд освітніх проектів у сфері підвищення кваліфікації. *Вісник Черкаського державного технічного університету. Серія : технічні науки*, 2019. № 1. С. 23–32.

34. Woźniak, M., Boratyńska-Sala, A. The Potential of Creative Methods for IT Project Management. In: Nowak, R., Chrzyszcz, J., Brad, S. (eds) Systematic Innovation Partnerships with Artificial Intelligence and Information Technology. TFC 2022. *IFIP Advances in Information and Communication Technology*, 2022. Vol 655. Springer, Cham. [https://doi.org/10.1007/978-3-031-17288-5\\_21](https://doi.org/10.1007/978-3-031-17288-5_21).

35. Blankevoort PJ. Management of creativity. *International Journal of Project Management*, 1983. Volume 1. Issue 1. PP 33-36, ISSN 0263-7863, [https://doi.org/10.1016/0263-7863\(83\)90036-4](https://doi.org/10.1016/0263-7863(83)90036-4). URL: <https://www.sciencedirect.com/science/article/pii/0263786383900364>.

36. Данченко О.Б. Кузьмінська Ю.М. Креативний потенціал команди як фактор успіху проекту. *Управління проектами та Розвиток виробництва: Зб.наук.пр.* Луганськ: вид-во СЛУ ім. В.Даля, 2012. № 3 (43). С. 70-74. URL: <http://www.pmdp.org.ua/images/Journal/43/12debfup.pdf>.

37. Кузьмінська Ю. М., Данченко О. Б. Креативні технології управління освітніми проектами. *Innovative technologies in science and education. European experience*. Vienna, 2017. Vol. 1. P. 102–107.

38. Близнюкова І.О., Данченко О.Б., Тесленко П.О., Заруцький С.О. Концептуальна модель креативного управління командою ІТ проекту. *Project, Program, Portfolio Management. РЗМ-2021: Тези доповідей VI Міжнародної науково-практичної конференції*: [у 2т.]. // Відповідальний за випуск П.О. Тесленко. Том 1. Одеса. : ІШІР, 2021. С. 81–83. URL: <https://drive.google.com/drive/u/2/folders/1A19cjGj7O0NouNq9exxMQfd19AugkhLg>.

39. A Guide to the Project Management Body of Knowledge (PMBOK® Guide-Sixth Edition / Agile Practice Guide Bundle (HINDI). Project Management Institute. Publ., 2017. 115 p. URL: [www.PMI.org](http://www.PMI.org).

40. A Guide to the Project Management Body of Knowledge (PMBOK® Guide) and the Standard for Project Management. Seventh Edition. USA. PMI, 2021. 274 p.

41. Пепчук С.М., Пальонна Т. А., Коваленко Д. В. Залежність успіху розробки програмного забезпечення від якості командної роботи. *Вісник ЧДТУ*.

*Зб. наукових праць Черкаського державного технологічного університету. Серія: Економічні науки*, 2018. Вип. 48. С. 74-82. <https://doi.org/10.24025/2306-4420.0.48.2018.127039>.

42. Прокопенко Т.О., Ободовський Б.П. Дослідження впливу компетентностей членів проєктної команди на ефективність проєкту в галузі інформаційних технологій. *Вісник Національного технічного університету "ХПІ". Сер. : Стратегічне управління, управління портфелями, програмами та проєктами : зб. наук. пр.* Харків : НТУ "ХПІ", 2020. № 2. С. 50-55. URI :<https://repository.kpi.kharkov.ua/handle/KhPI-Press/46626>. DOI: 10.20998/2413-3000.2020.2.7.

43. Gofman A., Moskowitz H. Isomorphic permuted experimental designs and their application in conjoint analysis. *Journal of Sensory Studies*, 2010. Vol. 25., No. 1. pp. 127-145.

44. Бушуєв С.Д., Ярошенко Р.Ф. Методології управління проєктами на моделях класу «рушійні сили – опір». *Управління розвитком складних систем*, 2010. № 2. С.11-15

45. Teslenko P., Antoshchuk S., Krylov V. Increasing probability of successful projects complete. *Proceedings of the International Research Conference at the Dortmund University of Applied Sciences and Arts took place on June 30th -July 1st 2017 for the seventh time*, 2017. Dortmund : the Dortmund University. P. 28-30.

46. Teslenko P., Polshakov I., Bedrii D. Strategic management of evolving project-oriented organization. *Science and Education a New Dimension, Economics IV*. Budapest, 2016. Vol. (2). Issue: 94, p. 33-35. available at: [http://www.seanewdim.com/uploads/3/4/5/1/34511564/econ\\_iv\\_2\\_\\_94.pdf](http://www.seanewdim.com/uploads/3/4/5/1/34511564/econ_iv_2__94.pdf).

47. Томашевський В.М. Моделювання систем. К.: Видавнича група BHV, 2005. – 332 с.

48. Siegfried, R. Modeling and simulation of complex systems: A framework for efficient agent-based modeling and simulation. Springer, 2014.

49. Quin, Angus J. "Cybernetics in management." *Systems Science: Addressing Global Issues*. Boston, MA: Springer US, 1993. 199-204.

50. Kasai, T., & Miller, R. E. Homomorphisms between models of parallel computation. *Journal of Computer and System Sciences*, 1982. 25(3). pp. 285-331.

51. Jakobson, G., Buford, J., & Lewis, L. Situation management: Basic concepts and approaches. In *Information Fusion and Geographic Information Systems: Proceedings of the Third International Workshop*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, May. pp. 18-33.

52. Desouza, K. C., & Hensgen, T.(2005) *Managing information in complex organizations: semiotics and signals, complexity and chaos*. Published by Routledge, 264 p. ISBN 9780765613615.

53. Бондар О. В. Ситуаційний менеджмент. Навч. посіб. 2-ге вид., перероб та доповн. К.: Центр учбової літератури, 2012. 388 с.

54. Marakas, G. M. *Decision support systems in the 21st century*. Upper Saddle River: Prentice Hall, 2003. Vol. 134.

55. Joslyn, C., & Rocha, L. Towards semiotic agent-based models of socio-technical organizations. In *Proc. AI, Simulation and Planning in High Autonomy Systems (AIS 2000) Conference*, Tucson, Arizona, 2000. pp. 70-79.

56. Jason Gordon. Contingency Approach or Situational Approach to Management – Explained. URL:[https://thebusinessprofessor.com/en\\_US/management-leadership-organizational-behavior/contingency-approach-or-situational-approach-to-management](https://thebusinessprofessor.com/en_US/management-leadership-organizational-behavior/contingency-approach-or-situational-approach-to-management).

57. Бушуєва Н. Надихаюча інтуїція і креатив в управлінні інноваційними проектами /Бушуєва Н., Бушуєв С., Бушуєв Д., Бушуєва В. *Управління розвитком складних систем*. Київ: КНУБА, 2022. Вип. 49. С. 12-18. DOI: <https://doi.org/10.32347/2412-9933.2022.49.12-18>. URL: <http://mdcs.knuba.edu.ua/article/view/265111>.

## РОЗДІЛ 3

### МЕТОДИ КРЕАТИВНОГО УПРАВЛІННЯ КОМАНДАМИ ІТ-ПРОЄКТІВ

#### **3.1 Удосконалення методу формування креативної команди ІТ-проєкту з інноваційною складовою**

Стандартний підхід до формування команди проєкту містить у собі аналіз задач, які необхідно виконати для отримання продукту проєкту. Саме кількість цих задач, їх обсяг та тривалість визначають фаховий та кількісний склад команди проєкту. Крім того, методи формування проєктних команд різняться за галузевою направленістю проєктів [1].

Згідно з новітнім стандартом з управління проєктами [2], планування структури команди проєкту треба розпочинати з визначення навичок, рівня кваліфікації та досвіду роботи в аналогічних проєктах, які необхідні для виконання робіт проєкту. Слід зазначити, що попередня теза присутня й у новому стандарті, оскільки «.. визначення навичок для робіт проєкту, що є аналогічними попередньому досвіду», саме й означає, що необхідно провести аналіз завдань нового проєкту, визначити їх обсяг та складність.

Agile practice guide [3] визначає, що ефективні agile-команди, як правило, складаються з трьох-дев'яти осіб, які повинні розміщуватися в єдиному просторі і на 100% повинні бути віддані команді. Agile команда має бути самоврядною, її члени самостійно вирішують, хто і як виконуватиме роботу наступного періоду. Agile-команди замість керівника мають слугуючого лідера, який підтримує свою команду. Така команда обов'язково має бути крос-функціональною, за рахунок цього вона колективно володіє роботою і всі разом мають всі необхідні навички для завершення проєкту [4, 5]. Це формує синергетичний ефект, з допомогою якого можливий функціональний розвиток продукту.

Таким чином, Agile practice guide [3] визначає портрет ефективної команди, яка в змозі вирішувати інноваційні завдання. Метод формування команди ІТ-проєкту з інноваційною складовою має спиратися саме на портрет такої команди.

Тоді виникає питання, досягнення яких показників і якої величини, буде сигналізувати про завершення етапу формування команди? Відсутність зовнішнього жорсткого адміністративного управління в таких самокерованих командах, що самоорганізуються, виключає формування якихось планів та шляхів їхньої реалізації [5, 6]. Критерієм ефективності такої команди є успішне завершення проєкту, тобто ефективність гнучкої команди визначатиметься задоволеністю кінцевого споживача продуктом проєкту та задоволеністю самої команди від успішного завершення проєкту.

І хоча вимірність зазначеного критерію цілком можлива, але час його появи - це фініш проєкту, а команда формується на початку проєкту. Тоді вирішенням цього протиріччя, на думку автора, є формування команди за компетентнісним та психологічним критерієм.

У [7] стверджується, що у невеликих команд може не вистачити резерву для покриття усіх потреб проєкту за рахунок індивідуальних компетентностей, а великі команди можуть не досягнути стану самоорганізації. Але, зі збільшенням команди знижується і задоволеність власною роботою. Ці характеристики можуть бути враховані, як обмеження кількісного складу команди.

Стандарт з управління проєктами рекомендує такий перелік питань, які далі зможуть сформулювати вимоги та критерії для обрання кандидата до команди проєкту [2].

Перша вимога - це місце фізичного знаходження команди проєкту. Зазвичай кросфункціональні компетентності зможуть проявитися лише за умови, коли команда проєкту разом перебуває в одному місці. Друга вимога - це культурні погляди команди, культура, традиції, ментальність. Наступним є визначення, як здійснюється управління розвитком команди проєкту, за рахунок яких інструментів.

В результаті формування команда має набути наступних ознак [8]:

- відкриті комунікації всередині команди;
- спільне розуміння мети проєкту та здобутків від його успішного завершення;



- спільна відповідальність за кінцевий результат;
- довіра між членами команди;
- співпраця всередині команди, генерація ідей;
- адаптивність до середовища та ситуації;
- стійкість команди як можливість швидкого відновлення у разі виникнення проблем або збоїв;
- розширення прав, можливостей та повноважень кожного члена команди задля прийняття рішень щодо способу своєї роботи;
- отримання визнання та вдячності за виконану роботу, що збільшить імовірність продовжувати роботу з більшим натхненням та наснагою.

Зазначимо на протиріччя, яке формується об'єктивно, а саме за рахунок властивостей та особливостей ІТ-проектів з інноваційною складовою, що вимагають гнучких засобів управління та самокерованих команд, що самоорганізуються.

Протиріччя полягає у розподіленні у часі та місці точки прикладання управлінського впливу, та точки моніторингу очікуваної реакції від управлінського впливу. Тобто виникає управлінська колізія.

Тому формування команди ІТ-проекту з інноваційною складовою не закінчується на етапі ініціації, а продовжується аж до завершення проекту, тобто передачі замовнику продукту ІТ-проекту, та отримання його оцінки щодо задоволення власних очікувань.

Таким чином, команда, що формується, має відповідати таким ознакам. Вона має бути [8]:

- самокерованою (без зовнішнього централізованого керування);
- такою, самоорганізується;
- з феноменом групової емпатії;
- мінімально-життєздатною.

Зазначені ознаки при формуванні команди вимагатимуть реалізації не тільки індивідуальних критеріїв, які стосуються окремих членів команди, але й інтегральних або групових критеріїв, що стосуються команди цілком.

Такими критеріями є [8] :

- поведінкова модель команди;
- ступінь згуртованості команди;
- рівень взаємодії у команді;
- інтелектуальний капітал команди.

Поведінкова модель команди визначається поведінковими особливостями членів команди [9] та їх соціальними ролями [7, 9]. Ключову роль відіграє повнота поведінкової моделі та рівномірність розподілу ролей.

У якості підґрунтя МЖК [10] використаємо мінімально повну модель команди Raymond M. Belbin [11]: генератор ідей, критик, виконавець. За умови перевищення кількісного складу команди (тобто, у випадку, коли членів команди більше трьох), необхідно дотримання рівного співвідношення ролей кожної категорії: ролей дії, ролей аналізу та соціальних ролей.

Застосування техніки дизайн-мислення до управління інноваційними ІТ-проектами вимагає виділення додаткового типу ролей, а саме роль емпата. Усі члени команди, що формується, мають володіти здібностями емпатії у більшому чи меншому ступені. Але ця роль має бути додатковою до означених вище. Роль емпата за своїми властивостями входить до категорії «соціальних», але в [7] показано, що для успішного завершення проєкту бажана перевага ролей дії над ролями аналізу та соціальними ролями.

В разі непропорційного співвідношення ролей в команді для методу, що розробляється, бажано переважання соціальних ролей над ролями дії та аналізу. Таким чином, зазначене твердження сформулюємо у вигляді допущень для проєкту та для методу [8]:

- додавання ролі емпата у команду ІТ-проєкту з інноваційною складовою збільшить задоволеність замовника результатом проєкту, оскільки, як було показано в першому та другому розділі, такі проєкти характеризуються високим

рівнем невизначеності, конфліктності, гнучкості, інноваційності, і дана роль буде сугувати додатковим каналом ефективної взаємодії із замовником/користувачем проекту впродовж всього ЖЦП;

- більша кількість соціальних ролей в команді ніж ролей дії та аналізу збільшить ймовірність успішного завершення ІТ-проекту з інноваційною складовою, що також зумовлено особливостями таких проектів.

Згуртованість визначається привабливістю команди для кожного її члена, бажанням залишатися в її складі, визнанням кожного члена групи, при тому, що переважає співробітництво та групові комунікації та взаємна підтримка [12].

Вплив згуртованості на результативність команди має забезпечити самокерованість та самоорганізацію команди [13-17]. Тому метод формування команди проекту має розроблятися саме на основі згуртованості та зв'язків приязні між членами команди [12].

При цьому залишається відкритим питання про роль лідера, його контролю та посередництва [14]. Як було зазначено раніше, зовнішній адміністративний контроль та регулювання мають бути відсутніми для команд такого типу. При тому, на думку автора, роль слугуючого лідера, або, «координатора команди ІТ-проекту» (ККІТП) (цей термін було запропоновано у першому розділі (підрозділ 1.2) ще збільшується завдяки необхідності застосовувати та покращувати емпатійні властивості кожному з членів команди впродовж всього проекту. І, оскільки підвищення згуртованості команди проекту має призвести до підвищення ефективності роботи команди, то підвищення ролі лідера з контролю за взаємодії та розвитку емпатії має також призвести до підвищення ефективності роботи команди, і, врешті решт, до підвищення задоволеності замовника продуктом проекту.

Рівень взаємодії у команді буде визначатися соціальною поведінкою та соціальними зв'язками, також через групову інтеграцію в команді, через рівень сукупності зв'язків з іншими учасниками проекту, як форма соціального капіталу [12, 14].

У [18] показано, що попередній досвід взаємодії членів команди з іншими учасниками проекту (замовником / користувачем / підрядником / постачальником/ спонсором/ініціатором тощо) позитивно впливає на результативність команди. У зв'язку із поставленим критерієм мінімізації кількісного складу команди необхідно відстежувати доступність членів команди один для одного та для лідера, щоб мінімізація кількісного складу не призвела до небажаних зворотних процесів від самоорганізації та самокерованості.

Передумовою взаємодії є вірхний розподіл завдань у проєкті, що може вирішуватися через делегування. У цьому випадку знову треба розглянути роль лідера у команді у сенсі контролю ступеня делегування та завантаженості членів команди. Тому можемо стверджувати про необхідність підвищення доступності членів команди один для одного через підвищення значущості лідера.

Інтелектуальний капітал команди - це сумарний набір компетенцій, знань, умінь, навичок, здібностей команди [2, 13, 19-23]. Інтелектуальний капітал залежить від рівня освіти [24, 25], таланту, наявності обов'язкових та бажаних компетенцій, у тому числі: навчально-пізнавальних, для ІТ-проєктів – технічних: знання технічних аспектів, навичок програмування, комунікативних: уміння слухати, співпереживати, емпатувати, експертизи, швидкості та обсягу виконуваної роботи [16, 24, 25].

Оскільки професійні компетенції залежать від контексту проєкту, для аналізу інтелектуального капіталу ІТ-команд слід розглядати когнітивні здібності, як знання, набуті особистістю з власних пізнавальних процесів. Когнітивні здібності є значущими для інтелектуального капіталу учасника, оскільки мова йде про інтегральні крос-функціональні компетенції, які необхідні для самоорганізації та саморегуляції команди [26].

Оскільки сукупність моделей, методів та засобів, що розробляються в дисертації, спирається на емпатію, як властивість особистості, то інтелектуальний капітал команди слід розглядати як невід'ємний компонент властивостей команд, що застосовують техніки дизайн-мислення та МЖП для успішного завершення ІТ-проєкту.

В [27] показано, що інтелектуальний капітал експерта команди, або її лідера, суттєво впливає на результативність команди в цілому. Незважаючи на те, в контексті дисертації керівництво командою передбачається як колегіальне, з відсутністю централізованого керівництва (команда сама формалізує проблеми та знаходить шляхи їх подолання), керівник команди позиціонується як «служуючий лідер», або ККІТП, тим не менш, існує особа-експерт, яка на власний розсуд приймає рішення в умовах неповноти інформації. І тоді на перший план виходить людський фактор, який вносить суттєву непередбачуваність у поведінці складних нелінійних відкритих систем.

Реалізація засобів управління для таких систем залежить від кваліфікації та досвіду розробника (фахівця-експерта), оскільки більшість існуючих на сьогодні методів створення інноваційних ІТ-продуктів ґрунтуються на застосуванні експертних оцінок та рекомендацій з досвіду з минулого. Суб'єктивний фактор при цьому має істотний вплив при прийнятті помилкових рішень в процесі проектування, що призводить, безпосередньо, до погіршення показників проекту.

Отже, виникає науково-технічне протиріччя між наступними чинниками.

З одного боку, знайшли широке розповсюдження системи управління, що застосовуються для автоматизації широкого кола складних динамічних об'єктів та процесів в різних галузях сучасної економіки. Головною вимогою, що висувається до таких систем, є забезпечення заданих технологічних показників ефективності їх функціонування та простоти реалізації.

З іншого боку, відсутні універсальні методи та засоби, які дозволили б здійснювати створення та впровадження даних систем з усуненням або мінімізацією впливу рівня кваліфікації і досвіду фахівців-експертів та низки суб'єктивних чинників на процес проектування, а потім і на процес управління.

Наведене протиріччя може бути розв'язане шляхом зниження або усунення суб'єктивного впливу експертних знань та будь-яких евристичних факторів на процес проектування систем формування та управління командою ІТ-проекту для підвищення їх показників якості.

Для цього потрібно створити нові або удосконалити існуючі методи формування команди ІТ-проєкту для забезпечення досягнення заданих показників команди як складної динамічної самокерованої системи.

Задачу формування мінімально-життєздатної команди ІТ-проєкту [21] сформуємо через залежність між компетентностями проєктної команди за умови мінімізації кількості її членів.

Для команди ІТ-проєкту введемо такі позначення [8]:

– множина кандидатів до проєктної команди ІТ-проєкту

$$X = \{x_1, x_2, \dots, x_n\}, \quad (3.1)$$

де  $n$  – кількість претендентів в проєктну команду,

$$1 \leq n \leq N_{\text{обм}}, \quad (3.2)$$

де  $N_{\text{обм}}$  – максимальна кількість претендентів в команду проєктів, яких запрошують, обмежується керівником або замовником проєкту;

– множина компетентностей, якими має володіти кожен  $i$ -ий претендент

$$K_i = \{k_{i1}, k_{i2}, \dots, k_{iM_i}\}, \quad (3.3)$$

де  $i=(1,n)$ ,  $M_i$  – кількість компетенцій, якими володіє  $i$ -ий претендент.

Задачу формування команди ІТ-проєкту зведемо до задачі математичного програмування. Кожному претендентові з (3.1) можна поставити у відповідність певний вектор у багатовимірному просторі з координатами (3.3).

Кандидати заздалегідь будуть поділені на групи, що відповідають різним психологічним типам особи. Вектор (3.3) для кожного претендента можна отримати в результаті його експертного оцінювання.

Процедура створення МЖК актуалізує завдання оптимізації організаційної структури та методів управління командою ІТ-проєкту. Загальний підхід

заснований на формуванні команди, яка об'єднана однією метою, здатна досягати мети автономно і злагоджено за мінімальних управлінських впливів.

Тоді задачу формування МЖК сформулюємо наступним чином [8]:

$$\begin{cases} \sum_{i=1}^n x_i \rightarrow \min \\ \sum_{j=1}^n E(K_j) \rightarrow \max, \end{cases} \quad (3.4)$$

де  $x_i$  — приймає значення 1 (якщо претендента зарахували до складу команди) або 0 (якщо претендента не зарахували до складу команди),

$E(K_j)$  — ефективність команди від множини компетенцій  $K_j$ , яку можна оцінювати в балах або в кількості успішно виконаних задач згідно плану проекту (в разі застосування даної моделі не на етапі формування команди, а перегляду складу команди впродовж ЖЦП).

Причому

$$1 \leq \sum_{i=1}^n x_i \leq n \quad (3.5)$$

Тобто, задача формування МЖК сформована як набуття командою максимальної сумарної ефективності від компетенцій, якими володіють усі члени команди, за умови мінімізації кількості членів [28].

Підсумовуючи все сказане вище, тобто всі умови, передумови, обмеження, сформулюємо узагальнений метод формування мінімально-життєздатної команди ІТ-проекту з інноваційною складовою, яка застосовуватиме технології дизайн-мислення та МЖП у якості провідних технологій розробки програмного продукту [8]:

1. Як було зазначено в концептуальній моделі креативного управління командою проекту з інноваційною складовою (рис. 2.4), процедура управління командою ІТ-проекту розпочинається з аналізу проектних вимог, очікувань замовників та споживачів продукту проекту. Це дає змогу визначитися із

баченням контурів майбутнього продукту та визначитися з ролями, які мають бути присутніми в команді проєкту для забезпечення його успішного завершення.

2. Перелік ролей формується за напрямками та категоріями, що були окреслені вище, це такі категорії: професійні, емоційні, емпативні, Win-Win. Останні дві позиції є унікальними в межах методу, що розробляється в дисертації. Емпативні властивості особи мають забезпечити підґрунтя використання технології дизайн-мислення, як засобу управління ІТ-проєктом. Категорія Win-Win має забезпечити інтегральні властивості команди, коли особистісне задоволення кожного члена команди призведе до формування синергетичного командного ефекту задля успішного завершення проєкту.

3. На наступному етапі формуються компетентності, якими мають володіти члени команди задля успішного завершення проєкту. Якщо компетенції сформовані не повністю, або інформації недостатньо, слід повернутися до попередніх етапів (до п.1 методу) через більш детальну взаємодію із стейкхолдерами, задля розширення бачення проєкту.

4. Коли компетентності сформовані повністю, розпочинається процес формування переліку претендентів для участі в команді проєкту. Багато експертів зазначає, що самокерована команда можлива лише тоді, коли її члени мали попередній досвід роботи один з одним. Тобто, незнайомі до проєкту особи не зможуть утворити самокеровану команду, що самоорганізується.

5. Далі претенденти проходять систему тестувань, які відображають наступні категорії: професійні, емоційні, емпативні, Win-Win [21]. Зазначимо, що технічно форма тестів не буде розподілена на окремі частини.

6. Після тестування будуть обрані претенденти, які за відповідями можуть утворити бажану команду, для цього буде проведено перевірку щодо закриття ними усіх компетентностей, які забезпечать проєкту його успішне завершення. Якщо якась з компетентностей в результаті відбору залишилася не закритою обраними претендентами, повертаємось до п.4 методу, до формування нового переліку претендентів.



7. Якщо усі компетентності, які необхідні для реалізації проекту, закриті, починається процес мінімізації кількісного складу команди. Звернемо увагу на те, що критерій мінімально-можливої кількості за технологічними можливостями не є головним. Команда, що самоорганізується та самокерується, повинна мати час на: ділове спілкування (а не лише тільки кодування); обмін досвідом, поточними справами та проблемами; постійну взаємодію із стейкхолдерами; навчання та розвиток. Саме ці питання вирішують наступні кроки методу.

8. Спочатку аналізуються компетентності, які присутні у декількох претендентів. Якщо відповідь «Ні», то формування МЖК завершено.

9. Якщо відповідь «Так», то перевіряється можливість зменшення кількості командного складу, не втративши компетентностей, та можливостей для ділового спілкування. Якщо зменшення чисельності команди призведе до обмеження можливостей ділового спілкування в команді, то і приймається рішення про те, що МЖК створена.

10. Якщо можливо зменшити кількість членів команди (відповідь «Ні»), то прибираємо одного з претендентів і знову перевіряємо команду на повноту компетентностей та можливостей ділового спілкування (переходимо до п.7 методу).

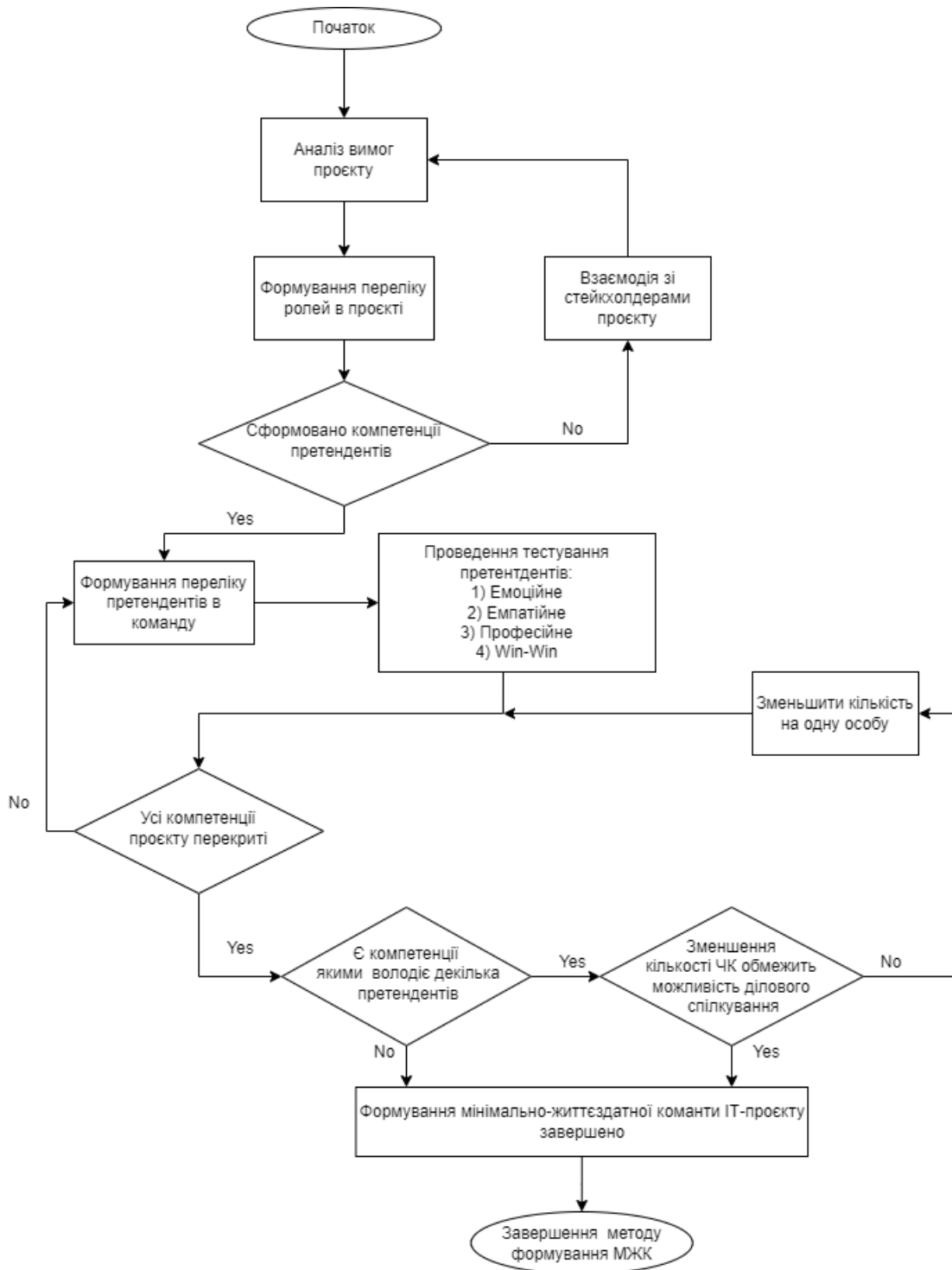
Блок-схема алгоритму методу наведена на рис. 3.1.

Таким чином, в роботі запропонований метод формування мінімально-життєздатної команди ІТ-проекту з інноваційною складовою [8]. Така команда буде самокерованою та такою, що самоорганізується, з властивостями емпатії до проблем замовника та стратегією Win-Win всередині команди.

### **3.2 Застосування генетичного алгоритму для мінімізації кількісного складу команди ІТ-проекту**

Розроблений в 3.1 метод формування команди ІТ-проекту з інноваційною складовою ставить за мету створення мінімально-життєздатної команди, яка за визначенням має забезпечити перелік необхідних компетентностей та навичок у

членів команди, при мінімальній кількості її членів, при цьому такий мінімальний склад не повинен обмежувати час на спілкування як всередині команди, так і зовні (із замовником, кінцевими споживачами продукту проєкту) [28].



**Рис. 3.1. Метод формування мінімально-життєздатної команди IT-проєкту [8]**

Ще одним обмеженням може виступати наявність фінансових ресурсів на оплату праці відповідних категорій працівників [21]. Таким чином, успіх реалізації проекту залежить не тільки від висококваліфікованої професійної проектної команди, але й від однозначного розуміння усіх процесів проекту його зацікавленими сторонами. Таке комплексне бачення має бути інтегроване в єдину систему термінів, понять, проектних дій, як це представлено у [30].

Як було зазначено в 3.1, в процесі управління командою проекту, як складною системою, має місце науково-технічне протиріччя між необхідністю застосування засобів автоматизації складних динамічних об'єктів та процесів та відсутністю універсальних методів та засобів, які б мінімізували вплив рівня кваліфікації і досвіду фахівців-експертів та низки інших суб'єктивних чинників на процес проектування, а потім і на процес управління [28]. Це протиріччя має бути розв'язане через зниження суб'єктивного впливу експертних знань на процес формування та управління командою ІТ-проекту. Попередня формалізація процесів формування команди виявила багатокритеріальність такої задачі, а функція управління матиме не один, а декілька екстремумів, які не можуть бути зведені до одного - інтегрального. Крім того, функція управління є складною та динамічною, тобто буде змінюватися в часі впродовж ЖЦП.

Метою цього підрозділу є обґрунтування та розробка методу мінімізації кількості членів команди, за рахунок динамічного планування виконання задач проекту, яке враховуватиме не лише характеристики самих задач, але й прогнози потреб у нових ресурсах та/або нових компетентностей ресурсів та вартості цих ресурсів із заданими компетентностями. Крім того, як було зазначено вище, обмеженням є обов'язкове виділення часу на спілкування/обговорення членами команди поточної ситуації в проекті, задля створення можливостей для прояву синергетичного ефекту та групової емпатії.

Мінімізація кількості членів команди проводиться за умови обмежень на крайній термін виконання кожного із завдань з урахуванням технології створення мінімально-життєздатного продукту та з урахуванням створення команди управління проектом, яка буде незмінною впродовж усього ЖЦП.

Метод формування МЖК базується на зменшенні кількості членів команди за рахунок розширення діапазону компетентностей кожного з учасників відбору. При цьому обмеженнями є:

- 1) умова обов'язкового володіння емпатією [29];
- 2) наявність додаткового часу (не пов'язаного з програмуванням): 2.1) на комунікації всередині команди та із замовником; 2.2) на тренінги/семінари для збільшення рівня емпатії та збільшення професійних компетенцій;
- 3) умова наявності попереднього досвіду спільної роботи та/або досвіду роботи в команді;
- 4) фінансові обмеження із боку замовника на оплату робіт проєкту.

Тоді для забезпечення всіх необхідних робіт проєкту доцільно вибрати невелику кількість типів ресурсів, які будуть задіяні в команді. Ці типи ресурсів мають бути обрані з урахуванням типів завдань, які потрібно вирішувати. Ми називатимемо ресурси претендентами, учасниками конкурсного відбору, до того, як вони будуть рекомендовані для участі в проєкті. Після рекомендації та згоди претендента на участь у команді проєкту, називатимемо його членом команди.

Далі процедура відбору буде базуватися на наступних припущеннях: чим більше номер типу претендента, тим менший його компетентнісний діапазон і менше його вартість, а також, якщо завдання може ефективно вирішуватися *i*-м претендентом, значить вона не гірше може бути виконана і претендентами з меншими номерами. Ще однією умовою роботи і навіть існування команди ІТ-проєкту є те, що кількісний склад не може сильно змінюватись в межах ЖЦП, особливо в бік зменшення. Мається на увазі, що після закінчення періоду завантаженості даного ресурсу, наприклад, протягом двох тижнів або місяців, він видаляється з команди.

При такому підході однозначно буде втрачено синергетичний ефект (або він і не встигне сформуватися в таких умовах) і вся перевага самоорганізованої та самоврядної мінімально-життєздатної команди буде втрачена [21].

Разом з тим, технологія створення мінімально-життєздатного продукту передбачає поступове розширення функціоналу продукту в його наступних

прототипах, що вимагатиме і збільшення ресурсного забезпечення.

Тому запропонований підхід враховує необхідність зміни чисельного складу команди, залежно від поточних умов та формування пропозицій щодо прийняття рішень, а самоорганізована та самоврядна команда на основі цих даних прийматиме остаточне рішення. Зважаючи на складність і багатокритеріальність розв'язуваної задачі, її пропонується вирішувати з використанням засобів інтелектуального аналізу [32], а саме в системі, яка побудована на генетичному алгоритмі [32, 34].

Генетичні алгоритми зазвичай - це алгоритми глобального пошуку. Вони проводять оптимізацію, спираючися на принципи еволюції та природнього відбору. В роботі був використаний генетичний алгоритм з бінарним кодуванням. Він використовує бінарні рядки як хромосоми - носії генетичної інформації, що представляють можливі рішення.

Цей тип генетичного алгоритму складається з таких основних етапів [35]:

1. Ініціалізація популяції. На цьому етапі створюється випадкова популяція бінарних рядків.
2. Оцінка. Кожен бінарний рядок (хромосома) оцінюється за допомогою функції пристосування, яка визначає якість рішення, що він представляє.
3. Селекція. Хромосоми обираються для відтворення на основі їхньої пристосованості.
4. Кросовер (схрещування). Пари обраних хромосом обмінюються бітами для створення нових хромосом.
5. Мутація. Випадкові біти у хромосомах змінюються для внесення випадкових змін.

Генетичний алгоритм з бінарним кодуванням використовується для розв'язання широкого спектру проблем оптимізації, включаючи задачі розміщення, планування, проектування та інших. Він є ефективним і гнучким інструментом, який може адаптуватися до різних типів проблем і обмежень.

Через свою відносну простоту і ефективність, генетичний алгоритм з бінарним кодуванням може бути реалізований за допомогою стандартних

бібліотек та інструментів. Наприклад, в розробленій системі було використано бібліотеку DEAP (Distributed Evolutionary Algorithms in Python) для реалізації генетичного алгоритму [36].

Команда проєкту, яка використовує для розробки технології створення МЖП і ДМ, має чотири можливі типи трудових ресурсів:

R1 – зарезервовані ресурси (Reserved Instances) – ресурси з постійними навантаженнями;

Цей чисельний склад є на постійній основі в команді, тому динамічно він змінюватися не може. При цьому можуть бути ситуації, коли ці ресурси не повністю задіяні в запланованих завданнях. В цьому випадку зарезервовані ресурси зможуть виконувати додаткові завдання.

Reserved Instances повинні забезпечувати наявність усіх технічних компетентностей, що вимагаються проєктними роботами, а також обов'язкове вміння емпатувати проблемам замовника, демонструвати психологічну сумісність та мати досвід/можливість роботи в команді.

Наступними за пріоритетом йдуть ресурси R2 із меншим набором технічних компетентностей. Однак до них, як і раніше, висуваються такі ж вимоги - вміння емпатувати проблемам замовника та психологічну сумісність. Досвід роботи в команді не є обов'язковим.

Ресурси цього мають пріоритети другого порядку, їх вартість менша від вартості ресурсів R1.

Ресурси третього типу (R3) повинні володіти технічними компетенціями мінімум двох різних типів та володіти здатністю емпатії. Відповідно цей тип ресурсів має ще нижчі пріоритети і вартість.

Найнижчий пріоритет призначимо ресурсам R4, які володіють лише однією технічною компетентністю з необхідного в проєкті діапазону.

Для динамічного управління командою на кожному етапі ЖЦП буде вказано:

- тривалість виконання завдання у дискретах часу, що відповідають інтервалу часу доступності ресурсу;

- мінімальний набір компетентностей, який відповідає цьому завданню;
- крайній термін завершення виконання поточного завдання;
- наявність фінансових можливостей у замовника для покриття витрат на роботу сформованої мінімально життєздатної креативної команди проєкту.

Відповідно до цього, кожному ресурсу при запуску завдання формується вектор стану команди. Поточний час приймемо за  $\Delta$ ,  $(t+1)$  – тривалість виконання завдання,  $(\Delta+t+1)$  – крайній термін завершення завдання [33, 37].

Початок вектора стану відповідає етапу чи задачі, які виконуються першими. Якщо під час  $\Delta$  це завдання чи етап (які можуть відповідати, наприклад, одному спринту) було виконано, то вектор стану зменшується на один елемент. Якщо цей етап за час  $\Delta$  не виконаний, то всі одиниці зсуваються праворуч і вектор коротшає на один елемент зліва.

Структурно, система управління міститиме блок прогнозування, який визначатиме найближчим часом чисельно-компетентнісний склад проєктної команди.

Якщо в момент часу  $\Delta$  блок прогнозування видав прогноз щодо появи завдання, виконання якого можливо наявними в команді ресурсами, то всі одиниці у векторі стану зсуваються праворуч на інтервали часу, коли прогнозується виконання відповідних етапів цими ресурсами. Осередки ліворуч заповнюються нулями, що означає, що завдання очікує виконання зазначеними вище ресурсами. При скасуванні прогнозу стан вектора відновлюється.

Іншою можливою ситуацією вектора станів може бути склад із одних одиниць, що означає відсутність вільних членів команди. У цьому випадку відбувається пошук претендента за нижчою категорією [38].

Для формування команди з претендентів ресурсів істотним моментом є рівень запропонованої замовником ціни за необхідний вид ресурсу.

Тоді, власне, процедура формування мінімально-життєздатної команди алгоритмічно відповідатиме мінімізації цільової функції, яка розраховує необхідну кількість членів команди за необхідними компетентностями від поточного моменту часу на  $D$  дискретів вперед, які обмежуються ЖЦП.

$$K(\Delta) = K_1(\Delta) + K_2(\Delta) + K_3(\Delta) + K_4(\Delta) \quad (3.6)$$

де  $K(\Delta)$  – сумарна кількість членів команди, що покривають усі необхідні компетенції та вимоги на дискрет часу  $\Delta$ ;

$K_1(\Delta)$  – сумарна кількість членів команди, які відповідають типу ресурсів R1 на дискрет часу  $\Delta$ ;

$K_2(\Delta)$  – сумарна кількість членів команди, які відповідають типу ресурсів R2 на дискрет часу  $\Delta$ ;

$K_3(\Delta)$  – сумарна кількість членів команди, які відповідають типу ресурсів R3 на дискрет часу  $\Delta$ ;

$K_4(\Delta)$  – сумарна кількість членів команди, які відповідають типу ресурсів R4 на дискрет часу  $\Delta$ ;

Сумарна кількість членів команди типу ресурсу  $R_i$  на дискрет часу  $\Delta$  залежатиме від таких складових:

$$K_i(\Delta) = \sum_{n=1}^N \sum_{m=1}^M s_m^i(\Delta) \times d_{nm}^i(\Delta) \times T_{nm}^i(\Delta), \quad (3.7)$$

де  $i = [1, 4]$  - номер типу ресурсу;

$N$  – кількість робіт проєкту;

$M$  – кількість претендентів у команду проєкту;

$s_m^i(\Delta) = [1, 0]$  – фінансова доступність забезпечення  $i$ -ого типу ресурсу в дискрет часу  $\Delta$  (1 – ресурс доступний, 0 - недоступний);

$d_{nm}^i(\Delta) = [1, 0]$  – плановане використання  $m$ -ого претендента в  $n$ -й задачі в дискрет часу  $\Delta$  (1 – ресурс запланований для використання, 0 - незапланований);

$T_{nm}^i(\Delta) = [1, 0]$  – планована затримка реалізації  $n$ -ї задачі з  $m$ -м претендентом в дискрет часу  $\Delta$  (1 – присутня планова затримка завдання, 0 - відсутня).



Мінімізація чисельного складу команди проєкту проводиться кожен поточний дискрет часу  $\Delta$  за рахунок вибору відповідних значень коефіцієнтів  $K_i(\Delta)$  генетичним алгоритмом при зазначених вище обмеженнях [36].

Довжина хромосоми для кожного розрахунку визначається кількістю завдань та кількістю відповідних типів ресурсів [33, 39]. Кількість хромосом, що піддавалися мутаціям, а також коефіцієнт мутацій визначається при проведенні експериментів, результати якого будуть представлені в четвертому розділі.

В результаті запропонований метод формування МЖК ґрунтується на генетичному алгоритмі [33] зменшення чисельного складу команди з повним забезпеченням компетентнісного набору ресурсів для успішного завершення проєкту. Процедура враховує динаміку завдань, що запускаються [39], а також зміну компетентностей і навичок членами команди в межах ЖЦП.

### **3.3 Метод креативного управління командою ІТ-проєкту з інноваційною складовою**

Нарощування інноваційної діяльності на основі творчого управління командами дозволяє організаціям залишатися конкурентоспроможними і адаптуватися до ринкових вимог, що швидко змінюються. В той же час, творчі та інноваційні процеси складні та залежать від індивідуальних та групових зусиль, що визначає складність роботи самих команд. У своєму розвитку команда проходить низку етапів, кожен із яких характеризує особливості її функціонування. Спільним для всіх етапів є питання ефективності команди. На думку Johnsson (2014), щойно сформована творча команда навряд зможе виконувати складну інноваційну роботу без належного налаштування [108].

Метод креативного управління командою ІТ-проєкту з інноваційною складовою пропонується реалізувати, спираючись на наступні обмеження:

1. Формується перелік показників, яких має досягнути креативна команда ІТ-проєкту.
2. Впродовж проєкту проводиться вимірювання зазначених показників.

3. Проводиться порівняння поточних показників з еталонними/запланованими.

4. За результатами порівняння формуються управлінські впливи, які мають призвести до досягнення або наближення до еталонних значень.

Креативність учасників команди проєкту дозволяє їм координувати свої дії та знаходити найкраще рішення у процесі розробки продукту. Цим вимогам відповідає процес дизайн-мислення (Re-Enter Design Thinking), орієнтований на швидке створення прототипів, безперервний зворотний зв'язок та гіперітеративну розробку, що дозволяє всім членам команди вкладати особистісні ресурси та координувати свої дії у процесі розробки нових ітерацій рішення [10].

Для моделювання впливу емпатії на процес творчого управління командою у роботі використано модель емоційного регулювання Дж. Гросса [41].

Сутність креативного управління командою ІТ-проєкту з інноваційною складовою будемо розглядати наступним чином.

У якості початкових умов вирішення задачі управління будемо вважати, що команда сформована за методом формування мінімально-життєздатної команди ІТ-проєкту (із застосуванням концепції Win-Win), яка володіє властивостями емпатії, який запропонований в підрозділі 3.1 дисертаційного дослідження.

Як було зазначено у методі формування мінімально-життєздатної команди [42], в алгоритмі формування присутня процедура вибору, коли відповідальна особа/експерт ІТ-проєкту приймає рішення щодо обрання кандидата, між критеріями його професійних навичок, та навичок володіння емпатією та вмінням працювати в команді проєкту.

Більшість дослідників креативного управління проєктами відзначає складнощі, пов'язані з реалізацією творчого підходу в керованому середовищі. За основні чинники інноваційного управління проєктами приймають платформу, процес та людей [40, 43].

Платформу відносять до цифрового каналу, який використовується для управління та реалізації творчих проєктів, та робить проєкт зрозумілим для всіх. Іншими словами, це система комунікацій в проєкті.

Будемо вважати, що на час завершення створення мінімально-життєздатної команди ІТ-проєкту вона буде відповідати всім вимогам та ресурсо-фінансового забезпечення, що були доступні проєкту під час процесу формування команди. Тобто команда була «сформована правильно».

Технологію ДМ відносять до гнучких [21] і на неї правомірно поширюються принципи Agile [43], які спрямовані на вирішення двох основних завдань: а) скоротити час виведення продуктів на ринок; б) прискорити час прийняття рішень на рівні проєктних команд.

Тому, подальший розвиток команди проєкту має буде обумовлений цими задачами та проводитися за напрямками:

1) розвиток емпатійних навичок та вмінь застосовувати їх до роботи у проєкті;

2) формування групової емпатії;

3) розвиток та вдосконалення професійних навичок задля вирішення задач проєкту;

4) формування позитивного мікроклімату в команді та емоційної культури, яка забезпечує різноманітність та свідомість емоційного реагування, розвиток емпатії як основи для цілеспрямованого використання емоцій у професійній діяльності та спілкуванні; спілкування сприймається як важлива частина управління проєктами, креативність сприяє підвищенню його ефективності. Якість процесу прийняття рішень у команді безпосередньо пов'язана з індивідуальною творчістю учасників та кліматом у групі;

5) досягнення індивідуальних цілей для кожного члена команди впродовж проєкту;

6) досягнення загально-командних цілей впродовж проєкту.

Міжособистісні відносини ідентифікують командне емоційне середовище і надають інформацію про події, що відбуваються, і визначають найбільш типові функції, які виконує емпатія в процесі управління командою:

- сприяє створенню взаєморозуміння у команді;

- зміцнює довіру учасників команди один до одного та до інших учасників проєкту;
- створює безпечний простір, в якому команда може вийти зі своєї зони комфорту та піти на ризик;
- призводить до більшої кількості інновацій та творчості;
- заохочує доброту та співчуття, що сприяє командній роботі;
- знижує рівень стресу за рахунок зведення до мінімуму конфліктів;
- знижує рівень робочого вигорання з урахуванням збільшення творчої складової.

Успішне функціонування інноваційного проєкту визначається наявністю творчого потенціалу команди, який підтримується обміном ідеями, знаннями та наявністю комунікацій.

Застосування технології «мінімальної життєздатності» означає, що команда створена без «запасу», або без «резерву», тому впродовж життєвого циклу проєкту може стати потреба у збільшенні кількості членів команди. У цьому випадку, при додаванні нових співробітників, необхідно повторно застосовувати методи формування, підбору, адаптації, формування загальних/індивідуальних цілей в команді. Можливо, за необхідністю збільшення кількості виконавців проєкту та/або низьким рівнем емпатії у них, буде доцільним виділити окрему команду Управління проєктом, яка буде володіти максимально можливим набором компетенцій та працювати на постійній основі.

Зазначимо, що описаний розвиток команди буде актуальним впродовж усього життєвого циклу проєкту, а це означає, що процеси управління не повинні припинятися та мають тривати до завершення проєкту.

Таким чином, структура системи управління командою має бути чотирьох канальною: 1) канал професійного зростання; 2) канал розвинення емпатії (групової емпатії); 3) канал розвитку емоційного інтелекту через застосування емоційного командного менеджменту; 4) канал досягнення спільної командної мети, яка уключає до себе індивідуальні цілі кожного члена команди. Ці канали

управління слід застосовувати паралельно, незалежно один від одного в сенсі аналізу поточних показників. З точки зору управлінського впливу, канали мають бути синхронізовані, задля формування одночасно-узгодженого впливу на суб'єкти управління (якими є члени команди проєкту) декількох факторів, щоб отримати синергетичний ефект.

Зазначимо, що вибір кандидатів здійснювався не за показником найбільшого агрегованого фактору, як пропонує більшість дослідників, а з урахуванням вагових коефіцієнтів, що визначатимуть пріоритет для конкретного проєкту, або конкретного часового етапу проєкту. Тому апріорі в команді будуть поточні компетенції, що не будуть максимально необхідними для конкретного члена команди (а для конкретного проєкту).

Цей «недолік» пропонується усунути за рахунок:

1) формування синергетичного ефекту командної взаємодії, який і покриє недостатні показники;

2) управління розвитком команди через розвиток емоційної культури, у тому числі, сприяння неформальним відносинам задля згуртування команди, як елементу підвищення мотивації. Емоційний менеджмент може бути представлений як процес управління емоційними ресурсами проєкту з підвищення ефективності його діяльності. Він передбачає визнання того факту, що у людей в процесі роботи виявляються емоції та виникають емоційні відносини, які формують енергетичний, ціннісний, інформаційний та мотивуючий ресурси емоційного організаційного середовища та визначають це середовище як невід'ємну складову організаційної діяльності;

3) окремо зазначимо, що «нестача» емпатії у окремих членів команди має бути нівельована та/або компенсована за рахунок формування феномену групової емпатії в команді.

Ще однією важливою властивістю емпатії є те, що вона сприяє встановленню толерантних відносин, що особливо важливо на початкових стадіях управління командою проєкту, коли рольове функціонування ще недостатньо узгоджено. Готовність до толерантного ставлення до висловлювань інших

незалежно від ступеня розбіжності зі своїми поглядами, здатність узгоджувати позиції і досягати консенсусу незалежно від початкової розбіжності точок зору та їх зіткнення характеризують поведінкову толерантність.

У керівництві зведення знань з управління проєктами [2] відзначається значущість спілкування для команди проєкту: «ефективні менеджери проєктів набувають балансу технічних, міжособистісних і концептуальних навичок, які допомагають їм аналізувати ситуацію та взаємодіяти належним чином» [40]. Як фактор, що покращує застосування навичок міжособистісного спілкування та сприяє виробленню більш продуктивних і цінних рішень, позначається креативність.

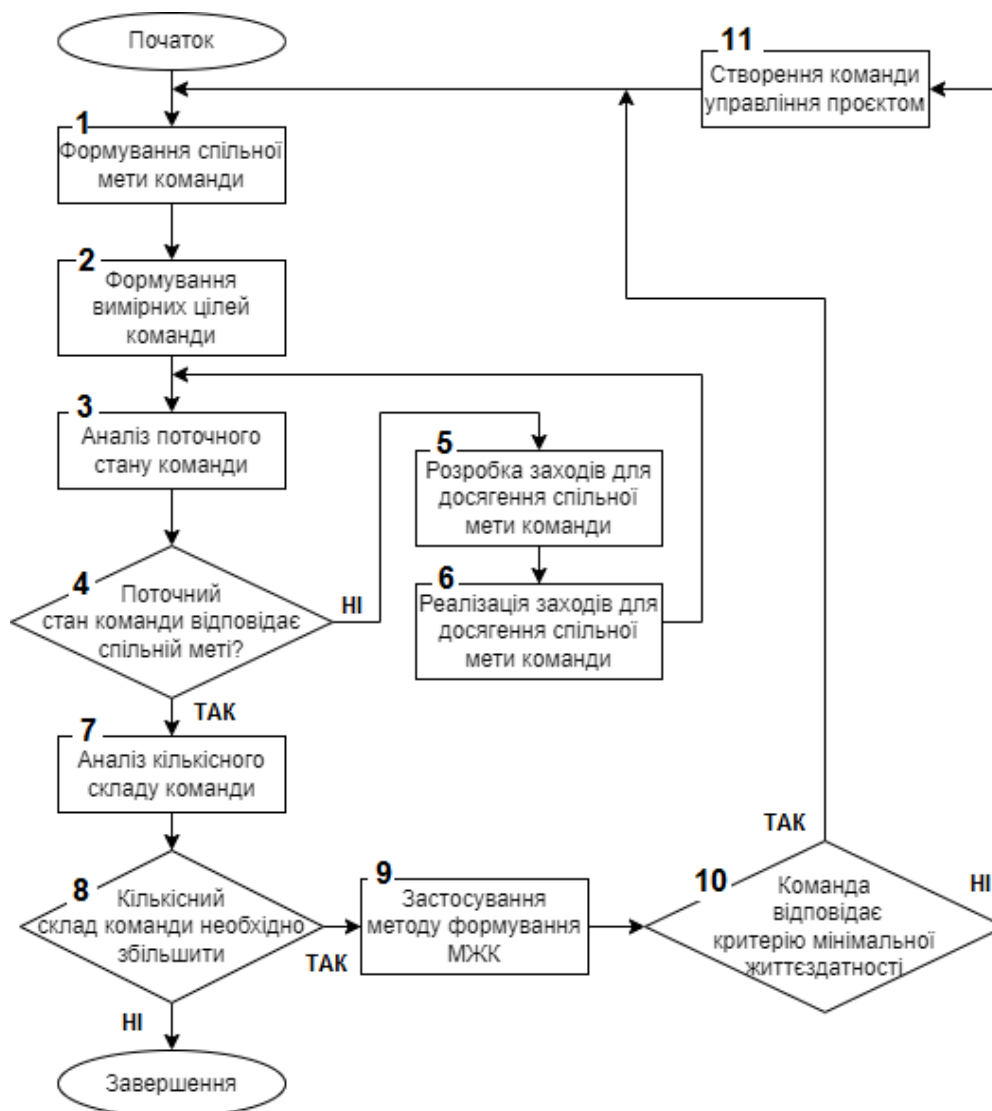
Ефективна команда проєкту характеризується наявністю неформальної атмосфери, умінням членів команди висловлювати свої ідеї та почуття та прислухатися один до одного, переводити конфлікти у конструктивне русло та приймати узгоджені рішення. Ефективності роботи творчої команди сприяє розподіл ролей між учасниками, для чого використовується класичний підхід за Р.М. Белбіном.

Все вище викладене сформулюємо у вигляді методу креативного управління командою ІТ-проєкту з інноваційною складовою, блок-схема алгоритму якого показана на рис.3.2:

1. Вважатимемо, що дія методу формування МЖК завершено (підрозділ 3.1) і мінімально-життєздатна команда створена.

2. Для цієї команди формулюється загальнокомандна мета, блок 1. Вона містить у собі очікування та побажання кожного окремого члена команди для особистого зростання та досягнення особистісних цілей. «Особистісний канал» на блок-схемі (рис. 3.2) не вказано. Вважаємо, що він входить як складова, у досягнення загальнокомандної мети. Це зроблено для того, щоб не ускладнювати рисунок.

3. Після встановлення та колегіального затвердження загальнокомандної мети формулюються вимірні показники, за якими буде перевірятися стан команди та розроблятися заходи задля досягнення мети, блок 2.



**Рис. 3.2. Метод креативного управління командою ІТ-проекту з інноваційною складовою**

4. Наступний крок (блок 3) якраз і передбачає перевірку поточного стану команди з урахуванням кроку 3. В даному блоці проводиться аналіз поточного стану команди, який включатиме в себе описані вище 4 канали. Для кожного напрямку будуть обрані стандартні спеціалізовані засоби. В більшості своїй вони будуть провадитися у вигляді невеликих тестів, в електронному вигляді. Ці тести члени команди будуть проходити на власному робочому місці, витрачаючи не більше 15-20 хвилин.

5. За результатами аналізу виконується перевірка відповідності поточного стану еталонним показникам (блок 4), у якості яких прийняті вимірні показники, які були отримані на кроці 2.

6. Якщо показники поточного стану не відповідають еталонним, то розробляються (блок 5), а потім реалізуються заходи (блок 6) задля досягнення вимірних загальнокомандних (та особистісних) показників, тобто для досягнення загальнокомандної мети.

7. Після реалізації зазначених в кроці 6 заходів проводять перевірку їх ефективності стосовно досягнення чи недосягнення вимірних загально командних цілей. Для цього повертаємось до блоку 3.

8. Якщо поточний стан команди відповідає заявленим цілям, то переходимо до блоку 7, в якому проводиться аналіз кількісного складу команди ІТ-проєкту. Як зазначалося раніше, з розширенням функціоналу продукту проєкту у кожному наступному прототипі може з'явитися потреба у додаткових ресурсах/виконавцях робіт.

9. Якщо в результаті перевірки (блок 8) кількість членів команди треба збільшити, то застосовується метод формування МЖК (блок 9).

10. Якщо команда у доданому складі відповідає критеріям мінімальної життєздатності (блок 10), то переходимо на початок (до блоку 1), з оновленням загальнокомандної мети, яка вже буде враховувати бачення нових членів.

11. Якщо команда у доданому складі не відповідає критеріям мінімальної життєздатності, тоді пропонується створити окрему команду управління проєктом [102] (блок 11), яка буде функціонувати на постійній основі, володіти властивостями емпатії. Після цього повертаємось на початок, до блоку 1, для оновлення загальнокомандної мети.

12. Завершення роботи методу відбувається після блоку 8, якщо буде прийняте рішення, що збільшувати кількісний склад команди непотрібно.

Зазначимо, що метод може бути запущений на будь-якій фазі ЖЦП. З іншого боку, постійне його застосування непотрібне, бо зміна емоційного стану команди, емоційної культури та емоційного менеджменту, емпатії, не



відбувається одразу після навчання, проходження тренінгу або семінару. Ці процеси відбуваються поступово, але тим не менш, вони мають бути керованими.

Тому оптимальним на думку автора проведення тестів, тобто перевірка результатів після навчання, має проводитися не частіше ніж раз за 2-3 тижня. Разом з тим, аналіз професійних навичок може проводитися одразу після навчання.

Саме тому, цю процедуру слід формалізувати та сформувати у вигляді інформаційної технології, яка буде представлена у четвертому розділі.

### **3.4 Висновки до третього розділу**

1. У третьому розділі розроблено методи управління командою ІТ-проєкту з інноваційною складовою, які спираються на технологію дизайн-мислення та концепцію створення МЖП. Потреба у вдосконаленні та розробки нових методів з'явилася тому, що команди, які потрібні для розробки проєктів з інноваційною складовою, мають бути креативними, самокерованими та саморганізованими. Це суттєво відрізняє їх від класичних методів формування та управління командами ІТ-проєктів, в яких головними чинниками та критеріями успішності були: спланованість, керованість, підпорядкованість. А головними критеріями відбору були виключно професійні здібності претендентів у члени команди.

Таким чином, нові реалії та нові вимоги до команди ІТ-проєкту потребують розробки нових та вдосконалення існуючих методів формування та управління креативними командами.

2. В підрозділі 3.1 вдосконалено метод формування команди ІТ-проєкту з інноваційною складовою, який в процесі відбору та формування команди спирається на креативні здібності претендентів. Крім того, запропонована процедура формування мінімально-життєздатної команди, в якій будуть сформовані сприятливі умови для розвитку процесів самокерованості та самоорганізації.

3. В підрозділі 3.2 запропоновано та обґрунтовано застосування генетичного алгоритму в процесах формування МЖК. В підрозділі показано, що задача формування є багатокритеріальною, а в рішенні відсутній головний критерій, який зазвичай вважається критерієм успішності проєкту. Тому, особа що приймає рішення, має орієнтуватися на власний досвід, уміння та компетенції.

Саме для виключення суб'єктивного впливу експерта, що має приймати рішення, і було запропоновано використання генетичного алгоритму.

4. В підрозділі 3.3 було розроблено метод креативного управління командою ІТ-проєкту з інноваційною складовою, який включає в себе продукування професійного зростання членів команди; розвинення їх емпатії (групової емпатії); розвиток емоційного інтелекту через застосування емоційного командного менеджменту; та досягнення спільної командної мети, яка включає в себе індивідуальні цілі кожного члена команди. Особливістю методу є те, що зазначені складові стануть не лише здобутками членів команди, а й єдино можливим важелем у забезпеченні успішного завершення ІТ-проєктів з інноваційною складовою.

5. За результатами дослідження опубліковано 5 статей [5, 8, 10, 28, 42] та 5 тез доповідей на конференціях [4, 6, 14, 21, 31].

### Список використаних джерел до розділу 3

1. Кузьмінська Ю.М., Данченко О.Б., Бедрій Д.І. Метод формування ефективних команд освітніх проєктів. *Вісник Національного технічного університету «ХПІ»*. Серія: Стратегічне управління, управління портфелями, програмами та проєктами: зб. наук. пр. Харків: НТУ «ХПІ», 2021. № 2(4). С. 46-53. DOI: <https://doi.org/10.20998/2413-3000.2021.4.6>. URL: <http://pm.khpi.edu.ua/article/view/229473>.
2. A Guide to the Project Management Body of Knowledge (PMBOK® Guide) and the Standard for Project Management. Seventh Edition. USA. PMI, 2021. 274 p.
3. A Guide to the Project Management Body of Knowledge (PMBOK® Guide-Sixth Edition / Agile Practice Guide Bundle (HINDI). Project Management Institute. Publ., 2017. URL: [www.PMI.org](http://www.PMI.org). 115 p.
4. Близнюкова І.О., Данченко О.Б., Тесленко П.О. Аналіз сучасних визначень ІТ-проєктів. Управління проєктами у розвитку суспільства. Тема: «Управління проєктами в умовах пандемії COVID-19»: тези доповідей. Київ : КНУБА, 2021. С.100-104. URL:<http://eprints.kname.edu.ua/58522/1/%D0%A2%D0%B5%D0%B7%D0%B82021.pdf>.
5. Близнюкова І.О., Семко І.Б., Кійко С.Г. Огляд сучасних методологій управління командами ІТ-проєктів. Управління розвитком складних систем. Київ, 2020. №43. С. 60-66. (0,4 д. а.). DOI: 10.32347/2412-9933.2020.43.60-66. URL: <http://mdcs.knuba.edu.ua/article/view/219835>.
6. Близнюкова І.О., Тесленко П.О., Данченко О.Б. Інструменти управління ІТ-проєктами з інноваціями. *Збірка тез VIII Міжнародної НТК «Інформатика. Культура. Технології»* ІКТ-2021. Одеса : ІКС, 2021. С. 84-86. URL: [http://ics\\_conf.tilda.ws/ict\\_eng](http://ics_conf.tilda.ws/ict_eng).
7. Margarita André, María G. Baldoquín, Silvia T. Acuca. Formal model for assigning human resources to teams in software projects. *Information and Software Technology*, 2011. № 53. pp. 259–275.

8. Близнюкова І.О. Метод формування креативної команди ІТ –проєкту. *Вісник Національного технічного університету «ХПІ». Серія: Стратегічне управління, управління портфелями, програмами та проєктами: зб. наук. пр.* Харків, 2023. № 1(7). – С. 12–18. <http://pm.khpi.edu.ua/article/view/286393>.
9. Team building criteria in software projects: A mix-method replicated study / Fabio Q.B. da Silva, A. César C. França, Marcos Suassuna and others. *Information and Software Technology*, 2013. № 55. pp. 1316–1340.
10. Близнюкова І.О. Концепція створення мінімально життєздатного продукту та дизайн-мислення в управлінні командою ІТ-проєкту / І.О. Близнюкова, П.О. Тесленко, О.Б. Данченко, В.М. Меленчук. *Вісник Національного технічного університету «ХПІ». Серія: Стратегічне управління, управління портфелями, програмами та проєктами: зб. наук. пр.* Харків: НТУ «ХПІ», 2021. №2(4). С. 11 – 17. DOI: 10.20998/2413-3000.2021.4.2. URL: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/52329>.
11. The Nine Belbin Team Roles. URL : <https://www.belbin.com/about/belbin-team-roles>.
12. Ballesteros-Pérez , P. Human resource allocation management in multiple projects using sociometric techniques / P. Ballesteros-Pérez, Ma. C. González-Cruz, M. Fernández-Diego. *International Journal of Project Management*, 2012. № 30. pp. 901–913.
13. Yilmaz, Murat. An examination of personality traits and how they impact on software development teams / Murat Yilmaz [and others]. *Information and Software Technology*, 2017. № 86. pp. 101–122.
14. Близнюкова І.О. Технології дизайн-мислення в управлінні командою ІТ-проєкту /І.О. Близнюкова О.Б. Данченко, П.О. Тесленко, В.І. Куваєва. *Управління проєктами: стан та перспективи : Матеріали XVII Міжнародної науково-практичної конференції*. Миколаїв : Видавець Торубара В.В., 2021. С. 13 – 14.
15. Liu, Wen-Hsing. Jennifer A. Cross. A comprehensive model of project team technical performance. *International Journal of Project Management*, 2016. № 34. pp. 1150-1166.

16. Team building criteria in software projects: A mix-method replicated study / Fabio Q.B. da Silva, A. César C. França, Marcos Suassuna and others. *Information and Software Technology*, 2013. № 55. pp. 1316–1340.
17. Teamwork quality and project success in software development: A survey of agile development teams / Yngve Lindsjörn, Dag I.K. Sjøberg, Torgeir Dingsøy and other. *The Journal of Systems and Software*, 2016. № 122. pp. 274–286.
18. Behrendt Sebastian, Richter Alexander, Trier Matthias. Mixed methods analysis of enterprise social networks. *Computer Networks*, 2014. № 75. pp. 560–577.
19. Ford Robert C., Piccolo Ronald F., Ford Loren R. Strategies for building effective virtual teams: Trust is key. *Business Horizons*, 2017. № 60. pp. 25-34.
20. Software Extension to the PMBOK® Guide Fifth Edition. Project Management Institute. Publ., 2013. URL: [www.PMI.org](http://www.PMI.org). 240 p.
21. Близнюкова І.О. Концептуальна модель креативного управління командою ІТ проєкту. /І.О. Близнюкова, О.Б. Данченко, П.О. Тесленко, Заруцький С.О. *Тези доповідей VI Міжнародної науково-практичної конференції РЗМ-2021* // Відповідальний за випуск Тесленко П.О. Одеса: ІШІР, 2021. С. 81-83. URL:<https://drive.google.com/drive/u/2/folders/1A19cjGj7O0NouNq9exxMQfd19AugkhLg>.
22. Raiden Ani B., Dainty Andrew R.J., Neale Richard H. Current barriers and possible solutions to effective project team formation and deployment within a large construction organisation. *International Journal of Project Management*, 2004. № 22. pp. 309–316.
23. Expanding the horizons of digital social networks: Mixing big trace datasets with qualitative approaches / Eoin Whelan, Robin Teigland, Emmanuelle Vaast and other. *Information and Organization*, 2016. № 26. pp. 1–12.
24. Caupin, Gilles. ICB IPMA Competence Baseline Version 3.0 / Gilles Caupin, Hans Knoepfel, Gerrit Koch, Klaus Pannenbäcker, Francisco Pérez-Polo, Chris Seabury. *International Project Management Association*, 2006. 200p.

25. Hsu, Shu-Chien. Understanding the complexity of project team member selection through agent-based modeling / Shu-Chien Hsu [and others]. *International Journal of Project Management*, 2016. № 34. pp. 82–93.

26. Teslenko P., Antoshchuk S. Evolution-attractive project management. *Управління проектами: Стан та перспективи: Матеріали XII Міжнародної науково-практичної конференції*. Миколаїв : НУК, 2017. С. 145 – 146.

27. Indika Dissanayake, Jie Zhang & Bin Gu. Task Division for Team Success in Crowdsourcing Contests: Resource Allocation and Alignment Effects. *Journal of Management Information Systems*, 2015. Vol. 32 №2 pp. 8-39. DOI: 10.1080/07421222.2015.1068604.

28. Близнюкова І.О., Тесленко П.О., Малахова Д.О. Особливості формування команди управління ІТ-проектом. *Вісник національного технічного університету «ХПІ» : Зб.наук.пр. Серія : Стратегічне управління, управління портфелями програмами та проектами*. Харків: НТУ «ХПІ», 2022. № 2 (6). С. 14 – 20. DOI: <https://doi.org/10.20998/2413-3000.2022.6.3>. URL: <http://pm.khpi.edu.ua/article/view/262312>.

29. Forming of integrated understanding of project terms: Faulk’s algorithm as one of the formalized approaches /Pepchuk S., Kolomytseva O., Verenych O., Danchenko O., Bielova O., Palonna T. *Entrepreneurship and Sustainability*, 2020. Issues 7(4): 3215-3225. DOI: [https://doi.org/10.9770/jesi.2020.7.4\(42\)](https://doi.org/10.9770/jesi.2020.7.4(42)), (Scopus).

30. Данченко О. Б. Методологія інтегрованого управління відхиленнями в проектах : автореф. дис. ... д-ра техн. наук : 05.13.22 / Данченко Олена Борисівна ; Київ. нац. ун-т буд-ва і архітектури. - Київ, 2015. - 45 с. - Бібліогр. : с. 35 – 41.

31. Близнюкова І.О., Данченко О.Б., Тесленко П.О. Протидія наслідкам емпатії у команді управління ІТ-проектом. *Управління проектами у розвитку суспільства. Тема: «Управління проектами в очікуванні глобальної кризи»: тези доповідей*. Київ: КНУБА, 2022. С. 75 – 77.

32. Bushuyeva Natalya, Kutcenko Maryna. Data mining techniques in projects with multinational teams. *XIV International Scientific and Technical Conference*

«Computer Sciences and Information Technologies», September 17 – 20, 2019. Lviv, Ukraine. URL: <https://ieeexplore.ieee.org/document/8929801> (Scopus).

33. Buontempo F. “Genetic Algorithms and Machine Learning for Programmers”. The Pragmatic Programmers, LLC, 2019, 234p.

34. Mitchell, Melanie (1998) An introduction to genetic algorithms. Cambridge: MIT Press, 1998. 158 p. ISBN 0–262–13316–4.

35. Banzhaf, W. Genetic programming: an introduction / Banzhaf, W., Nordin, P., Keller, R. E., & Francone, F. D. San Francisco: Morgan Kaufmann Publishers, 1998.

36. DEAP: Evolutionary algorithms made easy / Fortin, F.A., De Rainville, F.M., Gardner, M.A., Parizeau, M., & Gagné, C. *Journal of Machine Learning Research*, 2012. Vol.13. pp. 2171-2175.

37. Kaur G. & Bala A. An efficient resource prediction-based scheduling technique for scientific applications in cloud environment. *Concurrent Eng Res Appl*, 2019; Vol. 27 No 2:112–125.

38. Gawali M. B. & Shinde S. K. Task scheduling and resource allocation in cloud computing using a heuristic approach. *Journal of Cloud Computing: Advances, Systems and Applications*, 2018; Vol. 7 No 4:1-16. DOI: <https://doi.org/10.1186/s13677-018-0105-8>.

39. A Johnson’s-Rule-Based Genetic Algorithm for Two-Stage-Task Scheduling Problem in Data-Centers of Cloud Computing / Xiong Y., Huang S., Wu M., She J. & Jiang K. *IEEE Transactions on Cloud Computing*, 2019. Vol. 7 No 3: 597 – 610. DOI: 10.1109/TCC.2017.2693187.

40. Johnsson, M. Creating High-performing Innovation Teams. *Journal of Innovation Management*, 2017. Vol. 5 (4): 23-47. URL: <https://www.diva-portal.org/smash/get/diva2:1170247/fulltext02>.

41. Gross, J. J. (Ed.). (2015). Handbook of emotion regulation (2nd ed.). The Guilford Press. 669 p. ISBN: 9781462520732.

42. Blyznyukova, I., Teslenko, P. (2023). Formation of a minimum viable IT project team using the genetic algorithm. *Technology Audit and Production*

Reserves, 2(2(70)). (0,45 д. а.). DOI: 10.15587/2706-5448.2023.277930. URL: <https://journals.uran.ua/tarp/article/view/277930>.

43. Shore, J., & Warden, S. (2021) The art of agile development. O'Reilly Media, Inc., 2nd edition, 537 p. ISBN 1492080691.



## **РОЗДІЛ 4**

### **ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ КРЕАТИВНОГО УПРАВЛІННЯ КОМАНДОЮ ІТ-ПРОЄКТУ З ІННОВАЦІЙНОЮ СКЛАДОВОЮ**

#### **4.1 Формалізація даних для застосування інформаційної технології**

##### **4.1.1 Опис проєкту**

Проект, у якому були випробувані результати дисертаційного дослідження, на етапі його завершення можна було класифікувати як середньо-великий. Він виріс разом із клієнтом — від початкової системи IoT до інтелектуальної системи аналізу даних у реальному масштабі часу. При цьому запланований спочатку апаратно-програмний комплекс залишився частиною інтелектуальної системи управління. На жаль, детальніший опис проєкту обмежений правами власності та комерційною таємницею. Представлений опис покликаний показати досить стрімкі зміни цілей та завдань проєкту, зміну методології управління та технології розробки від класичної водоспадної до «дуже гнучкої». Відповідно все це позначилося і на команді проєкту, чисельність якої та вимоги до неї змінювалися протягом ЖЦП [1].

Було сформовано нову команду під великий обсяг завдань, що змінилися. Процес підбору команди став одним із найскладніших та найдовших етапів. Було проаналізовано значну кількість резюме та проведено співбесіди. Особливі вимоги були пред'явлені до керівника проєкту та тимліда [2]. Далі вони вже брали участь у подальшому доборі команди, спираючись на власний досвід та компетентності.

Великий проєкт передбачає велику кількість комунікацій, що й вимагало змінити систему управління комунікаціями проєкту, а потім розробити автоматизовані засоби управління та обробки інформації.

Керівник проєкту відповідає за проєкт. Він організовує планування, формує таймінги з кожного завдання, визначає їх пріоритети, аналізує ризики. Крім того, він приймає ключові рішення про використання методології розробки та управління, рішення про запуск на продуктовому середовищі, залучення експертизи співробітників, бере участь при вирішенні проблем, що виникають. Таким чином, рішення, від яких безпосередньо залежить успішне завершення проєкту, приймається керівником проєкту суб'єктивно, на основі його знань, досвіду та компетентностей.

Відповідно, для керівника проєкту та для всіх інших ролей були сформовані переліки вимог.

Тімлід - остання ланка на проєкті, яка ухвалює рішення про запуск завдання у виробництво. Знайти готового тимліда досить складно, тому найефективніше готувати їх з власних програмістів. Зрозуміло, що не кожен програміст здатний бути тимлідом, у зв'язку з цим прописуються групові та індивідуальні цілі команди, про які було сказано у попередніх розділах.

Крім основних фахівців (дизайнерів, верстальників, програмістів, тестувальників) на великі проєкти часто потрібні додаткові ресурси, фахівці, що можуть підключатися впродовж ЖЦП [3].

Зазвичай у великому проєкті необхідні такі командні ролі:

1. Архітектор. Він дозволить мінімізувати ризики та забезпечить супровід розробленого функціоналу. Правильне підключення архітектора до робіт проєкту дозволить унеможливити, або значно зменшити рефакторинг.

2. Аналітик може підключитись на етапі проєктування. Зібрані аналітиком дані є основою майбутнього інтерфейсу.

У цьому проєкті веб-аналітика була застосована до тих завдань, де були спірні технічні та інтерфейсні питання. Такі протиріччя виникають не тільки при спілкуванні з клієнтом, але й всередині самої команди, яка після етапу емпатії поринула у проблеми замовника і самостійно проходить «шлях користувача продукту». У цьому випадку додавання ролі аналітика в команду дозволить

приймати проєктні рішення на основі вимірюваних результатів, наприклад, на основі спліт-тестів.

3. Проєктувальник продумує логіку роботи системи та взаємодію екранних форм. Як було сказано у першому розділі, розробка проєкту з використанням технології дизайн-мислення супроводжується створенням прототипів, а вони вимагають написання супровідної документації. Для рефакторингу функціоналу зі складними інтеграціями проєктувальник оптимізує поточну інтеграційну схему, виявляє недоліки та формує рекомендації щодо їх усунення.

Завантаження команди на проєктах може динамічно змінюватися. Вона може різко збільшитися, і різко зменшуватися. Тому існує можливість плаваючого обсягу завдань, що потребує коригування чисельності/завантаженості команди проєкту. Для цього можуть бути залучені віддалені команди/співробітники.

Як показує практика, на великих проєктах не можна обмежитися однією технологією розробки та реалізовувати весь проєкт тільки за водоспадом або за гнучкими технологіями. У цьому проєкті на різних етапах було використано комбінація методологій: спринти і канбан-дошки у межах ітерацій каскадної моделі. При цьому основою успішного завершення проєкту є участь усієї команди у процесах прийняття проєктних рішень.

У моделях та методах, розроблених у другому та третьому розділах, особлива увага приділялася часу на взаємодію команди з замовником/користувачем продукту проєкту та на обговорення робочих процесів усередині команди. Йдеться про необхідність узгодження з клієнтом робочих процесів (workflow), термінів проведення релізів, планування завдань та ретроспективи. І тому значна увага приділяється такій ролі в команді, яка буде контролювати розклад проєкту.

#### **4.1.2 Формалізація технічних компетентностей команди проєкту**

Основою формування команди проєкту є чітке бачення цілей та завдань проєкту. Крім професійних знань, потрібні мотиваційні тригери. Також важливим

є високий рівень підготовки кожного претендента та його зацікавленість, замотивованість участі у проєкті [4]. Відомо, що менш досвідчений, але мотивований співробітник більш значущий у підвищенні імовірності успішного завершення проєкту, ніж обмежений часом «супер-експерт». Тому оцінка рівня технічних компетентностей претендентів не може бути відокремлена від оцінки рівня ефективності їхньої командної роботи.

Виділяють шість критеріїв оцінки ефективності роботи в команді [5]:

- наявність професійного інтересу та стимулу до роботи;
- визнання досягнутих результатів;
- володіння міжособистісними відносинами;
- висока технічна база;
- висока кваліфікація претендентів;
- можливості професійного зростання.

Тому прийом претендента в команду проєкту на посаду програміста включає не тільки питання про його навички, але і різні логічні завдання, ІТ-кейси, завдання з розробки. Список таких завдань відомий і в рамках дисертаційного дослідження не розглядалися питання, пов'язані з їхньою генерацією чи вдосконаленням. Для інформаційної технології, що розробляється у роботі, передбачено використання стандартних професійних тестів для оцінки компетентностей претендентів залежно від вимог у конкретному проєкті, а також методики розшифровування отриманих результатів тестування.

Зазначимо лише, що оцінка результатів тестування має суб'єктивне забарвлення. Більше того, для одного проєкту оцінки претендента можуть бути інтерпретовані як особистий недолік, а в іншому як безперечна гідність.

Тому для «тонких налаштувань» розглянемо критерії, які забезпечать створення ефективної креативної команди:

- перелік необхідних компетентностей претендента, які корелюють із потребами проєкту;

- ефективні методики виявлення цих компетентностей у претендентів в команду проєкту;
- розробка правил, норм та традицій (корпоративної культури) креативної команди проєкту.

#### **4.1.3 Формалізація психологічних компетентностей команди проєкту**

Компетентності внутрішньокмандної взаємодії належать до складного багатокомпонентного завдання. Вміння внутрішньокмандної взаємодії можна визначити як готовність претендента на націлювання на результат шляхом творчого вирішення проблеми, самоврядування, самонавчання, самоосвіту, на вирішення конфліктів у команді [6].

Навички внутрішньокмандної взаємодії допомагають у створенні доброзичливої атмосфери в команді, отриманні нових навичок та вдосконаленні наявних, сприяють активній взаємодії та позитивній комунікації всередині команди, емпатії, отриманню знань та досвіду.

Проблема формалізації психологічних компетентностей у тому, що вміння внутрішньокмандної взаємодії складно виміряти кількісними і якісними показниками. Для цього використовують методи психологічної діагностики, що дозволяють виміряти рівень навичок, особистих якостей, готовності до діяльності, взаємодії у команді.

Діагностика особистісних якостей, як і технічних компетентностей у цій дисертаційній роботі виходить із загальновизнаних методів і моделей у цій галузі. До них відносяться: виявлення командних ролей за класифікатором М. Белбіна, методика «Самоаналізу особистості», оцінка самоконтролю у спілкуванні за тестом М. Снайдера, оцінка рівня комунікативності по тесту В.Ф. Раховського, тест К. Томаса за типами поведінки у конфлікті та інші інструменти, які будуть застосовуватися для конкретного проєкту та конкретних завдань.

Для побудови креативної команди, що використовує технологію ДМ і MVP [6], необхідно оцінити такі показники:

- емоційний інтелект особистості, як члена команди:
  - емпатія;
  - позитивний настрій;
  - натхнення;
  - управління емоціями;
  - контроль амбіцій;
  - абстрагування від особистих симпатій та антипатій;
- внутрішні взаємодії члена команди:
  - комунікабельність;
  - делегування обов'язків;
  - вирішення конфліктів;
  - толерантність;
  - налагодження діалогу;
  - правильна аргументація;
  - визнання своїх помилок та прийняття чужої точки зору;
  - надання допомоги та підтримки;
  - вміння вислухати та висловитися;
  - поважне ставлення до думки членів команди;
  - виконання трудового розпорядку та підтримання дисципліни;
  - володіння конструктивною критикою;
- особисті якості члена команди:
  - ініціативність та цілеспрямованість;
  - готовність до навчання, самонавчання та саморозвитку;
  - готовність взяти на себе роль, обов'язки, завдання;
  - готовність поставити мету та йти до неї;
  - готовність приймати колективні рішення;
  - підтримка командного духу;
  - креативність;
  - критичне мислення;
  - відповідальність за виконання завдань;

- організованість;
- рефлексія;
- лідерські навички.

Відповідні методики та тести оцінюють початковий рівень претендентів, надалі, при включенні претендентів у команду проєкту, вони здійснюють внутрішньокomандну взаємодію, проходять необхідне навчання та тренінги за відсутніми позиціями з метою корекції поведінки членів команди, і потім тестування повторюється через деякий інтервал часу.

Тривалість інтервалу між тестуваннями визначається складністю проєктних робіт, тривалістю ЖЦП, видом компетентності чи досвіду, який підвищувався з допомогою тренінгу чи навчання. Як правило, вважається за доцільне проводити повторну оцінку компетентностей після 2-3 тижнів по закінченню навчання, або дата тестування визначається спеціальним чином.

#### **4.2 Приклад формування команди ІТ-проєкту**

У даному розділі подано результати застосування розроблених у дисертаційній роботі моделей методів та засобів на практичних прикладах [7]. У цьому підрозділі прописано процедуру створення команди проєкту, який було презентовано у пункті 4.1.1. Як було зазначено раніше, проєкт передбачав розробку модуля для системи IoT [8]. Для успішного досягнення цілей проєкту було ідентифіковано склад учасників проєкту, визначені їх ролі, порядок взаємодії учасників проєкту, сформована команда проєкту.

Команда проєкту складається з: проєктного менеджера, тімліда, програмістів (1-3), тестувальників (1-2), інженера та science-інженера.

Склад команди визначено за обсягом робіт, що виконувалися за гнучкою технологією. Простий аналіз складу команди показав наявність 22% управлінців та 78% виконавців (розробників). Подальшу розробку плану управління командою ІТ-проєкту виконано з використанням RACI-матриці, як одного з

різновидів матриці відповідальності [9]. Натомість, самої матриці RACI теж існує декілька варіантів, в залежності від потреб деталізації:

- класична RACI;
- RACI (alternative);
- RACI (decisions);
- RASCI (RASIC);
- RACI-VS (VS-RACI);
- RACIO (CAIRO);
- DACI;
- RSI.

На ранніх стадіях життєвого циклу проекту зазвичай будують укрупнену матрицю відповідальності, більш пізніх - детальну [9].

В проекті були застосовані класичні значення матриці:

- accountable (відповідальний) – відповідає за виконання завдання, і має право приймати рішення, пов'язані зі способом виконання. За одну задачу проекту відповідає тільки одна особа;
- responsible (виконавець) – виконує завдання, але не відповідає за вибір методу її вирішення;
- consulted (консультант) – консультує щодо вирішення питань проекту та веде контроль якості виконання;
- informed (спостерігач) – його інформують про вже прийняте рішення, взаємодія з ним має односторонній характер.

Навіть за таким складом матриця RACI є зручним інструментом візуалізації та проектування будь-якого процесу управління.

Далі було спроектовано матрицю, в якій задіяні наступні ролі: R – виконавець, A – відповідальний, C – контролюючий; I – особа, яку інформують про прийняті рішення та поточний стан проекту. Фрагмент матриці відповідальності подано на рис. 4.1.

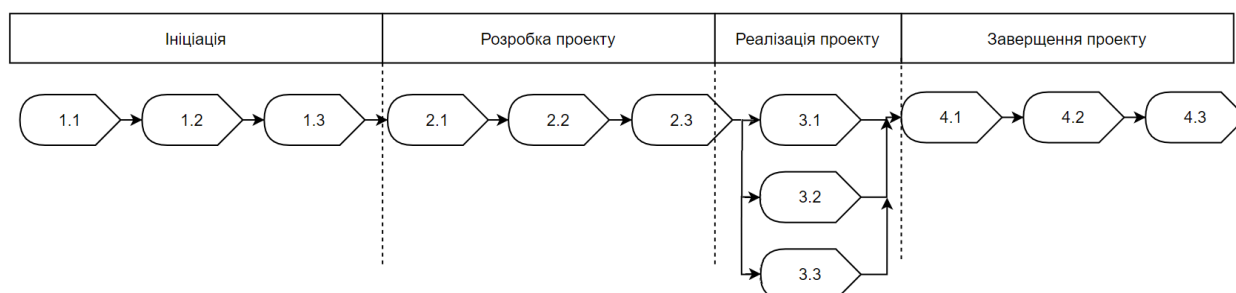
З урахуванням доданих ресурсів та відповідальних була сформована сітьова діаграма, яка в управлінні проектами відображає повний комплекс робіт із



встановленими між ними залежностями. Графічно вони виглядають як множина вершин, відповідних завданням, пов'язаних лініями, що представляють взаємозв'язки між роботами. Такий граф ще називають мережею типу вершина-робота чи діаграмою попередження і для даного проєкту він поданий на рис. 4.2.

Задача	Члени команди							
	Проектний менеджер	Тім лід	Програміст 1	Програміст 2	Тестувальник 1	Програміст 3	Тестувальник 2	Інженер Science інженер
1.1.1 Узгодження потреби в проєкті з замовником	A, R							
1.1.2. Визначення основних параметрів і обсягів робіт		A, R						
1.2. Формулювання основної мети і завдання	A, R							
1.3.1. Формування функціональних вимог		A, R						
1.3.2. Формулювання не функціональних вимог	A, R							
1.4. Формування команди проєкту	A, R							
2.1.1. Укладення контрактів з замовником	A, R							
2.1.2. Укладення контрактів з командою	A, R							
2.2.1. Створення календарного плану	A, R							

**Рис. 4.1. Фрагмент матриці відповідальності проєкту**



### Рис. 4.2. Сітьова діаграма проєкту

Подальші пропозиції щодо формування команди управління для команди розробників, ґрунтується на вертикальному та горизонтальному аналізі отриманої матриці (рис. 4.1). Матриця RACI дозволяє наочно контролювати виконання завдань та швидко виявляти перевантажених чи недовантажених виконавців, або таких, які не задіяні у виконанні завдань.

Дешифровка вертикального аналізу означає:

- багато "R". Не завжди людина здатна контролювати настільки багато дій;
- немає порожніх комірок. Слід звернути увагу, що людина навряд чи здатна ефективно працювати над великою кількістю завдань;
- немає "R" або "A". Необхідно звернути увагу на доцільність цієї ролі в команді проєкту;
- багато "A". Варто замислитись над розподілом відповідальності.

Горизонтальний аналіз:

- немає "R". Слід звернути увагу на пошук виконавця завдання, і навпаки, занадто велика кількість «R» може говорити про дублювання виконання;
- багато "A". Може виникнути плутанина через те, що багато виконавців відповідає за одну дію, згідно принципу відповідальності, така ситуація неможлива;
- всі комірки заповнені. Варто звернути увагу, що одним завданням може займатися занадто багато виконавців, що уповільнить процес;
- багато "C". Може зашкодити швидкості реалізації дії чи завдання.

За результатом вертикального аналізу найбільшу кількість R має Проєктний менеджер - 14 робіт на виконання, що показує його велику завантаженість. Також, велику кількість робіт на виконання має Тімлід, що також є високим навантаженням. Це веде до низької продуктивності, що збільшить заплановані строки та бюджет. І хоча, чим більше завантажений робочий час, тим менше простоїв та інших втрат робочого часу, але й тим вище рівень екстенсивного

використання праці і, відповідно, продуктивності праці. Проте екстенсивне зростання має чіткі законодавчі межі тривалості робочого дня й тижня.

Інші робітники мають декілька порожніх клітинок, що каже про вдалий розподіл робіт. Хоча б одна заповнена клітинка "R" або "A" свідчить про те, що усі робітники є необхідними для розробки проєкту.

Результати горизонтального аналізу. Усі завдання мають виконавців. Кожна задача має не більше однієї «A», що свідчить про те, що не буде проблем з перевіркою робіт, а значить й менше часу на вирішення питань між тими, хто затверджує виконання, тобто задачі розподілені вірно.

Таким чином, застосування методики RACI дозволяє не тільки організувати ефективно виконання робіт та поділ відповідальності, а й створити систему відстеження результативності управлінської роботи.

Основні проблеми розподілення робіт у команді, які було виявлено під час аналізу матриці відповідальності – два робітника мають високу завантаженість робіт управляючого характеру (планування попереднього бюджету, оцінка ризиків, формування команди проєкту та інші).

Для забезпечення більш продуктивної та швидкої роботи, пропонується додати у команду проєкту ще одного менеджера, що дозволить проводити деякі роботи паралельно. Фактично, ці зміни ґрунтуються на компетентнісному підході, який став невід'ємною частиною системи управління персоналом організацій за останні роки [10].

Персонал як стратегічний актив володіє знаннями, навичками та моделями поведінки, які необхідні для успішного завершення проєкту. І зазвичай, кожна особа має та вдосконалює доволі вузький спектр власних компетенцій. Натомість, невеликі команди ІТ-проєктів до 10, а інколи й менше осіб, вимагають від кожного члена команди виконувати різні типи робіт, які вимагають звичайно й різних компетенцій. При цьому знижується продуктивність праці. Але, як свідчить історичний досвід, саме розподілення праці у XIX сторіччі стало причиною прориву у виробництві, технологіях та й в цілому, у житті.

Також, було виявлено, що керівник проєкту відповідає за усі роботи постановки задач різних областей проєкту. Це веде до того, що менеджер повинен бути залучений в усі тонкощі реалізації. Це також знижує ефективність його роботи, а також може вести за собою недостатню компетентність у багатьох питаннях, та ризики невиконання цих робіт. Для вирішення цієї проблеми запропоновано включити до проєкту команду управління, яка буде безпосередньо залучена до управління проєктом та прийняття управлінських рішень. Менеджери та члени команди (виконавці) звітують перед менеджером проєкту та несуть відповідальність за реалізацію запланованих робіт та результатів (відповідальність може варіюватися від окремого виділеного результату (документу, рішення) до завершеного етапу).

До команди управління проєктом було додано п'ять менеджерів: проєктний менеджер, менеджер бекенд розробки, менеджер фронтенд розробки, менеджер апаратної частини та менеджер з машинного навчання (ML розробки).

З урахуванням цього було побудовано нову матрицю відповідальності для команди управління проєктом після переформування команди, яку подано на рис. 4.3.

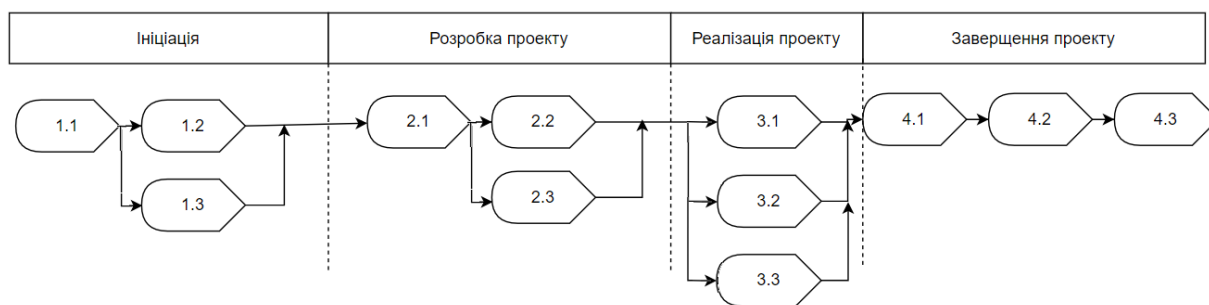
Проведений аналіз проєкту з врахуванням матриці відповідальності RACI показав можливість виявляти недоліки сформованої команди ІТ-проєкту, в плані виконання членами команди проєкту різних типів робіт, за різними компетентностями, що може знижувати продуктивність праці. На основі цього було запропоновано алгоритм формування команди управлінців, які мають володіти високим рівнем емпатії, чисельний склад цієї команди має бути незмінним продовж ЖЦП, а зміна чисельності команди, яку вимагатиме застосування технології MVP, буде проводитися під управлінням саме цієї команди управлінців. Це дало кращі результати по вартості та тривалості виконання проєкту. Нова сформована команда має більшу продуктивність, оскільки канали взаємодії з менеджерами проєкту налаштовані правильно. Таким чином, команда управління проєктом, що додалась до існуючого проєкту, сприяла покращенню усіх показників реалізації проєкту. Щодо проєкту, який

розглядається як приклад у дисертаційній роботі, було зафіксовано такі показники: зменшення бюджету на 7,3%; тривалості на 14 календарних днів [8].

Задача	Ресурси					
	Проектний менеджер 1	Проектний менеджер 2	Менеджер Бекенд розробки	Менеджер фронтенд роз-ки	Менеджер апаратної частини	Менеджер ML розробки
1	2	3	4	5	6	7
1.1.1 Узгодження потреби в проекті з замовником	A, R					
1.1.2. Визначення основних параметрів і обсягів робіт						
1.2. Формулювання основної мети і завдання		A, R				
1.3.1. Формування функціональних вимог						
1.3.2. Формулювання не функціональних вимог	A, R					
1.4. Формування команди проекту		A, R				
3.2.4. Навчання і тестування моделі						C
3.3.1. Постановка задачі				A, R		
3.3.2. Розподіл завдань у команді				A, R		
3.3.3. Написання програмного коду					A, R	
3.4.1. Написання переліку маршрутів			C			
3.4.2. Написання інструкції користувача		C				
4.1. Тестування продукції				C		
4.2. Підготовка документів угоди		A, R				
4.3. Здійснення угоди із замовником	A, R					

**Рис.4.3. Фрагменти матриці відповідальності для команди управління проектом**

Завдяки команді управління проектом вдалося розпаралелити задачі етапу ініціації та розробки, що відображено у оновленій сітьовій діаграмі (рис. 4.4).



**Рис. 4.4. Оновлена сітьова діаграма проєкту**

Таким чином, застосування команди управління ІТ-проєктом буде важливим та необхідним для команд, які використовують при розробці: технологію дизайн-мислення [10] та технологію MVP.

В даному випадку, недоліком команди управління проєктом буде збільшення кількості осіб, якщо критерієм була їх мінімізація, або ж збільшення бюджету, якщо він розраховуватиметься за місячною заробітною платою виконавців. Крім того, якщо критерієм оцінки є оптимізація процесів управління командою, застосування компетентнісного підходу, підвищення кваліфікації та застосування емпатії, то команда управління проєктом, як додаткова організаційна структура проєкту, безумовно має переваги, та рекомендується до застосування.

### **4.3 Розробка інформаційної технології креативного управління командою ІТ-проєкту**

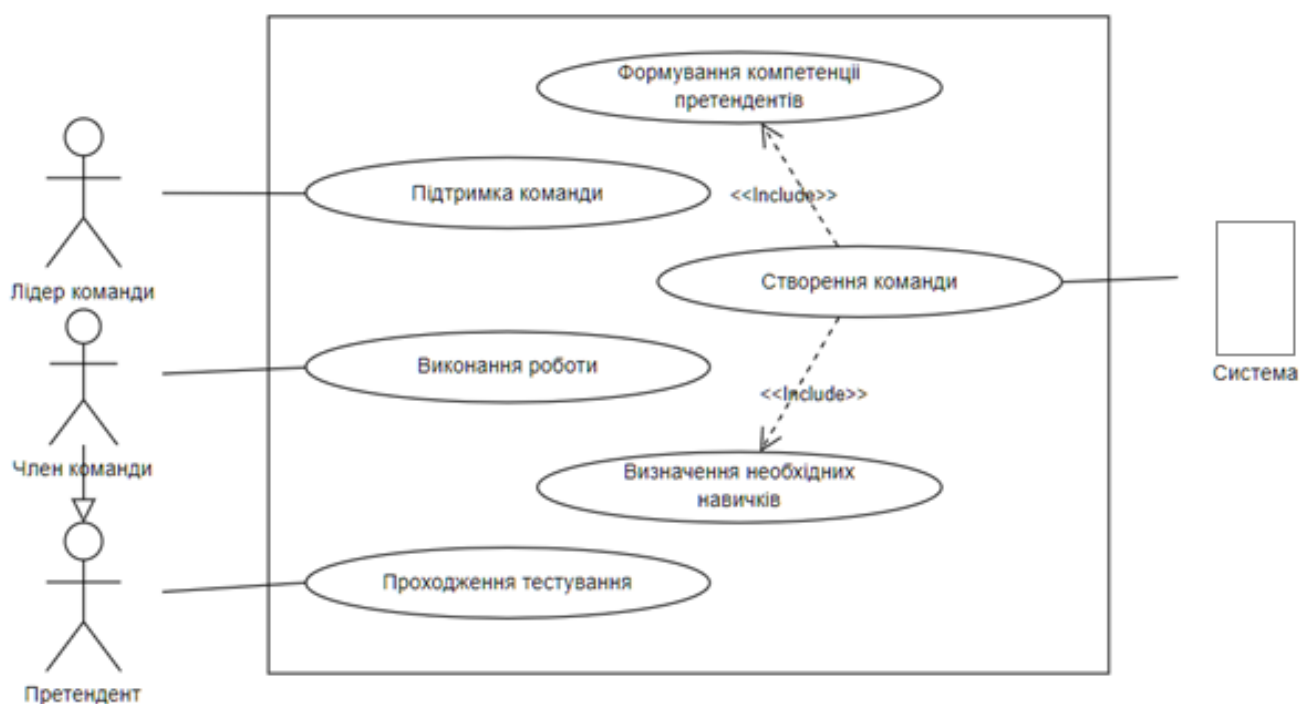
Інформаційна технологія створена з метою допомоги менеджерам проєктів в ефективному визначенні та формуванні команд, які будуть найкраще підходити для конкретних проєктів. ІТ містить у собі всі моделі та методи, що було розроблено впродовж дисертаційного дослідження. Крім того, застосування передових технологій та алгоритмів забезпечує високу точність налаштувань та ефективність відбору членів команди, враховуючи характеристики кожного, їх бажання та очікування, вимоги до проєкту та групову ціль команди.

Основою системи є генетичний алгоритм, що використовується для розв'язання оптимізаційної проблеми - вибору найкращих (за заданими критеріями) членів команди для досягнення максимальної ефективності - яка проявляється лише в успішному завершенні проєкту.

Генетичний алгоритм наслідує процеси природного відбору та дозволяє ІТ здійснювати ітеративний пошук оптимальних комбінацій співробітників на основі ряду критеріїв, таких як досвід, навички, вартість, сумісність та емпатія.

Ця система також автоматично враховує складність проєкту, яка розраховується на основі факторів, таких як терміни, потреби у навичках та обмеження бюджету. Визначення складності проєкту є цінним елементом, який дозволяє генетичному алгоритму точніше визначати оптимальні конфігурації команд.

Далі розглянемо технічну структуру системи, її ключові компоненти та особливості. Зазначимо також, що користувачі можуть взаємодіяти з системою та отримувати конкретні результати, що сприятиме ефективному управлінню проєктами та ресурсами. Діаграма акторів ІТ подана на рис. 4.5.



**Рис. 4.5. Діаграма учасників інформаційної технології креативного управління ІТ-проєктами**

В процесі розробки ІТ було використано кілька спеціалізованих бібліотек, які значно спрощують реалізацію складних алгоритмів та обробку даних:

Numpy - це відома бібліотека для Python, яка надає підтримку великих багатовимірних масивів та матриць, разом з великим набором математичних функцій для оперування цими масивами. В контексті даної системи, Numpy використовується для ефективної роботи з числовими даними, такими як характеристики співробітників і параметри проєктів.

Pandas є відкритою бібліотекою, що надає структури даних високого рівня (такі як DataFrame) та інструменти аналізу даних для мови програмування Python. У цій системі pandas використовується для ефективного управління та аналізу даних, таких як інформація про співробітників та проєкти.

DEAP (Distributed Evolutionary Algorithms in Python) - це відкрита бібліотека, що дозволяє легко розробляти та тестувати генетичні алгоритми. В системі DEAP використовується для реалізації генетичного алгоритму, який оптимізує вибір членів команди.

Scikit-learn - це потужна бібліотека для машинного навчання в Python, яка містить різноманітні алгоритми класифікації, регресії та кластеризації. У цій системі Scikit-learn використовується для додаткової аналітики та прогнозування.

Matplotlib використовується для створення графіків та візуалізації даних у Python. У контексті даної системи, ця бібліотека використовується для візуалізації характеристик проєктів та команд, дозволяючи менеджерам проєктів бачити графічне представлення результатів оптимізації.

В якості СУБД для цієї системи було обрано MongoDB. MongoDB є неструктурованою базою даних, яка використовує формат зберігання даних BSON, схожий на JSON. Вона добре підходить для роботи з великими обсягами даних і забезпечує високу продуктивність, гнучкість та масштабованість. Структура бази даних ІТ подана на рис. 4.6.

Однією з переваг MongoDB є те, що вона може легко справлятися з великою кількістю даних, що змінюються, таких як інформація про співробітників та проєкти. Це робить її ідеальним вибором для системи, що потребує швидкого



доступу до даних та можливості легко змінювати структуру даних без впливу на продуктивність.

Крім того, MongoDB дозволяє горизонтальну масштабованість, що дозволяє системі підтримувати високу продуктивність навіть при збільшенні обсягів даних. Це забезпечується за допомогою шардування, що дозволяє розподіляти дані між кількома серверами.



**Рис.4.6. Структура бази даних інформаційної технології креативного управління ІТ-проектами**

В контексті даної ІТ, MongoDB використовується для зберігання інформації про співробітників, проекти та історичні дані оптимізації. Це забезпечує швидкий доступ до даних та ефективне управління ресурсами.

Однією з ключових особливостей цієї ІТ є застосування генетичного алгоритму для оптимізації вибору членів команди для конкретного проекту. Генетичний алгоритм застосований, як пошуковий евристичний алгоритм, натхненний процесом природного відбору, який використовується для знаходження приблизних рішень в оптимізаційних та моделювальних задачах.

В даній системі, генетичний алгоритм імітує процес еволюції для пошуку оптимального набору співробітників для проєкту. Індивіди в популяції генетичного алгоритму представляють можливі комбінації співробітників, і вони оцінюються за допомогою функції пристосування, яка враховує різні характеристики проєкту та співробітників.

Основні операції генетичного алгоритму включають селекцію, схрещування (або рекомбінацію) та мутацію.

Селекція відбирає найкращі індивіди з популяції для утворення наступного покоління.

Схрещування комбінує хромосоми двох батьків для створення нових індивідів. Це може бути зроблено різними способами, такими як одноточкове схрещування, де випадкова точка вибирається на батьківській хромосомі та обмінюється для утворення нового потомства.

Мутація вносить невеликі випадкові зміни в хромосоми індивідів, що допомагає зберегти генетичне різноманіття популяції та забезпечує можливість виявлення нових точок пошуку в просторі рішень.

Даний процес повторюється через кілька поколінь, поки не буде досягнуто критерій зупинки, такий як максимальна кількість поколінь або задовільний рівень пристосування.

В даній ІТ генетичний алгоритм дозволяє автоматично та ефективно підбирати найкращі комбінації співробітників для проєктів, враховуючи різні фактори, такі як компетенції, вартість, якість роботи та терміни виконання.

Здатність системи масштабуватися є важливою характеристикою, яка визначає, наскільки добре вона може справлятися зі збільшенням навантаження. Для системи оптимізації команд проєктів це особливо актуально, оскільки кількість проєктів та співробітників може швидко рости.

Однією з ключових особливостей системи є її здатність горизонтального масштабування. Це означає, що замість збільшення потужності одного сервера (вертикальне масштабування), система може додавати додаткові сервери до пулу, щоб розподілити навантаження.

MongoDB, вибрана в якості бази даних, має вбудовані можливості горизонтального масштабування завдяки своїй підтримці шардування. Шардування дозволяє розподіляти набори даних через кілька серверів, ефективно розподіляючи навантаження та забезпечуючи високу доступність.

Крім того, система розроблена на основі Django, що є високопродуктивним вебфреймворком. Django підтримує різні механізми кешування, які можуть бути використані для оптимізації продуктивності та відгуку системи при високих навантаженнях. Також, можна використовувати різні вебсервери та балансувальники навантаження для подальшого масштабування системи.

Таким чином, ІТ має потужні можливості масштабування, що дозволяють їй ефективно справлятися зі збільшенням кількості даних та користувачів, не втрачаючи при цьому продуктивності та швидкодії.

#### **4.4. Реалізація розроблених моделей та методів у вигляді інформаційної системи**

Сучасний світ переповнений проєктами різного масштабу та складності, від невеликих внутрішніх ініціатив до глобальних підприємницьких проєктів. Управління проєктами стає все більш вимогливим, і тому пошук ефективних рішень та інструментів для оптимізації процесів та ресурсів є невід'язною частиною успіху. У цьому контексті, нова система, розроблена для оптимізації проєктних команд, вводить революційний підхід до управління проєктами.

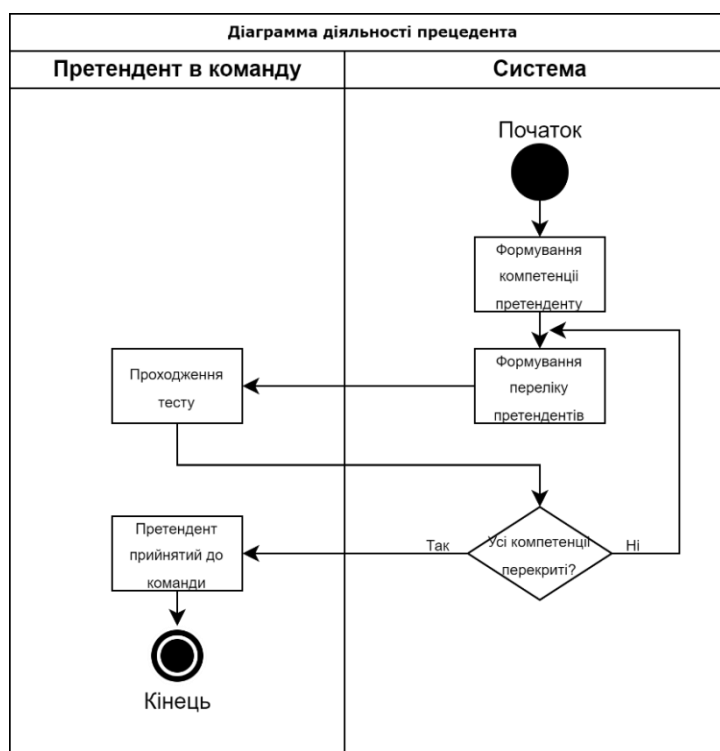
Система, що була розроблена, базується на мові програмування Python, яка відома своєю гнучкістю, широким спектром бібліотек та підтримкою різноманітних парадигм програмування. Python є однією з найпопулярніших мов для розробки комплексних систем, особливо коли йдеться про роботу з даними та машинне навчання, завдяки своїй високій продуктивності та легкості використання.

Щодо фреймворку, система використовує Django. Django — це високорівневий вебфреймворк на Python, який спрощує створення складних

веб-додатків, прискорюючи процес розробки та забезпечуючи чистоту та сучасність коду. Django дозволяє розробникам зосередитися на написанні логіки додатка, не турбуючись про низькорівневі деталі. Фреймворк також містить модулі для роботи з базами даних, аутентифікацією користувачів, шаблонами та іншими зручними функціями, що забезпечують швидку розробку та стабільність веб-проєкту.

Python та Django були обрані через їх здатність працювати в синергії, дозволяючи розробникам легко інтегрувати різноманітні бібліотеки та інструменти, що є важливими для реалізації складних алгоритмів та оптимізації роботи команди в цій системі.

Система оптимізації команд проєктів складається з кількох основних компонентів, які взаємодіють між собою, щоб надати користувачам засоби для створення проєктів, додавання співробітників та оптимізації команд за допомогою генетичного алгоритму. У якості прикладу розроблених процесів операцій ІТ, рис. 4.7 подано діаграму діяльності прецедента, а саме процес відбору претендентів до команди проєкту.



**Рис. 4.7. Процес відбору претендентів до команди проєкту**

Користувачі взаємодіють з системою через веб-інтерфейс, який розроблений з використанням Django. За допомогою інтерфейсу користувач створює нові проєкти, а саме описує проєкт (рис. 4.8) та продукт проєкту (рис. 4.9).

Назва проєкту: Інтелектуальна система управління розумним будинком для людей з особливими потребами		
Проблематика	Визначити тип, пакети та джерела даних задля оптимізації системи управління розумним будинком для людей з особливими потребами	
Мета проєкту	Мета проєкту Підвищити ефективність управління розумним будинком для людей з особливими потребами з метою забезпечення їхньої безпеки	
Опис продукту проєкту (до 200 слів)	Інтелектуальна система аналізу даних в системі IoT розумного будинку для людей з особливими потребами	
Бачення замовника	Отримання даних для побудови моделей поведінки людей з особливими потребами	
Очікування стейкхолдерів	Очікування стейкхолдерів Забезпечення зручних та комфортних умов проживання людей з особливими потребами через елементи та системи IoT	
Анотація проєкту (до 200 слів)	Планування та розробка проєкту створення розумного будинку для людей з особливими потребами на основі системи IoT з інтелектуальним аналізом даних. Аналіз наявних датчиків та мереж для збору даних. Визначення структури додаткових датчиків для збору даних що забезпечать плановане ефективне функціонування системи.	
Бачення розробників	Планування та реалізація апаратно-програмного комплексу IoT з інтелектуальним аналізом даних. Пошук оптимальної структури даних для забезпечення ефективного функціонування системи. Пошук оптимальної моделі аналізу даних. Розробки системи штучного інтелекту для забезпечення інтелектуального аналізу даних. Розробка та навчання нейронної мережі	
Опис технології розробки продукту	Опис технології розробки продукту Використання водоспадної моделі для розробки апаратної частини продукту проєкту. Використання гнучких технологій розробки для створення програмної частини ПП. Застосування технології MVP з прототипуванням та дизайн-мисленням.	
Ступінь інноваційності ПП. Апаратна частина (від 1 до 10)	Ступінь інноваційності технології розробки проєкту. Апаратна частина (від 1 до 10)	Ступінь інноваційності технології управління проєктом. Апаратна частина (від 1 до 10)
5	4	2
Ступінь інноваційності ПП. Програмна частина (від 1 до 10)	Ступінь інноваційності технології розробки проєкту. Програмна частина (від 1 до 10)	Ступінь інноваційності технології управління проєктом. Програмна частина (від 1 до 10)
8	9	6

**Рис. 4.8. Вікно введення для опису параметрів проєкту**

Обробка даних: Коли користувач вводить дані про проєкт та продукт проєкту, система зберігає цю інформацію в базі даних MongoDB. Додатково, система може виконувати попередню обробку даних за допомогою бібліотеки Pandas, зокрема, для нормалізації даних чи обробки відсутніх значень.

Формування команди. Коли користувач вибирає функцію «сформувати команду проєкту», система використовує генетичний алгоритм. Цей алгоритм працює з популяцією потенційних команд, оцінює їх ефективність за допомогою функції пристосування та застосовує генетичні операції, такі як селекція, схрещування та мутація, для пошуку оптимального складу команди.

Представлення результатів. Після завершення формування, система представляє результати користувачеві через веб-інтерфейс. За допомогою бібліотеки Matplotlib, система може візуалізувати результати, такі як якість рішення в різних поколіннях алгоритму.

Назва продукту	Інтелектуальна система управління розумним будинком для людей з особливими потребами				
Проблематика	Продукт є апаратно-програмним комплексом, що забезпечує формування, збір та інтелектуальний аналіз даних задля управління розумним будинком для людей з особливими потребами. Структура даних для аналізу не є визначеною та затвердженою, а потребує дослідження та вивчення при плануванні та розробці проєкту. Результати дослідження впливатимуть на структуру апаратної частини і на алгоритми та моделі інтелектуального аналізу даних.				
Функціонал продукту. Бачення замовника	1. модуль збору даних для забезпечення кліматичних та побутових умов 2. модуль для забезпечення організаційних. Медичних та супроводжувальних дій 3. система управління та реалізації умов проживання				
Функціонал продукту. Очікування стейкхолдерів	1. створення комфортних та безпечних умов для проживання осіб з особливими потребами 2. модуль допомоги				
Функціонал продукту. Бачення розробників	1. збір та групування даних з датчиків АЧ. 2. база даних 3. аналізатор даних. Нейронна мережа. 4. система управління розумним будинком 5. блок дослідження				
Анотація проєкту (до 200 слів)	Проект створення апаратно-програмного комплексу для управління розумним будинком для людей з особливими потребами. Етапу реалізації продукту передуватиме етап досліджень та пошуку оптимальної структури даних, засобів їх аналізу та системи управління.				
Перелік бібліотек і технологій розробки продукту	ESP-IDF, Spring, Hibernate, Home Assistant, Jetpack Compose, Kotlin Coroutines, Tensorflow				
Наявність аналогів. Ступінь інноваційності продукту проєкту (від 1 до 10)	Наявність технологій розробки. Ступінь інноваційності технологій розробки проєкту (від 1 до 10)	Наявність технологій проєктом. Ступінь інноваційності управління проєктом проєкту (від 1 до 10)	Ступінь бачення повноти функціоналу з боку замовника (від 1 до 10)	Ступінь бачення повноти функціоналу з боку розробників (від 1 до 10)	Рівень необхідного фінансування замовником (від 50% до 100%)
9/5	8/5	8/8	4	7	100%

**Рис. 4.9. Вікно введення для опису параметрів продукту проєкту**

Журналювання та моніторинг: Система також забезпечує журналювання та моніторинг дій користувачів та стану системи. Це включає в себе збір метрик, таких як час відгуку, використання ресурсів, та зберігання журналів операцій для подальшого аналізу.

Таким чином, система пропонує цілісний набір інструментів та ресурсів для оптимізації команд проєктів, враховуючи різні параметри та цілі, встановлені користувачем. Перш за все, ІС формує перелік вимог до команди проєкту на основі обробки даних про проєкт та про продукт проєкту. Приклад такого рішення приведений на рис. 4.10.

Унікальні функції розробленої ІС наступні:

1. Паралельно з формуванням вимог до команди проєкту, в рамках ІС проводиться тестування претендентів за технічними та психологічними компетенціями. Фрагмент результатів такого тестування подано на рис. 4.11.



2. Мультикритерійна оптимізація. Система здатна оптимізувати команди проєктів, враховуючи декілька критеріїв одночасно, таких як вартість, якість та терміни виконання. Це дає змогу знайти баланс між різними аспектами проєкту та тими ресурсами, які присутні на даний, конкретний час. Спираючись на це ІС.

Вимоги до команди проєкту	
Загальні технічні	
1	Аналітик – 1
2	Архітектор – 1
3	Си # Сеньор – 1
4	Си # Мідл – 2
5	Си # Джун – 3
6	Тестувальник – 1
7	Дизайнер – 1
Загальні (ступінь від 1 до 10) (для всіх членів)	
8	Вміння працювати в команді (від 1 до 10)
9	Досвід роботи в команді проєкту (від 1 до 10)
10	Досвід роботи з конкретними особами в команді проєкту (від 1 до 10)
11	Необхідний рівень емпатії (від 1 до 10)
12	Володіння декількома компетенціями/навичками (від 1 до 10)

**Рис. 4.10. Перелік вимог до команди проєкту**

3. Динамічна адаптація до зміни даних. Система може адаптуватися до зміни даних в реальному часі, наприклад, якщо змінюються характеристики співробітників або параметри проєкту, оптимізуючи команди відповідно до нових даних.

ФІО	Мови програмування	Знання технологій	Вміння працювати в команді	Досвід роботи в команді	Досвід роботи з особою	Рівень емпатії
Спіцин А.А. john.doe@gmail.com	C++ C#	Senior	Маються	Маються	Юрій Л.К.	Високий
Пепс А.Р. alex.ray@gmail.com	Python	Junior	Не визначено	Не визначено	Немає	Низький
Житомир Р.Р. kate.hunington@gmail.com	C++ Python	Senior	Немає	Немає	Немає	Негативний

**Рис.4.11. Фрагмент результатів тестування претендентів до команди проєкту**

Динамічна адаптація може проводитися як на початку проєкту, тоді пропонується рішення щодо персонального та кількісного складу команди, що є оптимальним на даний проміжок часу. Впродовж ЖЦП можуть бути змінені або доповнені проєктні завдання, що потребуватиме нового, або розширеного складу команди. це зазвичай може відбуватися при застосуванні технології MVP, для розробки програмного продукту. На рис. 4.12 подано приклад кількісного та персонального складу команди проєкту, який сформовано за результатами роботи генетичного алгоритму.

Оптимальний варіант складу мінімально життєздатної команди проєкту	
Загальні технічні	
1	Аналітик – Шевченко Ф.В.
2	Архітектор –
3	Си # Сеньор – Середа В. О.
4	Си # Мідл –
5	Си # Джун – Лисенко В.Т., Броварчук В. А., Васильчук Б. В.
6	Тестувальник – Тарашук С. О.
7	Дизайнер – Кравчук О.С.
Загальна кількість членів МЖК = 7 осіб	

**Рис. 4.12. Пропоноване рішення про персональний склад мінімально-життєздатної команди проєкту**

4. Розширена візуалізація результатів. Система надає користувачам розширені можливості візуалізації результатів оптимізації, включаючи графіки прогресу, гістограми розподілу характеристик та інші діаграми для глибокого аналізу результатів.

Ці унікальні функції роблять систему винятково потужним та гнучким інструментом для оптимізації команд проєктів, що відрізняє її від традиційних рішень на ринку.

Система надає інтуїтивний веб-інтерфейс для взаємодії з користувачами. Нижче наведено кроки, які користувач повинен виконати для роботи з системою:



Реєстрація та авторизація: На початку користувач повинен зареєструватися, вказавши свою електронну адресу та створивши пароль. Після реєстрації користувач може увійти в систему, використовуючи свої облікові дані.

Створення проєкту: На головній сторінці інтерфейсу користувач може створити новий проєкт, вказавши його назву та інші параметри, такі як бюджет, терміни та вимоги до якості.

Додавання співробітників: Користувач може додавати співробітників в систему вручну або завантажувати дані з файлу. Для кожного співробітника повинна бути вказана інформація, така як ім'я, вік, стаж роботи та спеціалізація.

Запуск оптимізації: Після того, як користувач ввів всі необхідні дані, він може запустити процес оптимізації команди за допомогою генетичного алгоритму. Система надає можливість налаштувати параметри алгоритму перед запуском.

Перегляд результатів: Після завершення оптимізації, користувач може переглянути результати на спеціальній сторінці, яка містить візуалізовану інформацію про оптимальний склад команди, а також інші метрики, такі як якість рішення та час виконання.

Експорт результатів: Система надає можливість експортувати результати оптимізації у різних форматах, таких як CSV, для подальшого аналізу або інтеграції з іншими системами.

Підтримка та зворотний зв'язок: В разі виникнення проблем або запитань користувач може звертатися до служби підтримки через вбудований модуль зворотного зв'язку в інтерфейсі системи.

Цей процес взаємодії забезпечує зручний та ефективний спосіб використання передових алгоритмів оптимізації для підбору найкращих команд проєктів.

Розроблена ІС креативного управління командою ІТ-проєктів представляє собою ефективне рішення для формування та управління командами проєктів, що базується на використанні генетичного алгоритму. Завдяки використанню мови програмування Python, фреймворку Django, ряду спеціалізованих бібліотек, а

також бази даних MongoDB, система забезпечує високу продуктивність та гнучкість.

Генетичний алгоритм, який є основою управління командою, відрізняється своєю адаптивністю та здатністю враховувати кілька критеріїв одночасно. Це дозволяє системі знаходити оптимальні комбінації претендентів для досягнення зазначених цілей проєкту.

Додатково, система має розширені можливості масштабування, що дозволяє ефективно обробляти великі набори даних та задовольняти потреби різних організацій незалежно від їх розміру.

У підсумку, представлена система є високотехнологічним і інноваційним рішенням для креативного управління командами ІТ-проєктів з інноваційною складовою, яке може знайти широке застосування серед керівників ІТ-проєктів та організацій, які прагнуть підвищити ефективність своїх проєктів та досягти конкурентних переваг на ринку.

#### **4.5 Висновки до 4 розділу**

1. В четвертому розділі представлені практичні результати застосування розроблених моделей та методів задля створення єдиної концепції креативного управління командою ІТ-проєкту, яка використовує технології дизайн-мислення та створення МЖП. Для проєктів з інноваційною складовою це може бути єдиним шляхом забезпечення успішного завершення проєктів і це може бути досягнуто через забезпечення самокерованості та саморганізованості команди ІТ-проєкту.

2. Згідно з розробленою у другому розділі концептуальною моделлю креативного управління командою ІТ-проєкту (підрозділ 2.2) спершу необхідно проаналізувати ІТ-проєкт та визначити вимоги до технічних та психологічних компетенцій членів команди. Саме це було подано у підрозділі 4.1, на прикладі конкретного ІТ-проєкту.

3. В підрозділі 4.2 запропонований підхід до створення креативної команди управління ІТ-проєктом. Доцільність формування такої ґрунтується на наступних

положеннях: 1) застосування технології створення МЖП передбачає збільшення проєктних завдань впродовж ЖЦП на етапах розробки прототипу продукту проєкту з розширеним функціоналом, що потребуватиме збільшення чисельного складу команди. 2) Технологія ДМ потребує наявності емпатії у всіх членів команди. Але це не завжди може бути доступним. Тому пропонується створити команду управління ІТ-проєктом, кожен з якої буде володіти емпатією, а до розробників та тестувальників вимоги щодо емпатії можуть бути пом'якшені. 3) Концепція створення МЖК може бути застосована лише до команди управління ІТ-проєктом, і збережена продовж усього ЖЦП, а зміна чисельності буде відображатися лише на команді розробників при зміні проєктного навантаження.

4. В підрозділі 4.2 показано, яким чином може бути створена команда управління проєктом з використанням матриці відповідальності RACI.

5. В підрозділі 4.3 представлено результати розробки інформаційної технології креативного управління командою ІТ-проєкту з інноваційною складовою, яка містить у собі результати дисертаційного дослідження.

6. В підрозділі 4.4 описаний розроблений веб-додаток, в якому реалізовано генетичний алгоритм, за допомогою якого менеджерам проєкту пропонується проєктні рішення щодо формування креативної команди ІТ-проєкту з інноваційною складовою.

7. За результатами дослідження опубліковано 3 статті [6, 7, 8].

## Список використаних джерел до розділу 4

1. A Guide to the Project Management Body of Knowledge (PMBOK® Guide) and the Standard for Project Management. Seventh Edition. USA. PMI, 2021. 274 p.
2. Managing Project Teams: URL: <https://www.ksl-training.co.uk/free-resources/performance-management/managing-project-teams/>.
3. Nataliia Dotsenko. Methodological support for formation of resource requirements in multi-project environment. URL: [https://www.researchgate.net/publication/340441984\\_Methodological\\_support\\_for\\_formation\\_of\\_resource\\_requirements\\_in\\_multi-project\\_environment](https://www.researchgate.net/publication/340441984_Methodological_support_for_formation_of_resource_requirements_in_multi-project_environment).
4. Precursors to engaged leaders in virtual project teams / Iorio J. et al. *International Journal of Project Management*, 2015. Vol. 33. PP. 395–405.
5. Katzenbach J., Smith D. (2015) *The Wisdom of Teams*. Boston: Harvard Business School Press, Reprint edition 304 p. ISBN 9781633691063.
6. Близнюкова І.О. Концепція створення мінімально життєздатного продукту та дизайн-мислення в управлінні командою ІТ-проєкту / І.О. Близнюкова, П.О. Тесленко, О.Б. Данченко, В.М. Меленчук. *Вісник національного технічного університету «ХПІ» : Зб.наук.пр. Серія : Стратегічне управління, управління портфелями програмами та проєктами*. Харків: НТУ «ХПІ», 2021. №2(4). С. 11 – 17. DOI: 10.20998/2413-3000.2021.4.2. URL: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/52329>.
7. Близнюкова І.О. Метод формування креативної команди ІТ –проєкту. *Вісник національного технічного університету «ХПІ» : Зб.наук.пр. Серія : Стратегічне управління, управління портфелями програмами та проєктами*. Харків: НТУ «ХПІ», 2023. № 1(7). с. 12–18. <http://pm.khpi.edu.ua/article/view/286393>.
8. Близнюкова І.О., Тесленко П.О., Малахова Д.О. Особливості формування команди управління ІТ-проєктом. *Вісник національного технічного університету «ХПІ» : Зб.наук.пр. Серія : Стратегічне управління, управління портфелями*

*програмами та проектами*. Харків: HTУ «ХІІІ», 2022. № 2 (6). С. 14 – 20. DOI: 10.20998/2413-3000.2022.6.3 URL: <http://pm.khpi.edu.ua/article/view/262312>.

9. Bob Kantor. The RACI matrix: Your blueprint for project success. URL: <https://www.cio.com/article/287088/project-management-how-to-design-a-successful-raci-project-plan.html>.

10. Udo, N. & Koppensteiner, S. *What are the core competencies of a successful project manager? Paper presented at PMI® Global Congress 2004—EMEA, Prague, Czech Republic*. Newtown Square, PA: Project Management Institute.

## ВИСНОВКИ

Дисертаційна робота вирішує актуальне науково-прикладне завдання, що полягало в розробці та вдосконаленні існуючих моделей, методів та інформаційних засобів управління командами ІТ-проектів з інноваційною складовою.

За результатами проведеного дослідження було зроблено наступні висновки.

1. Визначено сутність та особливості сучасних ІТ-проектів, досліджено специфіку управління ІТ-проектами з інноваційною складовою. Подано результати аналізу управління командами, що використовують гнучкі технології розробки програмного продукту, та сучасні креативні технології управління ІТ-проектами, такі, як технологія дизайн-мислення та концепція створення MVP.

Досліджено, що оточення ІТ-проектів має бути клієнт-орієнтованим та ціннісно-орієнтованим. З'ясовано, що задоволеність стейкхолдерів є головним критерієм успішного завершення ІТ-проектів з інноваційною складовою, що без проявленої емпатії до проблем замовника досягти неможливо.

В якості ефективних методологій управління ІТ-проектами прийняті клієнт-орієнтовні техніки гнучкого та креативного управління ІТ-проектами, а саме технологія дизайн-мислення, технологія створення мінімально-життєздатного продукту MVP, методика PoC. Показано, що саме ці техніки створюють найбільш успішне середовище для розробки ІТ-проектів з інноваційною складовою.

2. Проведений аналіз показав, що головним критерієм та складовою успішного завершення ІТ-проекту з інноваційною складовою є його команда. За результатами аналізу властивостей подібних команд було запропоновано та розроблено концептуальну модель креативного управління командою ІТ-проекту, яка дає змогу сформувати креативну, гнучку, мінімально-життєздатну команду управління ІТ-проектами з інноваційною складовою.

3. Розроблено семіотичну модель управління командою ІТ-проекту з інноваційною складовою та формальну ситуаційну модель, що слугує їй базисом.

Представлена модель на відміну від існуючих, дозволяє зміну набору правил, через базові елементи, семантичні та синтаксичні правила, аксіоми, що дозволить гнучко вносити зміни як в склад команди проекту, так і в процес прийняття рішень щодо управління командою на протязі ЖЦП.

4. Вдосконалено метод формування команди ІТ-проекту, який у процесі відбору та формування команди спирається на креативні здібності претендентів. В рамках методу запропонована процедура формування мінімально-життєздатної команди, в якій будуть сформовані сприятливі умови для розвитку процесів самокерованості та самоорганізації. Метод реалізовано на генетичному алгоритмі у зв'язку з тим, що задача формування є багатокритеріальною, а в рішенні відсутній головний критерій, який зазвичай, вважається критерієм успішності. Застосування генетичного алгоритму сприятиме виключенню суб'єктивного впливу експерта, що має приймати проєктні рішення при формуванні команди.

5. Розроблено метод креативного управління командою ІТ-проекту який включає продукування професійного зростання членів команди; розвинення їх емпатії (групової емпатії); розвиток емоційного інтелекту через застосування емоційного командного менеджменту; та досягнення спільної командної мети, яка включає в себе індивідуальні цілі кожного члена команди. Особливістю методу є те, що зазначені складові стануть не лише здобутками членів команди, а й єдино можливим важелем у забезпеченні успішного завершення ІТ-проєктів з інноваційною складовою.

8. Розроблено інформаційну технологію креативного управління командою ІТ-проекту, веб-додаток, в якому реалізовано генетичний алгоритм, за допомогою якого менеджерам проєкту пропонується проєктне рішення щодо формування команди ІТ-проекту з використанням розроблених вище моделей та методів.

9. Вдосконалена термінологічна база управління командами ІТ-проєктів за рахунок введення нових визначень «координатору команди ІТ-проєкту», «креативної команди ІТ-проєкту», «процесу створення креативної команди ІТ-проєкту» задля ефективного застосування на практиці розроблених моделей та методів управління ІТ-проєктами з інноваційною складовою.

10. Наведений приклад застосування на практиці розроблених інструментів креативного управління командою ІТ-проєкту.

Практичне значення результатів дисертаційного дослідження підтвердилися при застосуванні результатів дослідження при управлінні ІТ-проєктом на практиці через розроблену інформаційну технологію управління креативною командою.

Застосування в практиці управління ІТ-проєктами в компаніях розроблених дисертантом моделей, методів та інформаційних засобів дало позитивний ефект, оскільки була економія часу та фінансів в середньому на 5-7% порівняно з подібними проєктами компаній під час формування команд проєктів. Також відбувся розвиток якісного складу команд ІТ-проєктів та одночасно мінімізація кількості персоналу для реалізації проєктів орієнтовно на 9-10%.

Наукові та практичні результати дисертаційної роботи доцільно застосовувати в процесах управління командами ІТ-проєктів з інноваційною складовою.



## **ДОДАТКИ**

## ДОДАТОК А

## Акти впровадження результатів роботи

Товариство з обмеженою відповідальністю  
**«Фірма «МегаСтайл»**



тел. (0472) 520-550, 540-150  
 факс: (0472) 540-630  
 e-mail: office@megastyle.com

18000, Україна, м. Черкаси,  
 бул. Шевченка, 242а, к. 802  
 Р/р 2600521692 у ПАТ "Райффайзен Банк  
 Аваль" м. Київ МФО 380805 ЄДРПОУ 22803544

№ 52 від 12.06.23

на № \_\_\_\_\_ від \_\_\_\_\_

**АКТ**  
**використання результатів дисертаційної роботи**  
**Близнюкової Ірини Олександрівни**  
**«Інформаційна технологія креативного управління командами**  
**ІТ-проектів»**

Даним актом підтверджуємо, що розроблена Близнюковою І.О. інформаційна технологія креативного управління командами ІТ-проектів була використана в процесі реалізації кількох внутрішніх ІТ проектів компанії, що виконувались на протязі 2021р. та 2022р.

Використання запропонованого дисертантом програмного забезпечення, що розроблено на основі нових наукових моделей та методів, дало позитивний ефект, оскільки була економія часу на 5-7% порівняно з подібними проектами компанії під час формування команд проектів.

Результати дисертаційної роботи Близнюкової І.О. доцільно застосовувати в процесах управління командами ІТ-проектів.

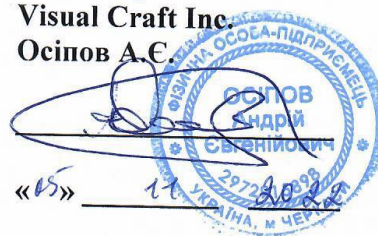
Комерційний директор  
 ТОВ «Фірма «МегаСтайл»



І.Г. Черкас

**ЗАТВЕРДЖУЮ**

Директор  
Visual Craft Inc.  
Осіпов А.Є.



«05»

11

2023

**АКТ**

**використання результатів дисертаційної роботи  
Близнюкової Ірини Олександрівни  
«Інформаційна технологія креативного управління командами  
ІТ-проектів»**

Даний акт підтверджує, що результати дисертаційної роботи Близнюкової І.О., а саме інформаційна технологія креативного управління командми ІТ-проектів, в основі якої лежать наукові результати, отримані здобувачем (семіотична модель управління командою ІТ-проекту, метод управління креативною командою ІТ-проекту та інші) застосовувались компанією в процесах управління наступними ІТ-проектами протягом 2022-2023:

- проект розробки прототипу програмного забезпечення для створення функціоналу веб серверу для промислових принтерів та маркувального обладнання, для забезпечення віддаленого доступу та управління обладнанням для компанії Videojet Inc.

- проект розробки прототипу веб-застосунку PitCrew для сканування водійських прав, отримання інформації про водія, генерування необхідних документів для проведення тест драйву авто.

Застосування розроблених Близнюковою І.О. науково-практичних інструментів є доцільним та дозволило знизити витрати часу та фінансових ресурсів на розробку продукту проектів на 9-11 %.

Директор

Осіпов А.Є.




**MIF PROJECTS**

mifprojects.com

**ЗАТВЕРДЖУЮ**

Операційний директор товариства  
з обмеженою відповідальністю

 «МІФ Проджектс»  
Ю. В. ДЬЯЧЕНКО  
«20» \_\_\_\_\_ 04 \_\_\_\_\_ 2023 р.

**АКТ**

**використання результатів дисертаційної роботи  
Близиукової Ірини Олександрівни  
«Інформаційна технологія креативного управління командами  
ІТ-проектів»**

Наукові та практичні результати дисертаційної роботи Близиукової І.О., а саме інформаційна технологія креативного управління командами ІТ-проектів, створена на основі моделей, методів та засобів формування команд

ІТ-проектів, планування їхніх задач під час реалізації проектів, та контролю виконання, застосовувались компанією в процесах управління наступними

ІТ-проектами протягом 2023 року:

- проект розробки прототипу програмного забезпечення для ресурсомістких геолокаційних обчислень;
- проект розробки прототипу веб-застосунку Інтернет магазину.

Застосування розроблених Близиуковою І.О. науково-практичних інструментів є доцільним та дозволило знизити витрати часу та фінансових ресурсів на розробку продукту проектів на 5-10 %.

Операційний директор,  
ТОВ «МІФ Проджектс»



Ю. В. Дьяченко

## ДОДАТОК Б

### Модель розвитку команди «Сходи Такмана»



## ДОДАТОК В

## Модель ефективності команди Drexler-Sibbet

