

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

ЕКОНОМІЧНИЙ ФАКУЛЬТЕТ

**КАФЕДРА МЕНЕДЖМЕНТУ
ІННОВАЦІЙНОЇ ТА ІНВЕСТИЦІЙНОЇ ДІЯЛЬНОСТІ**

**КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА
на тему: «ГНУЧКЕ УПРАВЛІННЯ ПРОЄКТАМИ ІТ-
ПІДПРИЄМСТВА»**

**здобувача освіти за ОС «магістр»
денної форми навчання**

**галузь знань 07 «УПРАВЛІННЯ ТА АДМІНІСТРУВАННЯ»
спеціальність 073 «МЕНЕДЖМЕНТ»
освітньо-наукова програма
«МЕНЕДЖМЕНТ ОРГАНІЗАЦІЙ І АДМІНІСТРУВАННЯ»**

ПІСАРЕВА ЯРОСЛАВА РОМАНОВИЧА

**Науковий керівник:
к. е. н., доцент
Приймак Василь Михайлович**

Рекомендовано до захисту
на засіданні кафедри менеджменту
інноваційної та інвестиційної діяльності
протокол № 14 від 14 травня 2026 р.

В. о. завідувача кафедри
_____ доцент Фірсова С.Г.

Київ – 2026

Київський національний університет імені Тараса Шевченка
Економічний факультет
Кафедра менеджменту інноваційної та інвестиційної діяльності

"Затверджую"

В. о. завідувача кафедри менеджменту інноваційної та
інвестиційної діяльності, доц. Фірсова С.Г.

«11» вересня 2025 р.

ЗАВДАННЯ

на кваліфікаційну роботу магістра
здобувача освіти за ОС «магістр» денної форми навчання
галузь знань 07 «Управління та адміністрування»
спеціальність 073 «Менеджмент»

ОНП «Менеджмент організацій і адміністрування»

ПІСАРЕВА ЯРОСЛАВА РОМАНОВИЧА

Тема роботи: «Гнучке управління проектами ІТ-підприємства»

затверджена на засіданні кафедри менеджменту інноваційної та інвестиційної діяльності
11.09.2025, протокол № 2,

редакційно уточнена на засіданні кафедри менеджменту інноваційної та інвестиційної
діяльності 09.03.2026, протокол № 11.

Строк завершення роботи: 05.05.2026.

Попередній захист роботи: квітень 2026 р.

Предмет дослідження: теоретичні, методичні та прикладні аспекти організації,
діагностики й удосконалення гнучкого управління ІТ-проектами у JustMarkets Tech.

Об'єкт дослідження: процес гнучкого управління проектами ІТ-підприємства.

Мета і завдання дослідження: Мета роботи полягає в обґрунтуванні теоретичних
засад, аналізі практики застосування та розробленні напрямів удосконалення гнучкого
управління проектами ІТ-підприємства на прикладі JustMarkets Tech.

Завдання:

6.1. Розкрити сутність, принципи, переваги, обмеження та бізнес-цілі гнучкого
управління проектами в ІТ-сфері.

6.2. Охарактеризувати методичні підходи до гнучкого управління ІТ-проектами.

6.3. Проаналізувати організаційну побудову системи управління ІТ-проектами у
JustMarkets Tech та роль основних функціональних напрямів у проектному циклі.

6.4. Провести діагностику ефективності застосування гнучких методів
управління проектами у JustMarkets Tech за продуктовими, організаційними,
технічними та клієнтськими критеріями.

6.5. Сформуванати напрями оптимізації процесів гнучкого управління проектами у
JustMarkets Tech на основі результатів проведеної діагностики.

6.6. Обґрунтувати впровадження інтегрованої моделі гнучкого управління ІТ-
проектами у JustMarkets Tech та оцінити економічну ефективність запропонованого
проекту.

Науковий керівник

доц. Василь ПРИЙМАК

Здобувач освіти



Ярослав ПІСАРЕВ

Календарний план виконання завдання

№	Зміст виконаної роботи	Строки виконання
1.	Вибір теми магістерської роботи	червень 2025
2.	Затвердження теми магістерської роботи	вересень 2025
3.	Ознайомлення з науково-інформаційними джерелами за обраним напрямом досліджень, виявлення наукової проблематики та формування бібліографії магістерської роботи	вересень – грудень 2025
4.	Підготовка тексту доповіді для участі у науковій конференції, підготовка й опублікування тез у матеріалах наукової (науково-практичної) конференції та наукової статті за обраним напрямом досліджень	жовтень 2025 – березень 2026
5.	Розробка плану магістерської роботи, визначення об'єкта, предмета, мети і завдань дослідження. Розробка завдань та графіку виконання кваліфікаційної роботи магістра. Узгодження їх із науковим керівником кваліфікаційної роботи магістра	січень – лютий 2026
6.	Пошук інформаційних матеріалів і робота над першим розділом. Оформлення першого розділу та подання його на розгляд науковому керівникові	січень – лютий 2026
7.	Пошук інформаційних матеріалів і робота над другим розділом. Оформлення другого розділу та подання його на розгляд науковому керівникові	березень 2026
8.	Підготовка третього (конструктивного) розділу	березень – квітень 2026
9.	Консультація з приводу оформлення роботи	квітень 2026
10.	Доопрацювання та остаточне оформлення роботи з урахуванням пропозицій попереднього захисту і зауважень наукового керівника	квітень 2026
11.	Передзахисти магістерської роботи	березень - квітень 2026
12.	Усунення зауважень, які були зроблені на підсумковому передзахисті роботи	до 05.05.2026
13.	Завершення написання магістерської роботи і подача науковому керівникові для підготовки відгуку	06.05.2026
14.	Перевірка роботи на текстові збіги	07-08.05.2026
15.	Зовнішнє рецензування магістерської роботи	травень 2026
16.	Рекомендація магістерської роботи до захисту на засіданні кафедри менеджменту інноваційної та інвестиційної діяльності	травень 2026

Науковий керівник

доц. Василь ПРИЙМАК

Здобувач освіти



Ярослав ПІСАРЕВ

ЗМІСТ

ВСТУП	5
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ГНУЧКОГО УПРАВЛІННЯ ПРОЄКТАМИ ІТ-ПІДПРИЄМСТВА	10
1.1. Сутність та принципи гнучкого управління проєктами в ІТ-сфері	10
1.2. Методичні підходи до гнучкого управління ІТ-проєктами	18
РОЗДІЛ 2. АНАЛІЗ ГНУЧКОГО УПРАВЛІННЯ ПРОЄКТАМИ У КОМПАНІЇ JUSTMARKETS TECH	28
2.1. Організаційна характеристика системи управління ІТ-проєктами	28
2.2. Діагностика ефективності застосування гнучких методів управління проєктами	37
РОЗДІЛ 3. УДОСКОНАЛЕННЯ СИСТЕМИ ГНУЧКОГО УПРАВЛІННЯ ПРОЄКТАМИ JUSTMARKETS TECH	46
3.1. Напрями оптимізації процесів гнучкого управління проєктами у JustMarkets Tech	46
3.2. Обґрунтування впровадження інтегрованої моделі гнучкого управління ІТ-проєктами у JustMarkets Tech	55
ВИСНОВКИ	68
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	71
ДОДАТКИ	78

ВСТУП

ІТ-підприємства функціонують в умовах швидкого оновлення цифрових продуктів, високої конкуренції, зміни вимог користувачів і постійного ускладнення технологічної інфраструктури. За цих умов ефективність проєктного управління залежить не лише від попереднього планування, а й від здатності команди оперативно уточнювати вимоги, погоджувати зміни між підрозділами, забезпечувати якісну технічну реалізацію та оцінювати результат після запуску продукту.

Особливого значення набуває гнучке управління проєктами, оскільки воно орієнтується на ітеративну розробку, регулярний зворотний зв'язок, пріоритезацію завдань, постачання робочого результату та постійне вдосконалення процесів. Для ІТ-підприємств гнучкі методи мають практичне значення через високу змінність вимог, потребу у швидкому виведенні оновлень на ринок, активну взаємодію між бізнесовими та технічними командами, а також залежність конкурентоспроможності від якості цифрового продукту.

Актуальність теми посилюється специфікою фінтех-сектору, де ІТ-проєкти мають підвищені вимоги до стабільності, безпеки, швидкості релізів, відповідності регуляторним обмеженням і якості клієнтського досвіду. Для брокерських платформ важливими є доступність сервісу, швидкість виконання операцій, надійність мобільного продукту, точність обробки клієнтських даних, узгодженість роботи продуктових, інженерних, операційних і compliance-команд. У таких умовах застосування Agile, Scrum, Kanban, Lean, DevOps і релізних метрик має значення для підвищення керованості ІТ-проєктів.

Теоретичні основи гнучкого управління проєктами сформовані у працях К. Beck, М. Beedle, А. van Bennekum, А. Cockburn, W. Cunningham, М. Fowler, J. Highsmith, R. С. Martin, К. Schwaber, J. Sutherland та інших авторів Agile Manifesto, де визначено пріоритет взаємодії учасників, робочого продукту,

співпраці із замовником і готовності до змін. Н. Dong, N. Dacre, D. Baxter і S. Ceylan розглядають Agile Project Management як сучасний підхід до управління проектами в умовах невизначеності та змінності вимог. К. Schwaber і J. Sutherland розкривають зміст Scrum як фреймворку, що ґрунтується на прозорості, інспекції, адаптації, ролях, подіях і артефактах. D. J. Anderson розвинув застосування Kanban у сфері розробки програмного забезпечення, акцентуючи увагу на візуалізації потоку, обмеженні незавершеної роботи та поступовому вдосконаленні процесів.

Порівняльні аспекти традиційного й гнучкого управління проектами досліджували N. Engelhardt, P. G. Pietrzak, M. Cieciora і K. Klimaszewski. У їхніх працях показано, що традиційні підходи більше відповідають проектам зі стабільними вимогами, тоді як Agile ефективніший за умов частих змін, активної участі замовника та потреби в регулярній перевірці результату. J. Y. Koi-Akrofi, H. A. Matey і G. Koi-Akrofi систематизували переваги й виклики Agile IT Project Management, зокрема адаптивність, комунікацію, самоорганізацію, ризик недостатньої документації, складність прогнозування строків і залежність результату від зрілості команди. D. Ćirić і D. Gračanin розглядали Agile Project Management ширше за межами програмної розробки, виділяючи бізнес-цілі гнучкого управління: безперервні інновації, адаптивність продукту, скорочення строків постачання, адаптивність людей і процесів та надійні результати.

Методичні аспекти застосування гнучких підходів у роботі IT-команд пов'язані також з Extreme Programming, Lean Agile та DevOps. К. Beck, W. Cunningham і R. Jeffries заклали основу Extreme Programming, де ключову роль відіграють короткі ітерації, часті релізи, автоматизоване тестування, рефакторинг і постійна взаємодія із замовником. Lean Agile орієнтується на зменшення втрат, швидку доставку цінності, навчання на основі зворотного зв'язку та підвищення якості продукту. DevOps-підхід розглядається як інтеграція розробки та експлуатації програмного забезпечення, що посилює безперервну інтеграцію,

автоматизоване тестування, моніторинг, релізний контроль і відповідальність команд за якість продукту протягом усього життєвого циклу.

Попри значну кількість досліджень, практичне питання діагностики ефективності гнучкого управління в межах окремого ІТ-підприємства залишається актуальним. Для компаній фінтех-сектору важливо не лише використовувати Agile-практики, а й оцінювати їх за продуктовими, технічними, організаційними та клієнтськими критеріями. Особливої уваги потребують потік завдань, релізні метрики, післярелізний аналіз, технічний борг і цифрова координація віддалених команд.

Метою кваліфікаційної роботи є обґрунтування теоретичних засад, аналіз практики застосування та розроблення напрямів удосконалення гнучкого управління проектами ІТ-підприємства на прикладі JustMarkets Tech.

Відповідно до поставленої мети визначено такі завдання:

1. розкрити сутність, принципи, переваги, обмеження та бізнес-цілі гнучкого управління проектами в ІТ-сфері;
2. охарактеризувати методичні підходи до гнучкого управління ІТ-проектами, зокрема Scrum, XP, Kanban, Lean Agile і DevOps;
3. проаналізувати організаційну побудову системи управління ІТ-проектами у JustMarkets Tech та роль основних функціональних напрямів у проектному циклі;
4. провести діагностику ефективності застосування гнучких методів управління проектами у JustMarkets Tech за продуктовими, організаційними, технічними та клієнтськими критеріями;
5. сформулювати напрями оптимізації процесів гнучкого управління проектами у JustMarkets Tech на основі результатів проведеної діагностики;
6. обґрунтувати впровадження інтегрованої моделі гнучкого управління ІТ-проектами у JustMarkets Tech та оцінити економічну ефективність запропонованого проекту.

Об'єктом дослідження є процес гнучкого управління проектами ІТ-підприємства.

Предметом дослідження є теоретичні, методичні та прикладні аспекти організації, діагностики й удосконалення гнучкого управління ІТ-проектами у JustMarkets Tech.

Базою дослідження є JustMarkets Tech, що забезпечує розробку та підтримку цифрових фінтех-рішень для глобального мультиактивного брокера JustMarkets. У роботі враховано функціонально-проектну побудову управління, взаємодію продуктових та інженерних команд, роль клієнтського зворотного зв'язку, значення DevOps-практик, релізного контролю та цифрової координації віддалених команд. Зміст роботи охоплює теоретичні основи, аналіз системи управління проектами у JustMarkets Tech і розроблення рекомендацій щодо її вдосконалення .

У процесі дослідження використано такі методи: аналіз і синтез для розкриття сутності гнучкого управління проектами; порівняльний метод для зіставлення традиційного та гнучкого управління; системний підхід для дослідження взаємодії підрозділів у межах ІТ-проектів; метод узагальнення для виокремлення критеріїв діагностики; експертно-аналітичний метод для оцінювання ефективності застосування гнучких методів; метод SMART для визначення вагомості критеріїв; розрахунковий метод для оцінювання економічної ефективності запропонованого проекту.

Інформаційну основу роботи становлять наукові праці з питань гнучкого управління проектами, Agile, Scrum, Kanban, Lean Agile, DevOps і DORA-метрик, матеріали Agile Manifesto та Scrum Guide, відкриті дані JustMarkets і JustMarkets Tech, аналітичні матеріали щодо ІТ-ринку, а також результати власних розрахунків автора.

Практичне значення роботи полягає у формуванні прикладних рекомендацій для підвищення ефективності гнучкого управління ІТ-проектами у

JustMarkets Tech. Запропоновано напрями оптимізації потоку завдань, релізних метрик, післярелізного аналізу, управління технічним боргом, клієнтського зворотного зв'язку та цифрової координації віддалених команд. Розроблено проєкт впровадження інтегрованої моделі гнучкого управління ІТ-проєктами, визначено етапи реалізації, склад пілотної команди, бюджет і показники економічної ефективності.

Результати роботи апробовано на XXIV Міжнародній науково-практичній конференції «Шевченківська весна 2026. Економічна резильєнтність в умовах глобальної полікризи», що відбулася 25-27 березня 2026 року, з тезами на тему «СУЧАСНІ ПІДХОДИ ДО УПРАВЛІННЯ ПРОЄКТАМИ У ІТ-ПІДПРИЄМСТВАХ» (Додаток А).

Робота складається зі вступу, трьох розділів, висновків, списку використаних джерел з 64 найменувань і 2 додатків. У першому розділі розкрито теоретичні основи гнучкого управління проєктами ІТ-підприємства. У другому розділі проаналізовано організацію управління ІТ-проєктами у JustMarkets Tech і проведено діагностику ефективності застосування гнучких методів. У третьому розділі розроблено напрями оптимізації та обґрунтовано впровадження інтегрованої моделі гнучкого управління ІТ-проєктами. Робота містить 25 таблиць і 2 рисунки.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ГНУЧКОГО УПРАВЛІННЯ ПРОЄКТАМИ ІТ-ПІДПРИЄМСТВА

1.1. Сутність та принципи гнучкого управління проєктами в ІТ-сфері

ІТ-сфера характеризується швидким оновленням технологій, високою конкуренцією, зміною вимог до цифрових продуктів і потребою в оперативному прийнятті управлінських рішень. У проєктній діяльності ІТ-підприємств це посилює значення підходів, які враховують невизначеність, часте уточнення завдань і необхідність постійної взаємодії між командою та замовником.

Традиційні моделі управління не завжди відповідають умовам ІТ-проєктів, оскільки орієнтуються на детальне попереднє планування та послідовне виконання етапів. Натомість розробка цифрового продукту часто потребує поетапного створення результату, регулярної перевірки функціоналу, зміни пріоритетів і швидкого реагування на нові вимоги.

Agile є гнучким підходом до управління проєктами, що ґрунтується на адаптивності, ітеративній організації роботи та швидкому реагуванні на зміну вимог. Основу Agile становлять положення Agile Manifesto, де пріоритет мають взаємодія учасників проєкту, робочий продукт, співпраця із замовником і готовність до змін замість жорсткого дотримання початкового плану. В управлінні ІТ-проєктами Agile орієнтований на роботу короткими циклами, регулярну оцінку проміжних результатів і уточнення подальших дій відповідно до потреб замовника [18, с. 674].

Для гнучких методів управління методами притаманні ознаки:

1. інкрементальна реалізація проєкту короткими ітераціями з регулярним переглядом планів;
2. постійна взаємодія між учасниками команди та замовником;

3. самоорганізація команди в межах визначених ролей і відповідальності;

4. оперативне коригування вимог без істотного ускладнення бюджету та строків виконання проєкту.

Управління ІТ-проєктами на основі Agile ґрунтується на поєднанні гнучкого планування, коротких циклів роботи, постійного зворотного зв'язку та активної участі команди у прийнятті рішень. На відміну від традиційних підходів, де основний акцент робиться на попередньо затверженому плані, Agile орієнтується на поступове створення результату, регулярну перевірку проміжних рішень і швидке коригування подальших дій відповідно до змін у вимогах замовника або умовах реалізації проєкту. Основними характеристиками Agile як гнучкого методу управління ІТ-проєктами є:

1. ітеративність та інкрементальність – реалізація проєкту короткими циклами з готовим функціоналом або проміжним результатом після кожної ітерації;

2. адаптивність – зміна пріоритетів, завдань і планів на основі актуальної інформації, зворотного зв'язку та уточнених вимог замовника;

3. самоорганізація команд – автономний розподіл завдань, вибір способів їх виконання та прийняття поточних рішень у межах визначеної відповідальності;

4. фокус на замовнику – активна участь замовника або представника бізнесу у визначенні вимог, оцінюванні проміжних результатів і коригуванні пріоритетів;

5. прозорість та комунікація – регулярні зустрічі, демонстрації результатів, відкритий обмін інформацією та своєчасне обговорення проблем;

6. спрямованість на цінність – орієнтація на робочий продукт і практичну користь для замовника, а не на обсяг документації чи формальне виконання плану;

7. постійне вдосконалення – регулярне оцінювання робочих процесів, виявлення проблемних місць і коригування підходів до організації командної роботи [15, 37].

У проєктній діяльності ІТ-підприємств вибір підходу до управління залежить від того, наскільки стабільними є вимоги, чи можливо наперед визначити кінцевий результат, як швидко змінюються пріоритети замовника та яким має бути рівень контролю за виконанням робіт (табл. 1.1.1).

Таблиця 1.1.1

Порівняння традиційного та гнучкого управління проєктами*

Критерій порівняння	Традиційне управління проєктами	Гнучке управління проєктами
Вихідне припущення	Повна специфікація системи до початку робіт; прогнозованість результату, строків і послідовності виконання.	Постійне уточнення рішення через швидкий зворотний зв'язок, зміни вимог і вдосконалення продукту під час роботи.
Характер управління	Командування і контроль; виконання за попередньо визначеним планом, контроль відповідності етапів і результатів.	Лідерство і співпраця; узгодження дій команди, замовника та власника проєкту в процесі виконання.
Комунікація	Формальна комунікація через плани, документацію, процедури, контрольні точки.	Неформальна комунікація через постійний обмін інформацією, уточнення завдань і швидке погодження змін.
Модель розробки	Життєвий цикл із послідовними фазами; перехід до наступного етапу після завершення попереднього.	Еволюційна й ітеративна модель; робота короткими циклами з поступовим уточненням продукту.
Контроль якості	Детальне планування і пізніе масштабне тестування після виконання основних робіт.	Постійний контроль вимог і безперервне тестування протягом виконання проєкту.
Управління відхиленнями	Фіксований обсяг робіт; за відхилень змінюються строки та/або ресурси.	Фіксовані строки й ресурси; за відхилень коригується обсяг робіт після погодження із замовником або власником проєкту.
Структура завдань	Work packages у межах Work Breakdown Structure; поділ проєкту на заздалегідь визначені пакети робіт.	Requirements або features з погляду користувача; поділ роботи на вимоги чи функції продукту.
Часові межі	Фаза завершується після виконання визначеного набору завдань.	Sprint має фіксовану тривалість; кількість виконаних завдань залежить від можливості завершення в межах sprint.

*джерело: складено автором на основі [20]

Після визначення ключових відмінностей між традиційним і гнучким управлінням проєктами доцільно оцінити, які практичні наслідки має застосування кожного підходу. Традиційна методологія забезпечує структурованість, чітку документацію та раннє визначення вимог, однак слабше пристосовується до змін під час виконання проєкту. Гнучка методологія краще працює за змінних вимог, активної участі замовника та потреби в регулярній перевірці результату, проте може створювати проблеми з прогнозуванням строків, документацією (табл. 1.1.2).

Таблиця 1.1.2

Переваги та недоліки традиційних і гнучких методологій управління проєктами*

Підхід	Переваги	Недоліки
Традиційні методології управління проєктами	<ol style="list-style-type: none"> 1. Використання прозорої структури проєкту. 2. Раннє визначення вимог і кінцевої мети. 3. Зручність управління, оскільки кожна фаза має визначені результати та процедуру перегляду. 4. Чітке передання інформації. 5. Формування зрозумілої документації. 6. Розподіл відповідальності між учасниками проєктної команди. 	<ol style="list-style-type: none"> 1. Пізнє тестування впроваджених рішень. 2. Потреба в глибокому аналізі потреб клієнта або кінцевого користувача на початку проєкту. 3. Необхідність точного визначення результатів проєкту до його початку. 4. Жорстке дотримання погоджених графіків. 5. Складність оцінювання прогресу під час реалізації проєкту. 6. Жорстке дотримання фаз проєкту.
Гнучкі методології управління проєктами	<ol style="list-style-type: none"> 1. Орієнтація успіху проєкту на надання бізнес-цінності клієнту або кінцевому користувачу. 2. Відкритість до внесення змін. 3. Постійна комунікація з клієнтом або кінцевим користувачем. 4. Легше вимірювання прогресу реалізації проєкту. 5. Адаптація проєкту до змінних вимог. 6. Регулярна або постійна перевірка прогресу реалізації проєкту. 	<ol style="list-style-type: none"> 1. Недостатньо підготовлена документація. 2. Відсутність чітко структурованих процесів. 3. Нечітке завершення строку реалізації проєкту. 4. Нижча прогнозованість, зокрема складність оцінювання обсягу потрібних робіт. 5. Надання нових функцій за рахунок накопичення технічного боргу. 6. Потреба в постійній співпраці та комунікації з учасниками проєктної команди.

*джерело: складено автором на основі [33, с. 518-521]

Традиційні методології більше відповідають проектам із заздалегідь визначеним обсягом робіт, послідовними етапами та фіксованими вимогами. Гнучкі методології доцільні за умов змінності вимог, потреби в регулярному зворотному зв'язку, поетапному створенні результату та постійному коригуванні пріоритетів

Agile-управління має обмеження у проектах із високими вимогами до формалізації, попереднього планування та регламентованого ухвалення рішень. Його застосування є менш обґрунтованим у сферах, де критичне значення мають стабільність процедур, нормативна визначеність і жорстке погодження етапів робіт, зокрема у будівництві, виробництві, військовій діяльності та охороні здоров'я. За таких умов висока змінність вимог і неповна деталізація плану на початку проекту можуть послаблювати контроль очікуваного результату.

Одним із ключових недоліків Agile є нижча прогнозованість кінцевого продукту. Відсутність остаточного плану на старті проекту може призвести до істотного відхилення фактичного результату від початкового задуму. В ІТ-проектах це проявляється через розширення обсягу робіт, перегляд початкових пріоритетів, зміну функціональних вимог і ускладнення оцінювання строків виконання.

Результативність Agile значною мірою залежить від професійної зрілості команди. Гнучкі методи потребують не лише технічної компетентності, а й управлінських, комунікаційних і соціальних навичок. За недостатнього досвіду учасників проекту виникають проблеми із самоорганізацією, розподілом відповідальності, погодженням рішень і підтриманням узгодженої роботи без постійного зовнішнього контролю.

Вагомим викликом є постійна участь замовника у процесі розробки. Agile передбачає регулярне уточнення вимог, тестування проміжних результатів, погодження функціоналу та швидке надання зворотного зв'язку. За низької залученості клієнта або нечіткості його вимог команда може спрямовувати

ресурси на другорядні функції, неправильно інтерпретувати очікування або втрачати узгодженість щодо цінності продукту.

Гнучке управління підвищує інтенсивність комунікації між учасниками проєкту. Розробники, тестувальники, замовники та інші залучені сторони мають постійно погоджувати завдання, зміни й результати. За збільшення кількості учасників зростає кількість комунікаційних зв'язків, що підвищує ризик непорозумінь, затримок, дублювання дій і втрати узгодженості між членами команди.

Недостатня документація також належить до поширених обмежень Agile. Оскільки вимоги часто деталізуються безпосередньо перед розробкою, опис функцій, рішень і характеристик продукту може бути неповним. Це ускладнює адаптацію нових працівників, знижує прозорість попередньо ухвалених рішень і підвищує ризик різного трактування поведінки системи.

Agile-проєкт може втрачати керованість за слабого початкового планування, нечіткого зворотного зв'язку та нестабільної комунікації. Особливо помітним є ризик *scope creep*, коли обсяг робіт поступово зростає, а продукт постійно змінюється. Для IT-підприємства це означає перевантаження команди, збільшення кількості незавершених завдань, зміщення уваги з пріоритетного функціоналу на другорядні зміни та зростання ризику технічного боргу.

На організаційному рівні впровадження Agile потребує зміни управлінської культури. Командно-контрольний стиль, бюрократичні процедури та високий рівень формалізації процесів обмежують ефективність гнучких методів. Для Agile важливими є співпраця, довіра, участь команди в ухваленні рішень, нижчий рівень управлінського контролю та готовність організації до адаптації процесів [29, с. 39-41].

Принципи Agile Manifesto визначають зміст гнучкого управління IT-проєктами, оскільки встановлюють вимоги до взаємодії команди із замовником, організації роботи, оцінювання прогресу та реагування на зміни:

1. пріоритет задоволення потреб замовника через раннє та регулярне надання робочого результату;
2. готовність до змін вимог навіть на пізніх етапах розробки, якщо це підвищує цінність продукту;
3. часте постачання робочого продукту через короткі проміжки часу;
4. постійна взаємодія між представниками бізнесу та розробниками протягом усього проєкту;
5. виконання проєкту мотивованими фахівцями, які мають необхідні умови, ресурси та довіру;
6. перевага прямої комунікації між учасниками команди як найбільш результативного способу обміну інформацією;
7. оцінювання прогресу за реально працюючим продуктом, а не за обсягом документації чи формальним виконанням процедур;
8. підтримання сталого темпу роботи команди, замовника й користувачів протягом тривалого періоду;
9. постійна увага до технічної якості, архітектури та проєктних рішень;
10. мінімізація зайвих робіт, функцій і процедур, які не створюють цінності для продукту;
11. формування найкращих вимог, архітектурних рішень і способів організації роботи в самоорганізованих командах;
12. регулярний аналіз командою власної роботи з подальшим коригуванням процесів для підвищення ефективності [15].

У межах гнучкого управління проєктами виділяють наступні бізнес-цілі:

- безперервні інновації – постійне оновлення продукту відповідно до поточних вимог замовника, змін ринку та потреб користувачів;
- адаптивність продукту – здатність продукту змінюватися, доповнюватися та масштабуватися відповідно до майбутніх вимог замовника;

- скорочення строків постачання – швидше отримання готових частин продукту, своєчасний вихід на ринок і підвищення окупності інвестицій;
- адаптивність людей і процесів – швидке реагування команди та управлінських процесів на зміни продукту, вимог замовника, технологічних умов і бізнес-середовища;
- надійні результати – отримання стабільного результату проєкту [16, с. 332].

Якщо принципи Agile Manifesto визначають загальні вимоги до роботи команди, взаємодії із замовником і реагування на зміни, то бізнес-цілі показують, який результат має отримати підприємство від застосування гнучкого управління.

Гнучке управління проєктами застосовується не лише у розробці програмного забезпечення, а й в інших проєктах із високою невизначеністю вимог. Для IT-підприємства його значення полягає у скороченні розриву між очікуваннями замовника і фактичним результатом роботи команди. Замість тривалого виконання попередньо затвердженого плану без проміжної перевірки Agile передбачає регулярне уточнення вимог, контроль готового функціоналу, швидке виявлення проблем і зміну пріоритетів до завершення проєкту.

У практиці IT-підприємства це означає перехід від управління окремими етапами до управління цінністю, яку отримує замовник або кінцевий користувач. Важливим стає не лише виконання запланованого переліку робіт, а й своєчасне створення функціоналу, що має прикладне значення для продукту. Саме тому в Agile значну роль відіграють backlog, короткі ітерації, регулярні демонстрації результату, тестування, зворотний зв'язок і перегляд пріоритетів.

Водночас гнучке управління не усуває потреби в плануванні, контролі та відповідальності. Воно змінює їхній зміст: планування стає поетапним, контроль пов'язується з готовністю продукту, а відповідальність розподіляється між командою, власником продукту та замовником. За таких умов ефективність

управління IT-проектом залежить від зрілості команди, якості комунікації, чіткості пріоритетів і здатності швидко реагувати на зміни без втрати керованості проекту.

Отже, гнучке управління проектами в IT-сфері варто розглядати як адаптивну систему організації роботи, що поєднує ітеративну розробку, постійне уточнення вимог, активну участь замовника та орієнтацію на робочий результат. Його перевага проявляється в умовах невизначеності, коли кінцевий продукт не може бути повністю деталізований на початку проекту. Разом із тим результативність Agile залежить від належної командної взаємодії, професійної підготовки учасників, участі замовника та здатності підприємства підтримувати баланс між гнучкістю, якістю, строками й обсягом робіт.

1.2. Методичні підходи до гнучкого управління IT-проектами

Практичне застосування Agile потребує не лише загального розуміння його принципів, а й вибору методичних підходів, за допомогою яких організовується щоденна робота команди. Саме вони визначають, як формуються завдання, як відбувається планування, яким чином контролюється прогрес, як перевіряється якість продукту та як команда реагує на зміну вимог.

Agile не є окремою методологією у вузькому розумінні. Він охоплює сукупність підходів, практик і фреймворків, які відрізняються за призначенням, масштабом використання та умовами застосування в управлінні IT-проектами (табл. 1.2.1).

Scrum сформувався у 1993 році як підхід до управління складними проектами у сфері розробки програмного забезпечення. Його автор – Jeffrey Sutherland. Фреймворк базується на емпіричній моделі управління, відповідно до якої прийняття рішень здійснюється на основі фактичних результатів, регулярного аналізу та коригування процесів.

Agile практики*

Практика	Характеристика та застосування
Scrum	Орієнтований на середні та великі проекти, передбачає ролі Product Owner, Scrum Master і команду розробки; фокус на плануванні, спринтах і ретроспективах.
Extreme Programming (XP)	Підходить для невеликих проектів і команд, акцент на код, тестуванні, парному програмуванні та постійному інтегруванні.
Crystal	Для малих проектів, гнучкий підхід до процесів і документації, підкреслює комунікацію та адаптивність.
Kanban	Візуалізація робочого процесу, обмеження незавершених завдань, постійне удосконалення процесу.
Гібридні підходи	Поєднують традиційні та agile практики, забезпечуючи баланс між структурованим управлінням і гнучкістю; застосовуються у різних галузях, включно з виробництвом, державним сектором і освітою.

*джерело: складено автором на основі [18]

Методологічну основу становлять три принципи: прозорість (відкритість інформації щодо цілей і прогресу), інспекція (систематичний перегляд результатів) та адаптація (коригування діяльності відповідно до змін середовища).

Scrum орієнтований на ітераційне створення продукту через короткі цикли – спринти тривалістю 1–4 тижні. Кожен спринт завершується формуванням інкременту – завершеної функціональної частини продукту, що відповідає встановленим критеріям якості (табл. 1.2.2).

Суттєвою характеристикою Scrum є самоорганізація команди, що передбачає автономний розподіл завдань і відповідальності в межах визначених ролей. Функції класичного менеджера проєкту розподіляються між учасниками команди, що формує децентралізовану модель управління.

Scrum застосовується в умовах високої невизначеності, змінності вимог та технологічної складності. Найпоширеніші сфери використання – ІТ, фінансові технології, інженерія, продуктовий менеджмент, цифрова трансформація бізнесу.

Ключові елементи Scrum*

Категорія	Елемент	Зміст
Ролі	Product Owner	Формування бачення продукту, управління вимогами та їх пріоритетністю
	Scrum Master	Процесна координація, підтримка принципів фреймворку, усунення організаційних бар'єрів
	Development Team	Реалізація завдань спринту та створення інкременту
Події	Sprint Planning	Планування обсягу робіт на ітерацію
	Daily Scrum	Щоденна синхронізація діяльності команди
	Sprint Review	Оцінювання інкременту та отримання зворотного зв'язку
	Sprint Retrospective	Аналіз процесу роботи та визначення напрямів удосконалення
Артефакти	Product Backlog	Упорядкований перелік вимог до продукту
	Sprint Backlog	Перелік завдань, відібраних для виконання у спринті
	Increment	Завершений функціональний результат ітерації

**Джерело: складено автором на основі [36]*

Серед переваг виокремлюють адаптивність, короткі цикли формування результату, прозорість виконання робіт та систематичний зворотний зв'язок. До обмежень належать відсутність формалізованого інструментарію управління ризиками, складність масштабування у великих структурах та залежність ефективності від професійної зрілості команди [36].

Extreme Programming (XP) виникла як відповідь на обмеження традиційної Waterfall-моделі, коли всі вимоги замовника збиралися на початку проекту і реалізація здійснювалася у строгій послідовності. Цей підхід часто призводив до непорозумінь: замовники змінювали або не усвідомлювали своїх вимог, а програмісти під час здачі продукту фактично виконували лише частину передбаченого обсягу. Для вирішення цих проблем з'явилася потреба у методологіях із короткими ітераціями та постійною взаємодією із замовником. XP, розроблена Кентом Беком разом із Ward Cunningham та Ron Jeffries у 1996 році, передбачає невеликі команди (до 12–14 осіб), часті релізи та активну участь замовника на всіх етапах розробки.

Методологія XP характеризується кількома ключовими особливостями. По-перше, команда працює колективно, а відповідальність за продукт розподілена між усіма учасниками. По-друге, XP передбачає постійне залучення замовника у процес розробки, що зменшує ризик помилкового трактування вимог та скорочує час на прийняття рішень. По-третє, метод підтримує короткі ітерації та часті релізи, що дозволяє адаптуватися до змін вимог та оцінювати результати роботи на ранніх етапах. По-четверте, XP мінімізує документацію, оскільки основний обмін інформацією відбувається через пряме спілкування та інтеграцію тестів у процес розробки.

Основні цінності XP включають:

- комунікація – принцип організації роботи, що забезпечує безпосередній обмін інформацією між розробниками та замовником;
- простота – підхід до проектування і реалізації, що зосереджується на виконанні актуальних вимог і мінімізації складності системи;
- зворотний зв'язок – процес отримання інформації про результати попередніх ітерацій для коригування подальшої розробки;
- сміливість – готовність змінювати код і дизайн без страху перед помилками;
- взаємоповага – взаємне ставлення між членами команди та замовником для підвищення ефективності співпраці.

Практики XP інтегровані в робочий процес команди. Представник замовника, який працює безпосередньо в команді, дозволяє швидко уточнювати вимоги та уникати затримок через документовані специфікації. Часті малі релізи забезпечують своєчасне тестування та оцінку роботи, зменшуючи ризик планувальних помилок. Використання метафор спрощує розуміння архітектури системи. Автоматизоване тестування та простий дизайн з мінімальною кількістю класів і функцій полегшують підтримку коду. Рефакторинг забезпечує гнучкість

коду, а парне програмування підвищує якість продукту та сприяє обміну знаннями.

Додаткові принципи організації роботи включають:

- колективна власність коду – модель розподіленої відповідальності, коли будь-який учасник команди може покращувати будь-яку частину коду;
- стандарти кодування – правила форматування та стилю коду, що забезпечують зрозумілість та спрощують взаємодію між розробниками;
- безперервна інтеграція – процес регулярного збирання та перевірки коду для забезпечення постійної працездатності системи;
- контроль робочого часу – обмеження робочого тижня до 40 годин для підтримки продуктивності та якості роботи.

Extreme Programming адаптується до специфіки стартапів через LIRE (Lightweight Process for E-business), який передбачає збір сценаріїв користувачів, перевірку прийнятності функціоналу, реалізацію сценаріїв, рефакторинг системи та виправлення коду. Це дозволяє малим командам швидко реагувати на потреби замовника та скоротити час виходу продукту на ринок.

При застосуванні XP у великих проектах необхідно вводити додаткові механізми: попереднє проектування архітектури, короткі релізи з пошарованою інтеграцією, залучення сурогатного замовника, гнучке парне програмування та стандартизовані інтерфейси для повторного використання коду.

Отже, XP має перелік особливостей, порівняно з іншими методологіями (табл. 1.2.3).

Незважаючи на переваги, XP має обмеження, зокрема:

- фокус на коді – ризик недостатньої уваги до архітектури та реалізації специфікацій;
- необхідність присутності команди – складність застосування у віддалених проектах через потребу безпосередньої взаємодії з замовником;

- мінімальна документація – ускладнення відстеження змін та збільшення ризику помилок;
- високий рівень стресу – підвищене навантаження може впливати на якість продукту.

Таблиця 1.2.3

Порівняння XP з іншими популярними методологіями

Методологія	Відмінності Extreme Programming (XP)
SCRUM	XP має коротші ітерації (1–2 тижні), гнучкі строки, команда строго слідує пріоритетам замовника, готова до негайного застосування без додаткових практик.
Agile (загалом)	XP не має посередників між замовником і командою; проект ділиться на маленькі частини, які можна змінювати в процесі без затримок.
Waterfall	XP ітеративна, з постійною взаємодією з замовником; зміни можна впроваджувати на будь-якому етапі.
Kanban	XP розділяє робочий потік на маленькі релізи; підтримує часту інтеграцію та швидку поставку функціоналу.
Lean	XP дозволяє реалізувати частини проекту не лише за запитом замовника, а й за плануванням коротких циклів; часті релізи та тісна взаємодія з клієнтом.

**Джерело: складено автором на основі [38, с. 9-10]*

Extreme Programming забезпечує високу гнучкість та ефективність командної роботи, особливо для проектів із змінними вимогами та невеликими командами, сприяє зменшенню кількості помилок та підвищенню мотивації учасників розробки.

Agile Kanban виник у виробництві Toyota у 1947 році як візуальна система управління процесами, заснована на принципі Just In Time (JIT), що дозволяє уникати перевантаження учасників процесу та скорочує час очікування завдань, підвищуючи ефективність роботи. У 2004 році Дейвід Дж. Андерсон адаптував Kanban для розробки програмного забезпечення, визначивши його як підхід до інкрементальних і еволюційних змін у процесах та системах організацій, який дозволяє підвищити прозорість і контроль за потоком завдань. Основні принципи методу включають:

- обмеження кількості завдань у роботі (WIP);

- візуалізацію робочого процесу;
- вимірювання та управління потоком завдань;
- чітке визначення політик процесу;
- використання моделей для виявлення можливостей покращення.

Практики Kanban передбачають:

- початок з наявних процесів та ресурсів;
- стимулювання інкрементальних змін;
- повагу до існуючих ролей і процесів;
- заохочення лідерства протягом усього циклу розробки.

Kanban-дошка є ключовим інструментом методу, що дозволяє візуалізувати робочий процес і контролювати кількість завдань на кожному етапі. Завдання представлені у вигляді карток, які переміщуються між колонками відповідно до поточного стану. Вона підвищує прозорість процесу, полегшує координацію роботи команди та зменшує втрати часу через перемикання між завданнями. Дошки можуть бути простими, з трьома стадіями (To Do, Doing, Done), або детальними, включаючи додаткові етапи, наприклад Backlog, Analysis, Development, Test, Deployment.

Впровадження Kanban забезпечує підвищення видимості та розуміння всього процесу розробки, ефективне контролювання потоку завдань, гнучкість та надійність, покращення комунікації та прозорості між учасниками команди та замовниками. Метод дозволяє:

- зменшувати час проходження завдань (lead-time);
- підвищувати продуктивність розробки програмного забезпечення;
- підтримувати безперервну доставку продукту.

На відміну від Scrum, Kanban не використовує фіксовані ітерації, а працює на основі pull-системи, що забезпечує постійну адаптацію до змін та контроль за потоком роботи.

Незважаючи на численні переваги, впровадження Kanban супроводжується низкою викликів. Основними проблемами є:

- визначення оптимальних WIP-лімітів для етапів дошки;
- складність візуалізації корисної інформації про стан проекту;
- необхідність висококваліфікованих учасників із перекривними навичками;
- ризик виникнення затримки всього процесу через локальні збої.

Для підвищення ефективності метод часто інтегрують із іншими підходами, такими як Scrumban, Lean або Earned Value Analysis, що дозволяє поєднувати переваги кількох методів і компенсувати обмеження Kanban [11].

Lean Agile характеризується високою гнучкістю як методологія у порівнянні зі Scrum або XP, оскільки має мінімальні обмеження щодо правил, методів та інструкцій. Основна мета Lean полягає у зменшенні втрат у процесі розробки програмного забезпечення та підвищенні цінності продукту у максимально короткий проміжок часу за рахунок усунення непотрібних дій під час розробки. На відміну від XP, яка фокусується на взаємодії клієнта та розробника, Lean визначає принципи та цінності для вищого керівництва організації та спрямована на оптимізацію всього бізнес-процесу, дотримуючись підходу «зверху вниз».

Сутність Lean Agile базується на семи основних цінностях:

- усунення всього, що не додає додаткової цінності продукту;
- навчання на основі кожної ітерації та зворотного зв'язку від користувачів;
- прийняття рішень у пізніших стадіях розробки на підставі максимальної наявної інформації;
- швидка доставка продукту;
- створення умов, у яких розробники можуть ефективно працювати;
- орієнтація на потреби клієнта;

- розуміння загальної картини бізнесу та інтеграції продукту у бізнес-процес.

Lean Agile оптимізує процеси головним чином через використання спеціалізованих інструментів і мінімальні правила, що дозволяє організаціям швидко адаптуватися до змін та концентрувати ресурси на створенні цінності.

Впровадження Lean Agile впливає на якість продукту та процесів розробки. Якість продукту визначається задоволенням потреб користувача, відповідністю вимогам та ефективністю функціонування, тоді як якість процесу визначає, наскільки методи розробки забезпечують створення продукту високої якості. Agile-підхід дозволяє ітераційно виконувати всі фази розробки, отримуючи зворотний зв'язок після кожного спринту. Це забезпечує швидку адаптацію до змін, підтримку високої якості продукту та задоволення потреб клієнта.

Під час розробки в Lean Agile оцінюються такі атрибути якості:

- у фазі збору вимог – функціональність, надійність, масштабованість, підтримуваність;
- у фазі проектування – ефективність, повторне використання, портативність, гнучкість;
- у фазі тестування та впровадження – надійність, продуктивність, усунення дефектів та підтримка ефективності після релізу.

Метрики оцінки якості включають функціональні (кількість прийнятих юзер-історій, співвідношення дефектів у тестуванні та виробництві, покриття коду) та нефункціональні (портативність, безпека, надійність, продуктивність, юзабіліті), що дозволяють передбачати можливі проблеми та підтримувати високий рівень якості продукту. Юзер-історії є ключовим інструментом визначення якості, оскільки вони описують дії користувача і допомагають точно формулювати вимоги до системи.

Водночас організації можуть зіткнутися з певними проблемами при впровадженні Lean Agile, включаючи культурні бар'єри, недостатність вимог,

проблеми комунікації в команді, відсутність належної підготовки та опору змінам. Усунення цих перешкод є критично важливим для успішного застосування Lean Agile та досягнення високої якості продукту і задоволеності клієнта [46, с. 76-77].

Отже, інтеграція Agile у проектні процеси формує прозору систему управління, де роль менеджера трансформується у фасилітатора та координатора, а автономія команд стимулює швидке реагування на зміни й безперервне вдосконалення продукту. Одночасно підвищується значення культури співпраці, залучення замовника та ціннісної орієнтації результатів, що впливає на якість продукту та процесів його створення.

Розширене застосування Agile поза межами ІТ-сфери демонструє його потенціал у підвищенні продуктивності та гнучкості організацій, проте передбачає вирішення викликів, пов'язаних із масштабуванням, навчанням персоналу та інтеграцією змін у існуючі структури. Таким чином, Agile виступає не лише як набір практик, а як цілісний підхід до управління, який поєднує технічні, організаційні та соціальні аспекти ефективної реалізації проектів.

РОЗДІЛ 2. АНАЛІЗ ГНУЧКОГО УПРАВЛІННЯ ПРОЄКТАМИ У КОМПАНІЇ JUSTMARKETS TECH

2.1. Організаційна характеристика системи управління IT-проєктами

JustMarkets – глобальний мультиактивний брокер, заснований у 2012 році, який обслуговує клієнтів більш ніж у 160 країнах. Основна мета компанії полягає у створенні зручного та прозорого торгового середовища, що дозволяє користувачам реалізувати свій інвестиційний потенціал. Компанія прагне забезпечувати високу якість виконання угод та професійне обслуговування клієнтів [56].

Стратегічна візія компанії полягає у тому, щоб стати найбільш орієнтованим на клієнта брокером у світі, забезпечуючи простоту та прозорість у процесі торгівлі та інвестування. Основу корпоративної культури формують цінності, які визначають підхід до всіх процесів та рішень, гарантують відповідність дій компанії її принципам і забезпечують узгодженість у роботі команд.

Команду JustMarkets характеризує динамічність та інноваційність. Співробітники з понад 10 країн створюють мультикультурне середовище, яке сприяє ефективній співпраці та обміну знаннями. Це дає змогу компанії адаптуватися до змін на ринку та впроваджувати сучасні управлінські та бізнес-підходи, що підтримують стійкий розвиток організації та професійне зростання персоналу.

Серед ключових аспектів корпоративного середовища варто виділити:

- надійність і довіра як всередині компанії, так і у взаємодії з клієнтами та партнерами;
- відкриту комунікацію між командами, менеджментом та C-level керівниками, що сприяє обміну ідеями та своєчасному вирішенню проблем;

– можливість отримати практичний досвід роботи у мультикультурному середовищі, об'єднаному спільними цілями та продуктом компанії [57].

JustMarkets Tech є компанією, що спеціалізується на розробці фінтех-рішень і працює в умовах високої динаміки ринку інформаційних технологій. Успішність компанії значною мірою залежить від здатності швидко реагувати на зміни вимог користувачів та забезпечувати своєчасне впровадження нових функціональних можливостей у програмні продукти. Для підвищення гнучкості процесів розробки та оптимізації ресурсів JustMarkets Tech використовує сучасні гнучкі методології управління розробкою, зокрема Scrum, Kanban та Lean.

У практиці компанії впровадження Agile-підходів забезпечує короткі ітерації роботи, регулярний зворотний зв'язок із користувачами та постійне вдосконалення процесів, що дозволяє підвищити якість продуктів і скоротити час виходу на ринок. Особлива увага приділяється оптимізації командної взаємодії, прозорості процесів та пріоритету клієнтських потреб, що відображається на стабільності та конкурентоспроможності продуктів JustMarkets Tech.

Одним із ключових аспектів підвищення ефективності розробки є впровадження практик DevOps. Важливість DevOps у процесі розробки програмних продуктів JustMarkets Tech полягає у забезпеченні безперервної інтеграції та доставки, що є ключовим фактором прискорення виходу нових функцій на ринок. DevOps інтегрує процеси розробки та експлуатації програмного забезпечення, створюючи середовище, де команди розробників і системних адміністраторів працюють спільно, координуючи зміни та автоматизуючи повторювані завдання.

DevOps – поєднання розробки (development) та експлуатації (operations), культурно-організаційний підхід, який інтегрує діяльність команд розробки програмного забезпечення та ІТ-операцій. У досліджуваній компанії він включає принципи, практики та технологічні рішення для оптимізації життєвого циклу

програмних продуктів. Основні цілі – скорочення часу розробки, автоматизація повторюваних процесів і підвищення рівня співпраці між командами.

У фінтех-секторі DevOps критично важливий через високу швидкість оновлень, вимоги до надійності систем та відповідності нормативним стандартам. Практики DevOps забезпечують безперервний цикл поставки програмного забезпечення, прискорюючи вихід продуктів на ринок і зменшуючи ймовірність помилок під час інтеграції та розгортання змін.

Основні результати впровадження DevOps:

- скорочення часу від розробки коду до розгортання завдяки автоматизації безперервної інтеграції та доставки (CI/CD).
- підвищення стабільності та якості продукту за рахунок автоматизованого тестування та моніторингу;
- зниження навантаження на ІТ-операційні команди та ризиків, пов'язаних з ручним виконанням процесів;
- можливість швидкого реагування на зміни ринкових умов, запити користувачів і нормативні вимоги, що забезпечує конкурентну перевагу [31, с. 12- 13].

DevOps об'єднує практики, культурні цінності та інструменти, спрямовані на інтеграцію розробки програмного забезпечення та ІТ-операцій, усунення бар'єрів і підвищення ефективності комунікації та співпраці. Термін походить від поєднання development (розробники, тестувальники, забезпечення якості) та operations (системні адміністратори, адміністратори баз даних, мережеві спеціалісти), підкреслюючи спільну відповідальність між командами.

Основу DevOps складають три компоненти:

- культура – спільні цінності, командне мислення та стандарти взаємодії;
- практики – реалізація безперервної інтеграції, автоматизованого тестування та деплою;

– інструменти – підтримка автоматизації, моніторингу та ефективної комунікації [27].

Методології Agile доповнюють DevOps, забезпечуючи ітеративне оновлення програмного забезпечення та дозволяючи бізнесу швидко реагувати на потреби ринку. Модель CAMS – Culture, Automation, Measurement, Sharing – відображає ключові цінності, що визначають впровадження DevOps. Проте успішна реалізація значною мірою залежить від того, наскільки організаційна культура відповідає практичним підходам.

На підприємстві формуються крос-функціональні команди, які включають розробників, тестувальників та фахівців з ІТ-операцій. Це забезпечує:

- безперервний обмін інформацією;
- швидке узгодження змін;
- оперативне реагування на потреби клієнтів та ринкові вимоги.

Важливою організаційною складовою є єдина культура відповідальності за якість продукту, яка охоплює весь життєвий цикл програмного забезпечення. Детальніше DevOps практики зображено у табл. 2.1.1.

Таблиця 2.1.1

DevOps практики у JustMarkets Tech

Практика	Функціональне призначення	Використані технології та інструменти
CI/CD (Continuous Integration/Continuous Delivery)	Автоматизація процесів інтеграції коду, збірки та розгортання програмного забезпечення, забезпечення безперервної доставки змін у продуктивне середовище	Jenkins, GitLab CI, Docker
Конфігураційне управління	Централізоване управління версіями конфігурацій, контроль стану серверів та середовищ, стандартизація розгортань	Ansible, Puppet, Chef
Infrastructure as Code (IaC)	Опис і автоматичне створення інфраструктури у вигляді коду, мінімізація людських помилок, забезпечення повторюваності та безпеки розгортань	Terraform, AWS CloudFormation

Моніторинг та логування	Постійне відстеження продуктивності систем, аналіз логів, раннє виявлення збоїв та відхилень у роботі сервісів	Prometheus, Grafana, ELK Stack
Автоматизоване тестування	Виконання функціональних, інтеграційних і безпекових тестів на кожному етапі життєвого циклу, скорочення часу перевірки змін	Selenium, JUnit, SonarQube
Хмарна інфраструктура та масштабування	Використання хмарних платформ для забезпечення доступності, масштабованості та відмовостійкості сервісів	AWS, Azure, Kubernetes

Складено автором на основі: [50]

Організаційна структура JustMarkets Tech має функціонально-проектний характер. У компанії виокремлені сталі функціональні напрями (рис. 2.1.1), зокрема Engineering, Product, Back Office, Customer Support, Business Development, Marketing, Finance & Accounting, Legal & Compliance, HR & Administration. Водночас реалізація IT-проектів не обмежується межами одного підрозділу, оскільки розробка та розвиток цифрових продуктів потребують постійної взаємодії продуктових, технічних, операційних і бізнес-команд.



Рисунок 2.1.1. Функціональна структура JustMarkets Tech

Джерело: розроблено автором на основі даних підприємства

Управління IT-проектами зосереджується навколо взаємодії Product та Engineering. Product-напряму формує вимоги до продукту, визначає пріоритети, узгоджує очікуваний результат із бізнес-цілями та потребами користувачів. Engineering відповідає за технічну реалізацію, тестування, підтримку стабільності сервісів і впровадження змін у цифрову платформу. Саме між цими

напрямами формується основний цикл проектної роботи: постановка завдань, оцінювання складності, планування виконання, розробка, перевірка результату та передання готового функціоналу в експлуатацію.

Організація проектного управління в JustMarkets Tech має міжфункціональний характер. До реалізації проектів, окрім Product та Engineering, залучаються Back Office, Customer Support, Legal & Compliance, Marketing і Business Development. Customer Support передає інформацію про клієнтські запити та проблеми користувачів, Back Office враховує операційні вимоги, Legal & Compliance перевіряє відповідність рішень регуляторним обмеженням, а Marketing і Business Development формують ринкові пріоритети та вимоги до розвитку продукту (табл. 2.1.2).

Таблиця 2.1.2

Напрями управління IT-проектами JustMarkets Tech

Напря́м	Роль у проектному управлінні
Продуктовий напрям (Product)	Формування бачення продукту, дорожньої карти розвитку, вимог, пріоритетів і показників результативності
Інженерний напрям (Engineering)	Технічна реалізація вимог, оцінювання складності, розробка, тестування, контроль якості, підготовка релізів
Торговий напрям (Trading)	Визначення вимог до торгових функцій, швидкості операцій, стабільності торгових процесів і ризик-інструментів
Клієнтська підтримка (Customer Support)	Передача інформації про звернення, проблеми, очікування та відгуки користувачів
Операційний супровід (Back Office)	Уточнення вимог до верифікації, фінансових операцій і внутрішніх процедур
Юридичний супровід і відповідність вимогам (Legal & Compliance)	Перевірка продуктових змін щодо фінансового регулювання, безпеки та вимог різних ринків
Маркетинг і розвиток бізнесу (Marketing, Business Development)	Формування ринкових, партнерських і регіональних запитів до продукту

Джерело: складено автором на основі [57]

Ядром управління IT-проектами виступає взаємодія продуктового та інженерного напрямів. Продуктовий напрям визначає зміст майбутніх змін, їхню

пріоритетність, очікуваний результат для користувача і бізнесу. Інженерний напрям забезпечує технічне виконання, якість, стабільність і своєчасний запуск змін. У вакансії Engineering Manager зазначено управління міжфункціональними інженерними командами, до складу яких входять backend, frontend, infrastructure та mobile-фахівці, а також співпрацю з Product, Trading і senior engineering leadership для перетворення бізнес-вимог у технічне виконання.

Проектна робота має циклічний характер. Спочатку формується потреба у зміні продукту. Джерелами виступають ринкові тенденції, поведінка користувачів, клієнтський зворотний зв'язок, технічні показники, вимоги регулювання або бізнес-пріоритети. Далі продуктовий напрям перетворює цю потребу на вимогу, визначає її місце у дорожній карті розвитку (roadmap) або переліку вимог і завдань (backlog), оцінює очікуваний результат і погоджує зміну із суміжними командами. Після цього інженерний напрям проводить технічне оцінювання, організовує розробку, тестування, перегляд коду, безперервну інтеграцію та постачання (CI/CD), управління релізами (release management) і аналіз результату після запуску.

Таблиця 2.1.3

Цикл гнучкого управління IT-проектами у JustMarkets Tech

Етап	Зміст управлінської дії	Основні учасники
Виявлення потреби	Фіксація ринкового запиту, проблеми користувача, технічного обмеження або регуляторної вимоги	Product, Customer Support, Trading, Marketing, Legal & Compliance
Формування вимоги	Опис зміни, визначення пріоритету, очікуваного результату та місця у roadmap або backlog	Product
Узгодження	Перевірка вимоги з позицій ринку, клієнтського досвіду, операційних процедур і регулювання	Product, Trading, Customer Support, Back Office, Legal & Compliance, Marketing
Технічне оцінювання	Визначення складності, залежностей, ризиків, ресурсів і впливу на платформу	Engineering
Реалізація	Розробка, тестування, перегляд коду, CI/CD, підготовка релізу	Engineering
Оцінювання результату	Аналіз технічних і продуктових показників після запуску	Product, Engineering, Trading, Customer Support

Джерело: складено автором

Найбільш значущу групу становлять проекти розвитку торгової платформи. Вони охоплюють торгові продукти, торгові функції, торгові сценарії, розширені графіки, алгоритмічну торгівлю, інструменти управління ризиками, стабільність платформи та швидкість виконання торгових операцій. Product Manager Trading відповідає за дорожню карту торгових продуктів, покращення функцій на основі клієнтського зворотного зв'язку й ринкових тенденцій, оптимізацію платформи за показниками продуктивності, зручності та надійності. Для оцінювання результату використовуються доступність платформи, швидкість виконання торгових операцій і задоволеність користувачів.

Мобільні IT-проекти пов'язані з розвитком мобільного торгового продукту. До їхнього змісту належать зручність застосунку, швидкість роботи, залученість користувачів, утримання клієнтів, конверсія, локалізація, мовна підтримка і регіональні платіжні рішення. Product Manager Mobile відповідає за бачення, стратегію і дорожню карту мобільного продукту, працює з ринковими тенденціями, конкурентними пропозиціями, новими мобільними технологіями та відгуками користувачів. У межах мобільних проектів передбачена співпраця з інженерними командами, маркетингом, клієнтською підтримкою, compliance-командами, юридичним напрямом та іншими зацікавленими сторонами.

Інженерні проекти спрямовані на технічну стійкість, продуктивність і масштабованість платформи. До них належать планування, оцінювання, CI/CD, перегляд коду, тестування, управління релізами, спостережуваність систем, продуктивність, масштабованість і зменшення технічного боргу. Технічний борг означає накопичені недоліки або тимчасові технічні рішення, які ускладнюють подальший розвиток продукту. Для брокерської платформи цей напрям має ключове значення, оскільки стабільність і швидкість сервісу безпосередньо впливають на якість користувацького досвіду.

Клієнтська підтримка виконує роль джерела інформації для продуктових змін. JustMarkets позиціонує клієнта як основну цінність, наголошує на

врахуванні кожного отриманого відгуку і наявності багатомовної підтримки 24/7. Дані клієнтської підтримки переходять у продуктовий напрям у вигляді повторюваних проблем, запитів, скарг або очікувань користувачів. Після опрацювання вони впливають на пріоритети у backlog, зміст наступних змін і показники оцінювання результату.

Юридичний напрям і compliance залучаються до проєктів, пов'язаних із торговими продуктами, мобільними функціями, фінансовими операціями, клієнтською верифікацією, безпекою та регіональними вимогами. У мобільному напрямі прямо зазначено відповідність фінансовому регулюванню та стандартам безпеки, а також співпрацю з legal і compliance-командами для впровадження необхідних оновлень. Через це регуляторна перевірка входить до процесу погодження продуктових змін ще до їхньої технічної реалізації.

Маркетинг і розвиток бізнесу формують ринкові та партнерські вимоги до продукту. Їхня участь важлива для проєктів, пов'язаних із виходом на нові ринки, локалізацією, залученням клієнтів, партнерськими програмами та адаптацією сервісів до регіональних умов. Продуктовий напрям узгоджує ці запити з очікуваною користю для користувача і бізнесу, після чого інженерний напрям оцінює технічну складність і виконує розробку.

На підприємстві простежуються такі основні види ІТ-проєктів:

- проєкти розвитку торгової платформи, спрямовані на вдосконалення торгових функцій, торгових сценаріїв, інструментів графіків, алгоритмічної торгівлі, управління ризиками, стабільності платформи та швидкості виконання торгових операцій;

- проєкти розвитку мобільного торгового продукту, пов'язані з покращенням мобільного застосунку, зручності користування, торгового досвіду, сповіщень, поповнення рахунку, локалізації, мовної підтримки та регіональних платіжних рішень;

- інженерні проекти стабільності, продуктивності та масштабування платформи, орієнтовані на надійність сервісів, швидкість роботи систем, технічний моніторинг, автоматизацію, тестування, управління релізами та зменшення технічного боргу;
- проекти покращення клієнтського досвіду, спрямовані на аналіз відгуків користувачів, підвищення зручності сервісу, задоволеності клієнтів, залученості, утримання, конверсії та якості взаємодії з цифровими продуктами;
- проекти регуляторної відповідності та безпеки, пов'язані з узгодженням торгових і мобільних продуктів із фінансовим регулюванням, вимогами різних ринків, стандартами безпеки та внутрішніми процедурами compliance;
- проекти ринкової та регіональної адаптації продукту, спрямовані на локалізацію продуктів, мовну підтримку, адаптацію мобільних сервісів до нових ринків і врахування регіональних платіжних методів.

2.2. Діагностика ефективності застосування гнучких методів управління проектами

Після визначення організаційної побудови управління IT-проектами у JustMarkets Tech доцільно перейти до оцінювання того, наскільки сформована система відповідає вимогам гнучкого управління. Встановлено, що проектна робота компанії зосереджена навколо взаємодії Product та Engineering, а до погодження змін залучаються Trading, Customer Support, Back Office, Legal & Compliance, Marketing і Business Development. Тому діагностика має охоплювати не лише технічну реалізацію завдань, а й якість формування вимог, швидкість погодження змін, релізний контроль і фактичний результат для користувача.

Для технічного блоку доцільно враховувати DORA-метрики, які охоплюють частоту розгортань, час проходження зміни до запуску, частку невдалих змін і час відновлення після збою.

Діагностика ефективності гнучкого управління в JustMarkets Tech має будуватися за проектним циклом: від появи продуктової потреби до оцінювання результату після релізу. Це дає змогу відокремити проблеми планування від проблем реалізації, а технічні обмеження, від проблем міжкомандної взаємодії.

Таблиця 2.2.1

Етапи діагностики ефективності гнучкого управління ІТ-проектами
JustMarkets Tech

Етап діагностики	Зміст оцінювання	Характеристика
Формування продуктової потреби	Визначення джерела зміни: клієнтський запит, ринок, торговий процес, технічна проблема, регуляторна вимога	Показує, чи проєкт починається з реальної потреби продукту або користувача
Опис і пріоритезація вимоги	Перевірка якості вимоги, її місця у дорожній карті розвитку та переліку завдань	Визначає, чи першочергово виконуються зміни з найбільшим впливом на продукт
Погодження між командами	Оцінювання участі Product, Engineering, Trading, Customer Support, Back Office, Legal & Compliance, Marketing і Business Development	Показує, чи враховані технічні, клієнтські, операційні, ринкові та регуляторні обмеження
Технічне оцінювання	Аналіз складності, залежностей, ризиків, ресурсів і впливу зміни на платформу	Знижує ризик нестабільного релізу або перевантаження команди
Реалізація і релізний контроль	Оцінювання розробки, тестування, перегляду коду, безперервної інтеграції та управління релізами	Характеризує технічну дисципліну виконання проєкту
Оцінювання після запуску	Аналіз продуктових і технічних показників після релізу	Показує фактичний ефект зміни для користувача, продукту й платформи
Коригування подальших дій	Перегляд пріоритетів після отримання результатів	Забезпечує повторюваний цикл розвитку продукту

Джерело: складено автором

На першому етапі оцінюється якість формування продуктової потреби. Для JustMarkets Tech джерелами змін виступають клієнтські звернення, поведінка

користувачів, вимоги торгової платформи, ринкові зміни, регіональна адаптація, технічні обмеження та вимоги фінансового регулювання. Ефективність цього етапу залежить від того, наскільки швидко потреба потрапляє до Product і перетворюється на зрозумілу вимогу. Недостатня якість вхідної інформації призводить до нечітких завдань, зайвих погоджень, повторної розробки та затримок у релізі.

Другий етап пов'язаний з оцінюванням пріоритезації. Для гнучкого управління важливо, щоб перелік завдань не був простим накопиченням запитів від різних команд. Його зміст має відображати цінність для користувача, вплив на бізнес-показники, технічну складність, регуляторні обмеження та ризик накопичення технічного боргу. У JustMarkets Tech це особливо важливо для торгової платформи й мобільного продукту, оскільки пріоритети одночасно залежать від швидкості операцій, стабільності сервісу, зручності застосунку, клієнтського досвіду та вимог різних ринків.

Третій етап охоплює міжфункціональне погодження. У межах IT-проектів компанії Product визначає зміст зміни, Engineering оцінює технічне виконання, Trading уточнює вимоги до торгових сценаріїв, Customer Support передає повторювані проблеми користувачів, Back Office перевіряє операційні наслідки, Legal & Compliance оцінює регуляторні ризики, Marketing і Business Development формують ринкові та партнерські запити. Ефективність погодження проявляється в тому, що вимога потрапляє в розробку без істотних суперечностей між підрозділами та без подальшого повернення через пропущені обмеження.

Четвертий етап стосується технічного оцінювання. Тут аналізується, наскільки Engineering визначає складність реалізації, залежності між системами, потребу в тестуванні, вплив на продуктивність, безпеку й стабільність платформи. Для брокерської платформи цей етап має ключове значення, оскільки помилка в технічному оцінюванні впливає не лише на строк виконання, а й на доступність сервісу, якість торгових операцій і навантаження на підтримку.

П'ятий етап охоплює реалізацію та релізний контроль. До основних параметрів належать якість тестування, перегляд коду, використання безперервної інтеграції та постачання змін, підготовка релізу, фіксація помилок і контроль впливу зміни на роботу платформи. DORA-метрики доцільні саме на цьому етапі, оскільки вони оцінюють швидкість і стабільність технічного постачання: як часто команда випускає зміни, скільки часу проходить від готовності зміни до запуску, яка частка релізів спричиняє проблеми та як швидко сервіс відновлюється після збою.

Шостий етап пов'язаний з оцінюванням результату після запуску. Для JustMarkets Tech результат релізу має визначатися не лише фактом завершення завдання. Важливими є доступність платформи, швидкість виконання торгових операцій, кількість звернень до клієнтської підтримки, залученість користувачів, утримання, конверсія, задоволеність клієнтів і стабільність роботи мобільного застосунку. Якщо після запуску показники не покращуються або з'являються повторювані проблеми, завдання потребує повторного аналізу й доопрацювання.

Сьомий етап завершує діагностичний цикл. Його зміст полягає у використанні результатів після релізу для перегляду дорожньої карти розвитку, переліку завдань і порядку погодження змін. У гнучкому управлінні значення має не лише швидкий запуск функціоналу, а й здатність команди коригувати подальші дії на основі отриманих даних. Саме цей елемент відрізняє формальне виконання завдань від реального ітераційного розвитку продукту.

Для оцінювання ефективності доцільно згрупувати критерії за управлінськими блоками (табл. 2.2.2). Наведені критерії охоплюють внутрішню та зовнішню ефективність управління. Внутрішня ефективність характеризує якість процесу: наскільки швидко формуються й погоджуються вимоги, чи не перевантажені команди, чи є контроль технічної якості, чи не накопичуються заблоковані завдання. Зовнішня ефективність показує вплив змін на продукт: чи

покращується стабільність платформи, швидкість операцій, клієнтський досвід, залученість і задоволеність користувачів.

Таблиця 2.2.2

Критерії діагностики ефективності застосування гнучких методів управління проєктами

Блок оцінювання	Критерії діагностики
Продуктове планування	наявність дорожньої карти розвитку; актуальність переліку завдань; зв'язок вимог із потребами користувачів; зв'язок вимог із бізнес-цілями; регулярність перегляду пріоритетів
Пріоритезація	зрозумілість критеріїв відбору завдань; врахування користувачької цінності; врахування технічної складності; врахування регуляторних обмежень; баланс між новими функціями і технічним боргом
Міжфункціональна взаємодія	участь продуктових та інженерних команд у плануванні; залучення торгового напряму до змін платформи; участь Legal & Compliance у регуляторно чутливих змінах; використання інформації від Customer Support; участь Back Office в операційних змінах
Потік завдань	середній час проходження завдання; кількість заблокованих завдань; кількість повернень на доопрацювання; рівномірність навантаження; обсяг незавершеної роботи
Технічне виконання	якість технічного оцінювання; наявність тестування; перегляд коду; використання CI/CD; контроль релізів; стабільність після запуску
Результативність релізів	частота релізів; час проходження зміни до запуску; частка невдалих змін; час відновлення після інциденту; кількість помилок після запуску
Клієнтський результат	задоволеність користувачів; кількість повторюваних звернень; залученість; утримання; конверсія; якість взаємодії з продуктом
Подальше вдосконалення	аналіз результатів після релізу; оновлення переліку завдань; коригування дорожньої карти; фіксація проблем процесу; управлінські рішення за підсумками оцінювання

Джерело: складено автором

Оцінювання варто проводити за п'ятибальною шкалою. Вона підходить для порівняння різних блоків управління між собою та подальшого виявлення слабких зон. Для кожного критерію бал має відображати не формальну наявність процесу, а якість його виконання.

Форма діагностичного оцінювання гнучкого управління ІТ-проектами

JustMarkets Tech

Блок оцінювання	Ключове діагностичне питання
Продуктове планування	Чи пов'язані вимоги з потребами користувачів, ринку, торгової платформи та бізнес-цілями
Пріоритезація	Чи потрапляють у розробку завдання з найбільшою продуктовою та бізнес-цінністю
Міжфункціональна взаємодія	Чи узгоджуються зміни між технічними, продуктовими, операційними, ринковими та compliance-командами
Потік завдань	Чи проходять завдання шлях від постановки до релізу без надмірних затримок і повернень
Технічне виконання	Чи забезпечені тестування, перегляд коду, CI/CD, релізний контроль і стабільність після запуску
Результативність релізів	Чи випуски змін є регулярними, контрольованими та не створюють критичних збоїв
Клієнтський результат	Чи впливають зміни на доступність, швидкість, зручність, утримання, конверсію і задоволеність користувачів
Подальше вдосконалення	Чи використовуються результати після релізу для перегляду пріоритетів і процесу управління

Джерело: складено автором

Практична діагностика за цією формою має показати не лише рівень застосування гнучких методів, а й характер проблем у проектному управлінні. Низькі оцінки за продуктовим плануванням свідчатимуть про слабкий зв'язок між вимогами та потребами користувачів. Низькі оцінки за міжфункціональною взаємодією вказуватимуть на затримки в погодженні змін або неповне врахування регуляторних, операційних чи ринкових обмежень. Низькі оцінки за технічним виконанням означатимуть ризики для стабільності платформи, якості релізів і накопичення технічного боргу. Низькі оцінки за клієнтським результатом показуватимуть слабкий вплив змін на практичний досвід користувача.

Для JustMarkets Tech найбільш важливими блоками діагностики є пріоритезація, міжфункціональна взаємодія, технічне виконання та клієнтський результат. Це пов'язано зі специфікою брокерської платформи, де будь-яка зміна

має одночасно відповідати вимогам торгового процесу, стабільності сервісу, регуляторного контролю та очікуванням користувачів. Саме тому ефективність гнучкого управління варто оцінювати не за кількістю виконаних завдань, а за здатністю команд швидко погоджувати важливі зміни, якісно їх реалізовувати, безпечно запускати та перевіряти фактичний вплив на продукт. Детальний розрахунок за критеріями наведено у додатку Б.

Таблиця 2.2.4

Розрахунок інтегральної зваженої оцінки ефективності застосування гнучких методів управління проєктами у JustMarkets Tech*

Критерій	Бал	Вага	Зважений бал
Продуктове планування	3,6	0,14	0,504
Пріоритезація	3,4	0,13	0,442
Міжфункціональна взаємодія	3,6	0,13	0,468
Потік завдань	2,8	0,10	0,280
Технічне виконання	3,8	0,16	0,608
Результативність релізів	2,8	0,11	0,308
Клієнтський результат	3,5	0,15	0,525
Подальше вдосконалення	3,0	0,08	0,240
Разом		1,00	3,375

*Джерело: складено автором

За результатами розрахунку інтегральна зважена оцінка становить 3,375 бала, або 3,4 бала з 5.

Найвищий бал отримало технічне виконання, 3,8, що свідчить про сильнішу позицію інженерної частини проєктного управління: розробки, тестування, контролю якості та підготовки релізів. Достатньо високі оцінки мають продуктове планування, 3,6, міжфункціональна взаємодія, 3,6, і клієнтський результат, 3,5. Це показує, що зміни у проєктах пов'язані з розвитком продукту, потребами користувачів і погодженням між ключовими командами.

Найнижчі оцінки мають потік завдань, 2,8, і результативність релізів, 2,8. Це вказує на слабші місця в контролі часу проходження завдань, заблокованих робіт, повернень на доопрацювання, частоти релізів, післярелізних помилок і

часу відновлення після інцидентів. Подальше вдосконалення, 3,0, також потребує посилення через регулярний аналіз результатів після запуску та оновлення backlog і дорожньої карти.

Найсильнішими елементами гнучкого управління проектами у JustMarkets Tech є технічне виконання, продуктове планування, міжфункціональна взаємодія та орієнтація на клієнтський результат. Основні проблемні зони зосереджені у прозорості потоку завдань, релізних метриках, післярелізного аналізу та контролі технічного боргу.

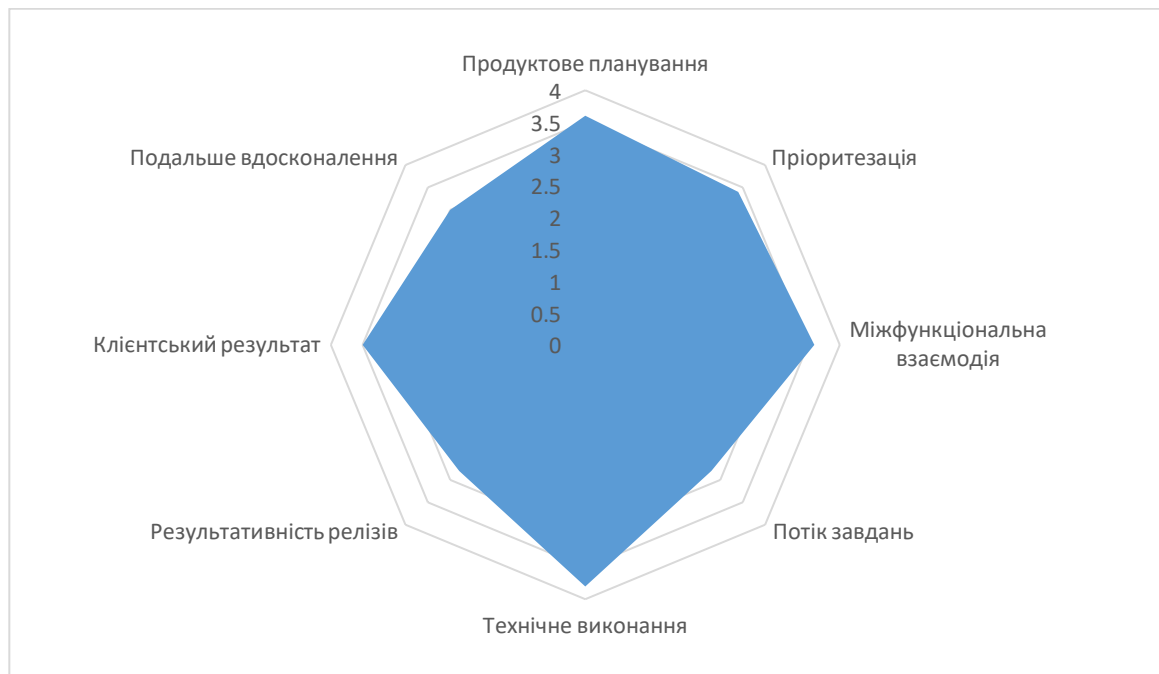


Рисунок 2.2.1. Багатокутник оцінювання ефективності застосування гнучких методів управління проектами у JustMarkets Tech

Джерело: складено автором

Отже, діагностика ефективності застосування гнучких методів управління проектами у JustMarkets Tech має охоплювати повний цикл роботи над змінами: формування потреби, опис вимоги, пріоритизацію, погодження між командами, технічне оцінювання, реалізацію, релізний контроль, оцінювання результату й

подальше вдосконалення. Така система оцінювання поєднує продуктиві, організаційні, технічні та клієнтські критерії, що відповідає характеру проектної діяльності компанії. Результати діагностики надалі стануть підґрунтям для визначення проблемних зон і формування напрямів удосконалення системи гнучкого управління проектами JustMarkets Tech.

РОЗДІЛ 3. УДОСКОНАЛЕННЯ СИСТЕМИ ГНУЧКОГО УПРАВЛІННЯ ПРОЄКТАМИ JUSTMARKETS TECH

3.1. Напрями оптимізації процесів гнучкого управління проєктами у JustMarkets Tech

Результати діагностики показали, що система гнучкого управління проєктами у JustMarkets Tech має сформовану продуктово-інженерну основу, однак потребує доопрацювання в частині прозорості потоку завдань, релізних метрик, післярелізного аналізу та контролю технічного боргу. Інтегральна зважена оцінка ефективності становить 3,375 бала з 5. Найнижчі оцінки отримали потік завдань і результативність релізів, по 2,8 бала, що вказує на потребу в посиленні кількісного контролю проходження задач, блокувань, повернень на доопрацювання, частоти релізів і післярелізних помилок. У попередньому розділі визначено, що управління IT-проєктами в JustMarkets Tech побудоване навколо взаємодії Product та Engineering, а до погодження змін залучаються Back Office, Customer Support, Legal & Compliance, Marketing і Business Development . Тому оптимізація має бути спрямована не на зміну організаційної структури, а на підвищення керованості чинного проєктного циклу.

1. Впровадження системного контролю потоку завдань.

У JustMarkets Tech проєктна задача проходить кілька етапів: формування вимоги, технічне оцінювання, розробку, тестування, погодження, реліз і аналіз результату. За участі кількох напрямів виникає ризик затримок на етапах погодження, уточнення вимог, тестування або перевірки відповідності регуляторним вимогам. Для усунення цієї проблеми доцільно запровадити єдину модель руху задачі від постановки до релізу.

Єдина модель має включати статуси: backlog, аналіз вимоги, технічне оцінювання, розробка, тестування, compliance-погодження, готовність до релізу,

реліз, післярелізний аналіз. Для кожного статусу потрібно закріпити відповідального учасника, критерій переходу на наступний етап і допустимий строк перебування задачі у відповідному статусі. Kanban Guide визначає обмеження незавершеної роботи, управління потоком і вимірювання часу проходження завдань як базові елементи Kanban-підходу. Для JustMarkets Tech це означає потребу в контролі трьох показників: часу проходження задачі, кількості задач у роботі та кількості завершених задач за період.

Для практичного впровадження потрібно:

- запровадити єдині статуси задач для Product, Engineering, Back Office, Legal & Compliance і Customer Support;
- встановити ліміти незавершеної роботи для стадій аналізу, розробки, тестування та погодження;
- фіксувати причини блокування задач, зокрема очікування рішення Product, технічну залежність, compliance-перевірку, уточнення операційних вимог або дефект після тестування;
- щотижня аналізувати задачі, що перебувають у роботі довше за допустимий строк;
- окремо позначати задачі, які повернулися на доопрацювання після тестування, релізу або погодження.

Результатом має стати скорочення затримок, кращий контроль навантаження на Engineering і зменшення кількості задач, які довго залишаються між стадіями. Для брокерської платформи це важливо через залежність торгових функцій, мобільних оновлень, верифікації та регуляторних змін від швидкості проходження задач через проєктний цикл.

2. Запровадження системи релізних метрик.

Діагностика показала, що релізний процес у компанії функціонує, але потребує чіткішого вимірювання. Для JustMarkets Tech важливо контролювати не лише факт запуску зміни, а й регулярність релізів, частку проблемних запусків,

час проходження зміни до продуктивного середовища та швидкість відновлення після інциденту.

Для цього доцільно використати DORA-метрики. Вони охоплюють частоту розгортань, час проходження зміни до продуктивного середовища, частку невдалих змін і час відновлення після збою. Ці показники характеризують швидкість і стабільність постачання програмних змін.

Таблиця 3.1.1

Рекомендовані релізні метрики для JustMarkets Tech*

Метрика	Зміст показника	Управлінське призначення
Частота релізів	кількість запусків змін за тиждень або місяць	оцінювання регулярності постачання змін
Час проходження зміни	період від готовності задачі до запуску в продукт	виявлення затримок між розробкою, тестуванням і релізом
Частка невдалих змін	частка релізів, після яких виникли інциденти, дефекти або відкат	оцінювання якості підготовки релізів
Час відновлення після інциденту	тривалість усунення збою після запуску	оцінювання здатності команди швидко стабілізувати сервіс
Кількість післярелізних дефектів	помилки, виявлені після запуску зміни	оцінювання якості тестування та релізного контролю

**Джерело: складено автором*

Релізні метрики потрібно закріпити за типами проєктів. Для торгової платформи основними показниками мають бути доступність сервісу, швидкість виконання торгових операцій, кількість інцидентів і час відновлення. Для мобільного продукту, швидкість застосунку, кількість помилок після оновлення, утримання користувачів, конверсія та звернення до підтримки. Для Back Office і compliance-змін, помилки у верифікації, час обробки операції, кількість повторних звернень і кількість повернень задач на доопрацювання.

3. Формалізація післярелізного аналізу.

У системі гнучкого управління запуск функції не завершує управлінський цикл. Після релізу необхідно оцінити, чи дала зміна очікуваний результат, які

проблеми виникли, які звернення отримала підтримка, які технічні дефекти з'явилися і які задачі мають повернутися до backlog.

Для JustMarkets Tech доцільно запровадити обов'язковий післярелізний аналіз для змін, що впливають на торгову платформу, мобільний застосунок, платежі, верифікацію, партнерські сервіси або регуляторні процеси. Scrum Guide визначає Sprint Review як перевірку результату і подальших дій, а Sprint Retrospective як інструмент підвищення якості та ефективності роботи команди. Для JustMarkets Tech ці практики варто адаптувати у формат післярелізного огляду.

Післярелізний аналіз має охоплювати такі питання:

- чи досягнуто очікуваного продуктового результату;
- які технічні показники змінилися після релізу;
- чи збільшилася кількість звернень до Customer Support;
- які дефекти виявлено після запуску;
- на якому етапі виникла затримка;
- які задачі потрібно повернути до backlog;
- які процесні проблеми повторюються між релізами.

Для змін, що спричинили інциденти, потрібен окремий розбір причин без персонального звинувачення. Blameless postmortem, тобто розбір інциденту без пошуку винного, орієнтований на з'ясування системних причин, навчання команди та запобігання повторенню проблем. Такий формат сприяє відкритому аналізу помилок і покращенню процесів реагування.

4. Окреме управління технічним боргом.

У діагностиці слабким підкритерієм визначено баланс між новими функціями і технічним боргом. Для JustMarkets Tech це має практичне значення через одночасний розвиток торгової платформи, мобільного продукту, інфраструктури, клієнтських сервісів і регуляторних процесів. Новий функціонал має вищу видимість для бізнесу, однак накопичення технічного боргу знижує

швидкість розробки, ускладнює тестування, підвищує ризик інцидентів і збільшує витрати на підтримку.

Технічний борг потрібно виділити в окрему категорію backlog. Кожна така задача має оцінюватися за чотирма параметрами: вплив на стабільність, вплив на безпеку, вплив на швидкість подальшої розробки, ризик інцидентів. У кожному циклі планування доцільно резервувати частину ресурсу Engineering на технічні задачі. Для JustMarkets Tech практичним орієнтиром може бути 15-20 % місткості інженерної команди на роботи, пов'язані з технічним боргом, стабільністю, автоматизацією тестування та покращенням інфраструктури.

Таблиця 3.1.2

Механізм управління технічним боргом у JustMarkets Tech

Елемент механізму	Зміст
Окрема категорія backlog	виділення задач технічного боргу поряд із продуктовими задачами
Оцінювання впливу	визначення впливу технічної задачі на стабільність, безпеку, швидкість розробки та ризик інцидентів
Резерв ресурсу Engineering	закріплення частки робочого часу на технічні покращення
Регулярний перегляд	розгляд технічного боргу під час пріоритезації разом із Product
Зв'язок із релізними метриками	аналіз впливу технічного боргу на дефекти, час відновлення та стабільність релізів

**Джерело: складено автором*

Реалізація цього напрямку зменшить ризик накопичення прихованих технічних проблем. Для JustMarkets Tech це особливо важливо через високі вимоги до доступності торгової платформи, швидкості операцій і безпеки фінансових процесів.

5. Систематизація клієнтського зворотного зв'язку.

Customer Support у JustMarkets Tech передає інформацію про звернення, проблеми, очікування та відгуки користувачів, що вже зафіксовано в організації проєктної взаємодії. Разом із тим повторювані звернення мають переходити у

продуктові задачі більш системно. Для цього потрібен реєстр повторюваних клієнтських проблем.

У реєстрі звернення доцільно групувати за продуктом, функцією, типом проблеми, частотою, впливом на користувача та відповідальним напрямом. Проблеми мобільного застосунку мають спрямовуватися до Product Mobile і Engineering, проблеми верифікації, до Back Office і Legal & Compliance, проблеми торгових операцій, до Trading і Engineering.

До backlog мають потрапляти не всі окремі звернення, а повторювані або критичні проблеми, що впливають на торговий досвід, операції або утримання клієнтів. Для пріоритезації клієнтських проблем варто використовувати такі параметри:

- частота повторення звернення;
- вплив на торгову операцію;
- вплив на мобільний шлях користувача;
- ризик втрати клієнта;
- зв'язок із регуляторними вимогами;
- складність технічного виправлення;
- очікуваний вплив на задоволеність користувачів.

Результатом стане точніший зв'язок між клієнтським досвідом і backlog. Product отримає не розрізнені сигнали від підтримки, а згруповані проблеми з оцінкою частоти та впливу.

6. Впровадження матриці погодження змін.

Проекти JustMarkets Tech мають міжфункціональний характер, тому значущі зміни потребують участі кількох напрямів. Надмірна кількість погоджень уповільнює потік, а недостатня кількість погоджень спричиняє повернення задач через пропущені регуляторні, операційні або торгові обмеження. Матриця погодження має визначати обов'язкових учасників залежно від типу зміни.

Матриця погодження змін у проєктах JustMarkets Tech*

Тип зміни	Обов'язкові учасники погодження	Результат погодження
Торгова функція	Product, Engineering, Trading, Legal & Compliance	підтверджені вимоги до функції, ризиків і регулювання
Мобільний продукт	Product, Engineering, Customer Support, Marketing, Legal & Compliance	узгоджені вимоги до користувацького досвіду, ринку й безпеки
Верифікація або клієнтські дані	Product, Back Office, Legal & Compliance, Engineering	погоджений процес обробки даних і вимоги до системи
Платежі або фінансові операції	Product, Back Office, Finance & Accounting, Legal & Compliance, Engineering	узгоджені операційні, фінансові й технічні вимоги
Інфраструктурна зміна	Engineering, Infrastructure, Product	визначений вплив на стабільність, продуктивність і релізи
Регіональна адаптація	Product, Marketing, Business Development, Legal & Compliance	підтверджені ринкові, мовні, регіональні та регуляторні вимоги

*Джерело: складено автором

Матриця зменшить кількість повторних погоджень і повернень задач на доопрацювання. Також вона чітко розмежує, які зміни потребують участі Legal & Compliance, Back Office або Trading ще до технічного оцінювання.

7. Впровадження критеріїв готовності задачі до розробки.

Частина затримок і повернень виникає тоді, коли задача надходить до Engineering із неповним описом, нечіткими критеріями приймання або пропущеними операційними й compliance-обмеженнями. Для JustMarkets Tech це особливо важливо у змінах, пов'язаних із торговими процесами, платежами, верифікацією, мобільним продуктом і регіональними вимогами.

У Scrum підготовка і уточнення Product Backlog підтримує готовність команди до планування та виконання роботи. Для JustMarkets Tech цей принцип потрібно перевести у внутрішній чек-лист готовності задачі до Engineering.

Задача має переходити в розробку за наявності таких елементів:

- опис бізнес-потреби;
- очікуваний результат для користувача або внутрішнього процесу;

- критерії приймання;
- пріоритет;
- пов'язані продуктові або технічні метрики;
- вплив на торгову платформу, мобільний продукт, верифікацію, платежі або операційний процес;
- потреба в участі Legal & Compliance;
- потреба в участі Back Office або Customer Support;
- технічні залежності;
- ризики для стабільності та безпеки.

Впровадження чек-листа зменшить кількість повторних уточнень, повернень задач і ситуацій, коли технічна команда починає роботу без повного розуміння очікуваного результату.

8. Впровадження карти післярелізних показників за типами проєктів.

У JustMarkets Tech різні проєкти мають різні критерії успіху. Для торгової платформи головними є доступність сервісу й швидкість операцій, для мобільного продукту, швидкість застосунку, утримання і конверсія, для верифікації, час проходження перевірки, для платежів, успішність і швидкість операцій. Єдина узагальнена оцінка релізу не відображає реального впливу зміни.

Таблиця 3.1.4

Карта післярелізних показників для JustMarkets Tech

Тип проєкту	Показники після релізу
Торгова платформа	доступність сервісу; швидкість виконання операцій; кількість інцидентів; звернення щодо торгових помилок
Мобільний продукт	швидкість застосунку; залученість; утримання; конверсія; оцінки користувачів; звернення до підтримки
Верифікація	час проходження верифікації; кількість відмов; кількість повторних звернень; помилки в даних
Платежі	успішність операцій; час обробки; кількість помилок; повторні звернення
Партнерські продукти	активність партнерів; завершення onboarding; помилки у виплатах або винагородах
Інфраструктура	кількість інцидентів; час відновлення; навантаження на системи; стабільність після релізу

**Джерело: складено автором*

Післярелізна карта має завершуватися одним із трьох рішень: залишити зміну без доопрацювання, повернути її до backlog або створити задачу на технічне чи продуктове покращення. Це посилить зв'язок між релізом і фактичним результатом для користувача, платформи та бізнесу.

Таблиця 3.1.5

Рекомендації щодо оптимізації процесів гнучкого управління проєктами у
JustMarkets Tech

Напрямок оптимізації	Проблема	Рекомендовані дії	Очікуваний результат
Контроль потоку завдань	слабка прозорість часу проходження задач, блокувань і повернень	єдині статуси задач; WIP-ліміти; фіксація блокувань; аналіз задач із перевищенням строків	коротший цикл проходження задач і менше затримок
Релізні метрики	недостатнє вимірювання якості релізів	DORA-метрики; облік частоти релізів; контроль невдалих змін; вимірювання часу відновлення	керованість релізного процесу
Післярелізний аналіз	слабка регулярність аналізу після запуску	післярелізні огляди; розбір інцидентів; оновлення backlog; фіксація причин проблем	перехід від запуску функцій до системного вдосконалення
Управління технічним боргом	витіснення технічних задач новим функціоналом	окрема категорія backlog; оцінка впливу боргу; резерв часу Engineering	нижчий ризик інцидентів і стабільніша платформа
Клієнтський зворотний зв'язок	недостатня структурованість повторюваних звернень	реєстр проблем; групування звернень; передання критичних тем у backlog	точніше врахування клієнтських потреб
Матриця погодження змін	затримки або повернення задач через пропущені вимоги	визначення обов'язкових учасників погодження за типом зміни	менше повторних погоджень і доопрацювань
Готовність задачі до розробки	неповний опис вимог перед Engineering	чек-лист готовності; критерії приймання; метрики результату; перевірка compliance-впливу	вища якість постановки задач
Карта післярелізних показників	слабкий зв'язок між релізом і фактичним результатом	показники за типом проєкту; рішення після релізу: залишити, доопрацювати, повернути в backlog	оцінювання ефекту кожної значущої зміни

*Джерело: складено автором

Запропоновані напрями оптимізації спрямовані на усунення проблем, виявлених у діагностиці. Для JustMarkets Tech першочергове значення мають контроль потоку завдань, релізні метрики, післярелізний аналіз і управління технічним боргом. Вони безпосередньо впливають на стабільність торгової платформи, швидкість запуску змін, якість мобільного продукту й ефективність роботи міжфункціональних команд. Систематизація клієнтського зворотного зв'язку, матриця погодження змін і критерії готовності задачі доповнюють ці напрями, оскільки зменшують кількість нечітких вимог, повторних погоджень і повернень на доопрацювання.

Отже, удосконалення гнучкого управління проектами у JustMarkets Tech має ґрунтуватися на посиленні вимірюваності та керованості чинного продуктово-інженерного циклу. Product і Engineering залишаються ядром управління змінами, але процес потребує чіткіших правил проходження задач, релізного контролю, післярелізного аналізу, управління технічним боргом і перетворення клієнтських сигналів на пріоритетні задачі. Реалізація запропонованих рекомендацій підвищить передбачуваність роботи команд, зменшить кількість затримок і післярелізних проблем, а також посилить зв'язок між гнучким управлінням, стабільністю платформи та користувацьким результатом.

3.2. Обґрунтування впровадження інтегрованої моделі гнучкого управління ІТ-проектами у JustMarkets Tech

Результати діагностики показали, що система гнучкого управління ІТ-проектами у JustMarkets Tech має сформовану продуктово-інженерну основу, однак потребує посилення в частині контролю потоку завдань, релізних показників, післярелізного аналізу та управління технічним боргом. У роботі визначено, що управління ІТ-проектами компанії побудоване навколо взаємодії

Product та Engineering, а до погодження змін залучаються Back Office, Customer Support, Legal & Compliance, Marketing і Business Development . Більшість працівників компанії працює віддалено, тому інтегрована модель має враховувати не лише технічний цикл розробки, а й цифрову координацію, асинхронну комунікацію, фіксацію рішень і прозорість статусів задач.\

Інтегрована модель має поєднувати чотири управлінські підходи: Scrum, Kanban, DevOps/DORA та післярелізний аналіз. Scrum-елемент потрібен для впорядкування продуктової роботи, пріоритезації, перегляду результатів і регулярного вдосконалення. Scrum Guide визначає Sprint Review як робочу сесію для аналізу виконаної роботи та подальших дій, а Product Backlog може коригуватися відповідно до нових можливостей і отриманих результатів . Kanban-елемент потрібен для прозорого руху задач через процес, контролю незавершеної роботи й виявлення затримок.

Kanban-підхід спрямований на оптимізацію потоку цінності через візуальну pull-систему, а WIP-ліміти створюють умови для більшої передбачуваності та зменшення перевантаження.

DevOps/DORA-елемент потрібен для вимірювання швидкості й стабільності технічного постачання змін. DORA-метрики охоплюють час проходження зміни, частоту розгортань і час відновлення після невдалого розгортання. Післярелізний аналіз потрібен для фіксації причин інцидентів, повторюваних проблем і рішень для наступних циклів.

Практика blameless postmortem передбачає аналіз проблем без персонального звинувачення та спрямована на навчання команди й покращення процесів .

Інтегровану модель доцільно побудувати як послідовність із шести взаємопов'язаних блоків. Її основою має бути не окремий інструмент, а повний управлінський цикл: від появи потреби до оцінювання результату після запуску.

Інтегрована модель гнучкого управління IT-проектами у JustMarkets Tech

Блок моделі	Зміст блоку	Основні учасники	Інструменти
Формування потреби	Визначення джерела зміни: користувач, ринок, торговий процес, технічна проблема, регуляторна вимога	Product, Customer Support, Trading, Marketing, Legal & Compliance	реєстр клієнтських проблем, продуктова аналітика, запити бізнес-напрямів
Пріоритезація	Відбір задач за впливом на продукт, користувача, стабільність, ризики й бізнес-цілі	Product, Engineering, Trading, Legal & Compliance	roadmap, backlog, матриця пріоритетів, оцінка технічного боргу
Погодження зміни	Узгодження вимог із технічними, операційними, торговими та регуляторними обмеженнями	Product, Engineering, Back Office, Legal & Compliance, Customer Support	матриця погодження змін, критерії готовності задачі
Реалізація	Розробка, тестування, перегляд коду, підготовка релізу	Engineering	CI/CD, code review, автоматизоване тестування, управління релізами
Контроль потоку і релізу	Вимірювання часу проходження задач, блокувань, частоти релізів, невдалих змін і відновлення після інцидентів	Engineering, Product, Infrastructure	Kanban-метрики, WIP-ліміти, DORA-метрики
Післярелізне вдосконалення	Аналіз результату, звернень, дефектів, інцидентів і впливу на продукт	Product, Engineering, Customer Support, Trading	післярелізний огляд, blameless postmortem, оновлення backlog

**Джерело: складено автором*

Запропонована модель має усунути розрив між продуктовим плануванням і технічним результатом. У чинній системі Product та Engineering формують ядро управління змінами, але слабшими залишаються кількісне вимірювання потоку, контроль післярелізних наслідків і повернення результатів аналізу в наступний цикл планування. Інтегрована модель закріплює цю послідовність: кожна зміна має пройти етап формування потреби, пріоритезації, погодження, технічної реалізації, релізного контролю та післярелізного аналізу.

Для практичного впровадження запропоновано проєкт «Впровадження інтегрованої моделі гнучкого управління IT-проєктами у JustMarkets Tech». Його зміст полягає у поєднанні наявної взаємодії Product та Engineering із системним контролем задач, релізними метриками, післярелізним аналізом, матрицею погодження змін, окремим управлінням технічним боргом і єдиним цифровим простором для віддалених команд.

Таблиця 3.2.2

Паспорт проєкту впровадження інтегрованої моделі гнучкого управління IT-проєктами у JustMarkets Tech

Елемент паспорта	Зміст
Назва проєкту	Впровадження інтегрованої моделі гнучкого управління IT-проєктами у JustMarkets Tech
Мета проєкту	Підвищення керованості IT-проєктів через прозорий потік задач, релізні метрики, післярелізний аналіз, контроль технічного боргу та цифрову координацію віддалених команд
Об'єкт впровадження	Проєктний цикл Product та Engineering із залученням Trading, Customer Support, Back Office, Legal & Compliance, Marketing і Business Development
Основна проблема	Недостатня прозорість часу проходження задач, заблокованих робіт, повернень на доопрацювання, частоти релізів, післярелізних помилок і рішень, що губляться у віддаленій комунікації
Формат реалізації	Пілотне впровадження на 2-3 продуктових напрямках: торгова платформа, мобільний продукт, операційно-регуляторні зміни
Тривалість проєкту	34 тижні, приблизно 8 місяців
Ключові учасники	Product Lead, Engineering Manager, Technical Project Manager, DevOps / Infrastructure, QA, Customer Support Lead, Legal & Compliance, Back Office
Особливість впровадження	Віддалена робота більшості працівників, потреба в асинхронній документації, фіксації рішень, єдиному просторі задач і прозорих післярелізних висновках
Основні інструменти	Kanban-дошка, WIP-ліміти, DORA-метрики, матриця погодження змін, чек-лист готовності задачі, реєстр клієнтських проблем, база рішень і післярелізних висновків
Очікуваний результат	Скорочення затримок, зменшення повторних доопрацювань, посилення релізного контролю, менше дублювання комунікацій, регулярне використання даних після запуску змін
Орієнтовний бюджет	4 795 001 грн

**Джерело: складено автором*

Методичною основою проєкту є Kanban-підхід до управління потоком завдань, DORA-метрики для оцінювання релізного процесу та Scrum-практики перегляду результатів і вдосконалення роботи команди. DORA визначає чотири ключові метрики програмного постачання: частоту розгортань, час виконання змін, частку невдалих змін і час відновлення після збою.

Проєкт має пілотний характер, оскільки одночасне впровадження на всі напрями створило б надмірне навантаження на команди.

Таблиця 3.2.3

Етапи реалізації проєкту впровадження інтегрованої моделі

Етап	Тривалість	Зміст робіт	Результат
1. Діагностика поточного процесу	3 тижні	аналіз руху задач, затримок, блокувань, повернень, проблем погодження, особливостей віддаленої взаємодії	карта процесу та перелік проблемних зон
2. Проктування інтегрованої моделі	4 тижні	визначення статусів задач, правил переходу між етапами, ролей, відповідальних осіб, матриці погодження змін	опис цільової моделі управління задачами
3. Налаштування цифрового середовища	5 тижнів	налаштування Kanban-дошки, DORA-дашбордів, простору для рішень, чек-листів, релізних даних і післярелізних висновків	єдиний цифровий простір для віддалених команд
4. Навчання пілотної команди	2 тижні	навчання учасників правилам роботи з моделлю, Kanban-потоком, релізними метриками, післярелізним аналізом і асинхронною взаємодією	підготовлена пілотна команда
5. Пілотне впровадження на обраних напрямках	8 тижнів	застосування моделі для торгової платформи, мобільного продукту та операційно-регуляторних змін	перевірка моделі в реальному проєктному циклі
6. Аналіз результатів пілоту	3 тижні	оцінювання часу проходження задач, блокувань, релізних показників, післярелізних дефектів, повторних уточнень і комунікаційних втрат	перелік коригувань до моделі
7. Коригування моделі та регламентів	3 тижні	уточнення статусів, WIP-лімітів, матриці погодження, чек-листів, правил післярелізного аналізу	оновлена модель управління
8. Масштабування на інші команди	6 тижнів	поступове поширення моделі на інші продуктові та технічні команди, контроль адаптації й підтримка учасників	інтегрована модель у ширшому проєктному контурі

**Джерело: складено автором*

Доцільно почати з процесів, де вплив на клієнтський досвід і стабільність сервісу є найбільшим: торгової платформи, мобільного продукту та операційно-регуляторних змін. Такий формат знижує ризик організаційного перевантаження та дає підстави протестувати модель до масштабування.

Загальна тривалість проєкту становить 34 тижні, або приблизно 8 місяців. Найбільше часу потребує пілотне впровадження, оскільки модель має бути перевірена в реальному циклі роботи над задачами, релізами та післярелізним аналізом. Значний період відводиться на налаштування цифрового середовища, що пов'язано з віддаленим форматом роботи більшості працівників і потребою в єдиному просторі для задач, рішень, метрик і документації. Масштабування винесене в окремий етап, оскільки різні команди можуть мати різну готовність до нових правил роботи.

На етапі діагностики поточного процесу потрібно зафіксувати фактичний шлях задачі від появи продуктової потреби до релізу. Основна увага приділяється точкам затримки: очікуванню рішення Product, уточненню вимог, технічним залежностям, тестуванню, погодженню з Legal & Compliance і поверненню задач на доопрацювання. Для віддалених команд додатково оцінюються дублювання повідомлень, незафіксовані рішення, очікування відповідей і розбіжності між статусом задачі в системі та фактичним станом роботи.

На етапі проєктування інтегрованої моделі потрібно визначити цільову схему руху задач, матрицю погодження змін, правила залучення суміжних команд, критерії готовності задачі до розробки та правила післярелізного аналізу. Саме на цьому етапі встановлюється, які зміни потребують обов'язкового погодження з Legal & Compliance, Back Office, Customer Support або Trading.

На етапі налаштування цифрового середовища потрібно закріпити однакові статуси для всіх пілотних напрямів: backlog, аналіз вимоги, технічне оцінювання, розробка, тестування, compliance-погодження, готовність до релізу, реліз, післярелізний аналіз. Для кожного статусу встановлюється відповідальний

учасник, критерій переходу на наступний етап і допустимий строк перебування задачі. Це зменшить кількість задач, які формально перебувають у роботі, але фактично очікують рішення іншої команди.

Цифрова координація має стати окремим блоком, оскільки віддалений формат роботи збільшує залежність від якості асинхронної комунікації. Усі рішення щодо зміни, погодження, ризиків, критеріїв приймання, релізу й післярелізних висновків мають фіксуватися в єдиному просторі. Це зменшить потребу в повторних уточненнях і втрати інформації між Product, Engineering, Customer Support, Back Office, Legal & Compliance та іншими напрямками.

Навчання пілотної команди має бути коротким, але прикладним. Основний зміст навчання охоплює правила роботи зі статусами задач, WIP-ліміти, релізні метрики, after-release review, правила фіксації рішень, матрицю погодження та критерії готовності задачі до розробки. Навчання має завершитися практичною симуляцією проходження задачі від формування вимоги до післярелізного аналізу.

Пілотне впровадження має охоплювати три типи змін: торгову функцію, мобільне оновлення та операційно-регуляторну зміну. Це дасть змогу перевірити модель у різних ситуаціях: із високою технічною складністю, із сильним впливом на користувацький досвід і з потребою в compliance-погодженні.

Після пілотного впровадження проводиться аналіз результатів. Оцінюються час проходження задачі, кількість заблокованих робіт, кількість повернень на доопрацювання, частота релізів, післярелізні дефекти, звернення до підтримки та втрати часу через повторну комунікацію. За підсумками цього аналізу коригуються статуси, WIP-ліміти, матриця погодження, чек-лист готовності задачі та правила післярелізного аналізу.

Після коригування модель масштабується на інші команди. Масштабування має бути поступовим, оскільки різні напрями можуть мати різну кількість задач, різні регуляторні обмеження та різну інтенсивність комунікації.

На цьому етапі важливо не лише поширити правила, а й забезпечити підтримку команд під час переходу до нової моделі.

Таблиця 3.2.4

Розподіл відповідальності учасників проєкту

Учасник	Зона відповідальності
Product Lead	вимоги, пріоритети, дорожня карта, зв'язок із продуктивними показниками
Engineering Manager	технічне оцінювання, якість реалізації, контроль технічного боргу
Technical Project Manager	координація впровадження, контроль строків, статусів і ризиків
DevOps / Infrastructure	релізні метрики, CI/CD, стабільність, час відновлення після інцидентів
QA	тестування, контроль дефектів, післярелізні помилки
Customer Support Lead	реєстр повторюваних звернень і передання проблем у Product
Legal & Compliance	погодження регуляторно чутливих змін
Back Office	операційні вимоги до верифікації, платежів і клієнтських процедур

**Джерело: складено автором*

Для економічного обґрунтування проєкту використано розрахунок витрат на пілотну команду, навчання, налаштування аналітики, інтеграцію даних, цифрову координацію та розроблення внутрішніх регламентів. Оскільки внутрішня фінансова інформація JustMarkets Tech не є відкритою, розрахунок побудовано як прогнозу модель для пілотного впровадження. Для визначення вартості робочого часу використано ринкові зарплатні орієнтири українського IT-ринку. За даними DOU, у грудні 2025 року медіанна зарплата Engineering Manager становила 7000 дол. США, Head of Product, 4900 дол. США, Delivery Manager, 4700 дол. США.

Для перерахунку зарплатних орієнтирів у гривню використано офіційний курс НБУ станом на 13.05.2026: 1 дол. США = 43,9709 грн. У розрахунках курс округлено до 43,97 грн за 1 дол. США.

Розрахунок здійснено як місячна ставка в гривнях × частка залучення × 7,8 місяця, де 7,8 місяця відповідає 34 тижням проєкту. Вартість участі пілотної команди становить 3 241 092 грн.

Таблиця 3.2.5

Склад пілотної команди проєкту впровадження інтегрованої моделі
гнучкого управління ІТ-проєктами у JustMarkets Tech

Роль	Кількість осіб	Місячна ставка, дол. США	Місячна ставка, грн	Залучення до проєкту	Вартість участі за 34 тижні, грн
Engineering Manager	1	7000	307796	20 %	480162
Product Lead / Product Manager	2	4900	215457	20 %	672226
Technical Project Manager / Delivery Manager	1	4700	206663	50 %	805986
Software Engineer	5	2400	105530	15 %	617350
QA Engineer	2	2200	96736	15 %	226362
DevOps / Infrastructure Engineer	1	3000	131913	20 %	205784
Customer Support Lead	1	1800	79148	15 %	92603
Legal & Compliance Specialist	1	2500	109927	10 %	85743
Back Office Specialist	1	1600	70353	10 %	54875
Разом	15				3241092

**Джерело: складено автором*

Найбільшу частку витрат формують Technical Project Manager, Product Lead / Product Manager та Software Engineer, оскільки саме ці ролі найбільше залучені до зміни правил роботи, налаштування потоку задач, цифрової координації та впровадження релізних метрик.

Загальний бюджет пілотного проєкту становить 4 795 001 грн. Основна частина витрат припадає на участь пілотної команди, оскільки впровадження інтегрованої моделі потребує не лише технічного налаштування інструментів, а й зміни правил постановки задач, погодження вимог, цифрової фіксації рішень, контролю релізів і післярелізного аналізу.

Таблиця 3.2.6

Бюджет проєкту впровадження інтегрованої моделі гнучкого управління
ІТ-проєктами у JustMarkets Tech

Стаття витрат	Розрахунок	Сума, грн
Участь пілотної команди у впровадженні	за табл. 3.2.5	3241092
Навчання команди Kanban, DORA, післярелізного аналізу та асинхронній взаємодії	15 осіб × 9000 грн	135000
Зовнішній Agile / DevOps-консультант	100 год × 2500 грн	250000
Налаштування Kanban-потoku, DORA-дашбордів і звітності	140 год × 1800 грн	252000
Інтеграція даних із Jira / GitLab / CI/CD / Customer Support	120 год × 1800 грн	216000
Налаштування простору цифрової координації та бази рішень	100 год × 1600 грн	160000
Розроблення матриці погодження, чек-листа готовності задач і регламентів	80 год × 1500 грн	120000
Резерв на непередбачені витрати	10 % від прямих витрат	435909
Разом		4795001

**Джерело: складено автором*

Економічний ефект розраховано через скорочення витрат робочого часу. Основними джерелами економії є зменшення повторних доопрацювань, скорочення часу блокування задач, зниження витрат часу на післярелізні дефекти, зменшення повторних уточнень між Product і Engineering, скорочення ручної звітності та зниження витрат, пов'язаних із віддаленою комунікацією.

Середній місячний фонд оплати праці пілотної команди становить:

$307\,796 + 430\,914 + 206\,663 + 527\,650 + 193\,472 + 131\,913 + 79\,148 + 109\,927 + 70\,353 = 2\,057\,836$ грн.

Середня кількість робочих годин на місяць:

$15 \text{ осіб} \times 160 \text{ год} = 2\,400$ год.

Середня погодинна ставка:

$2\,057\,836 / 2\,400 = 857,4$ грн/год.

Для розрахунку прийнято 857 грн/год.

Таблиця 3.2.7

Розрахунок очікуваного річного економічного ефекту від впровадження
проєкту

Джерело економічного ефекту	Поточні втрати часу	Очікуване скорочення	Економія часу на місяць	Розрахунок	Річний ефект, грн
Повторні доопрацювання задач	260 год/міс	30 %	78 год/міс	$78 \times 857 \times 12$	802152
Блокування задач і очікування погоджень	220 год/міс	25 %	55 год/міс	$55 \times 857 \times 12$	565620
Післярелізні дефекти та інциденти	180 год/міс	25 %	45 год/міс	$45 \times 857 \times 12$	462780
Повторні уточнення вимог між Product та Engineering	140 год/міс	30 %	42 год/міс	$42 \times 857 \times 12$	431928
Ручне формування звітності щодо релізів і задач	60 год/міс	50 %	30 год/міс	$30 \times 857 \times 12$	308520
Втрати часу через незафіксовані рішення у віддаленій роботі	120 год/міс	35 %	42 год/міс	$42 \times 857 \times 12$	431928
Дублювання комунікацій між командами	100 год/міс	30 %	30 год/міс	$30 \times 857 \times 12$	308520
Разом	1080 год/міс		322 год/міс		3311448

**Джерело: складено автором*

Очікуваний річний економічний ефект становить 3 311 448 грн. Він формується через економію 322 годин на місяць, або 3 864 годин на рік. Найбільший ефект очікується від скорочення повторних доопрацювань, блокувань, незафіксованих рішень і дублювання комунікацій, оскільки саме ці втрати посилюються у віддаленому форматі роботи.

Після впровадження моделі потрібні поточні витрати на підтримку аналітичних панелей, оновлення регламентів, навчання нових учасників, аудит процесу та підтримку бази рішень. Для пілотного масштабу ці витрати оцінено на рівні 480 000 грн на рік, або 40 000 грн на місяць.

Чистий річний ефект: $3\,311\,448 - 480\,000 = 2\,831\,448$ грн.

Показники економічної ефективності проекту з урахуванням віддаленого
формату роботи

Показник	Розрахунок	Значення
Початкові витрати на впровадження	за табл. 3.2.6	4795001 грн
Річний економічний ефект	за табл. 3.2.7	3311448 грн
Річні витрати на підтримку	40000×12	480000 грн
Чистий річний ефект	$3311448 - 480000$	2831448 грн
Строк окупності	$4795001 / 2831448$	1,69 року
Строк окупності в місяцях	$1,69 \times 12$	20,3 міс
Рентабельність за перший рік	$(2831448 - 4795001) / 4795001 \times 100 \%$	-40,9 %
Рентабельність за другий рік	$2831448 / 4795001 \times 100 \%$	59,0 %
Сукупний чистий ефект за 2 роки	$2831448 \times 2 - 4795001$	867895 грн

**Джерело: складено автором*

Економічний розрахунок показує, що проєкт потребує 4 795 001 грн початкових витрат. Очікуваний річний економічний ефект становить 3 311 448 грн, а чистий річний ефект після врахування витрат на підтримку, 2 831 448 грн. Строк окупності становить 20,3 місяця. Проєкт не дає повної окупності протягом першого року, але формує позитивний результат у дворічному періоді: сукупний чистий ефект за два роки становить 867 895 грн.

Окрім прямої економії робочого часу, проєкт має організаційний ефект. Скорочення повторних доопрацювань підвищить якість постановки задач. Зменшення блокувань і погоджень підвищить передбачуваність потоку. Контроль післярелізних дефектів зменшить навантаження на Engineering, QA, Customer Support і DevOps. Окреме управління технічним боргом посилить стабільність платформи та знизить ризик інцидентів у майбутніх релізах. Єдина база рішень і цифрова координація зменшать втрати інформації у віддаленій роботі та скоротять потребу в повторних уточненнях.

Отже, впровадження інтегрованої моделі гнучкого управління ІТ-проєктами у JustMarkets Tech є доцільним з управлінської та економічної позицій. Проєкт спрямований на слабкі зони, визначені під час діагностики: потік завдань,

релізні метрики, післярелізний аналіз, технічний борг і втрати комунікації у віддаленому форматі роботи. За прогнозним розрахунком впровадження окупається за 20,3 місяця, а основний ефект формується за рахунок скорочення втрат робочого часу, зменшення повторних доопрацювань, кращого контролю релізів, фіксації рішень і підвищення керованості продуктово-інженерного циклу.

ВИСНОВКИ

Гнучке управління проектами для IT-підприємства має найбільшу цінність тоді, коли воно забезпечує швидке перетворення змінних вимог на якісний цифровий продукт. Для IT-сфери ключовими умовами ефективності є короткий цикл створення результату, прозора пріоритезація, постійний зворотний зв'язок, технічна якість і здатність команди швидко змінювати порядок робіт зі збереженням керованості. У фінтех-секторі ці вимоги мають особливе значення, оскільки кожне оновлення продукту впливає на стабільність сервісу, швидкість операцій, клієнтський досвід і відповідність регуляторним вимогам.

Agile потребує поєднання гнучкості з чіткими правилами постановки задач, контролем технічної якості, вимірюванням релізів і регулярним аналізом результатів. Scrum, Kanban, XP, Lean Agile і DevOps виконують різні функції, тому для IT-підприємства найбільш обґрунтованим є їх комплексне використання. Scrum структурує роботу команди, Kanban підвищує прозорість потоку завдань, XP і DevOps посилюють технічну якість, а Lean Agile спрямовує процеси на зменшення втрат і створення цінності для користувача.

У JustMarkets Tech гнучке управління проектами має сформовану організаційну основу. Основний проектний цикл зосереджений навколо взаємодії Product та Engineering, а участь Trading, Customer Support, Back Office, Legal & Compliance, Marketing і Business Development забезпечує погодження продуктових, технічних, клієнтських, операційних і регуляторних вимог. Функціонально-проектна побудова управління відповідає специфіці брокерської платформи, оскільки розробка IT-рішень потребує одночасного врахування торгових процесів, стабільності сервісу, ринкових запитів і вимог до безпеки.

Оцінювання ефективності гнучкого управління показало нерівномірність розвитку окремих елементів системи. Найсильнішою стороною є технічне виконання, що свідчить про достатньо розвинений інженерний цикл, тестування,

CI/CD, контроль якості та підготовку релізів. Продуктове планування, міжфункціональна взаємодія і клієнтський результат також мають відносно сильні позиції, оскільки зміни пов'язуються з розвитком продукту, потребами користувачів і погодженням між основними командами.

Найбільш вразливими елементами залишаються потік завдань і результативність релізів. Саме вони визначають швидкість проходження вимоги від ідеї до запуску, кількість заблокованих задач, частоту повернень на доопрацювання, обсяг дефектів після релізу та швидкість реагування команди на інциденти. За умови прозорого вимірювання цих показників Agile-процес набуває більшої керованості й стає основою для обґрунтованих управлінських рішень.

Оптимізація гнучкого управління у JustMarkets Tech має спрямовуватися на посилення вимірюваності й керованості чинного продуктово-інженерного циклу. Найбільш доцільними є впровадження контролю потоку завдань, релізних метрик, післярелізного аналізу, управління технічним боргом, реєстру повторюваних клієнтських проблем, матриці погодження змін і критеріїв готовності задачі до розробки. Ці інструменти сприятимуть зменшенню втрат часу, повторних уточнень, затримок погодження, блокування задач і післярелізних дефектів.

Інтегрована модель гнучкого управління IT-проектами є найбільш придатним варіантом удосконалення для JustMarkets Tech, оскільки поєднує управлінські та технічні елементи в один цикл. Її цінність полягає у зв'язку між формуванням потреби, пріоритезацією, погодженням, технічною реалізацією, релізним контролем, післярелізним аналізом і оновленням backlog. За цієї моделі кожна зміна оцінюється за впливом на продукт, користувача, стабільність платформи та подальший розвиток системи.

Віддалений формат роботи більшості працівників підвищує значення цифрової координації. Для JustMarkets Tech важливо, щоб рішення, вимоги,

статуси задач, причини блокування, релізні показники й післярелізні висновки фіксувалися в єдиному інформаційному просторі. Це скорочує повторні повідомлення, підвищує прозорість взаємодії між командами та зменшує втрати інформації під час асинхронної роботи.

Економічне обґрунтування підтвердило доцільність запропонованого проєкту. Початкові витрати на впровадження інтегрованої моделі становлять 4 795 001 грн, очікуваний річний економічний ефект, 3 311 448 грн, чистий річний ефект після витрат на підтримку, 2 831 448 грн. Строк окупності становить 20,3 місяця, а сукупний чистий ефект за два роки, 867 895 грн. Основний результат формується за рахунок скорочення повторних доопрацювань, блокувань, післярелізних дефектів, ручної звітності, дублювання комунікацій і втрат часу через незафіксовані рішення.

Отже, подальший розвиток гнучкого управління проєктами у JustMarkets Tech має бути пов'язаний із переходом до вимірюваної інтегрованої системи. Найбільший потенціал підвищення ефективності зосереджений у прозорості потоку задач, контролі релізів, післярелізного аналізу, управлінні технічним боргом і цифровій координації віддалених команд. Реалізація цих змін здатна підвищити передбачуваність проєктної роботи, скоротити втрати часу, посилити стабільність брокерської платформи та покращити якість користувацького досвіду.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ільницький В. С. Гнучке управління проектами виробництва високотехнологічних продуктів в умовах цифровізації : дис. ... д-ра філософії : 073 Менеджмент. Львів : Національний університет «Львівська політехніка», 2024.
2. Приймак В. Управління проектами : навч. посіб. Київ : Київський національний університет імені Тараса Шевченка, 2017. 464 с.
3. Данилюк Н., Шулик Ю., Качан О. Сучасні підходи до управління проектною діяльністю ІТ-компаній. Наукові записки Національного університету «Острозька академія». Серія «Економіка». 2021. Вип. 22(50). С. 88–94.
4. Задоя В. О., Чаркіна Т. Ю., Корольов Д. С. Гнучкі методи та управління людськими ресурсами в системі проектного менеджменту. Вісник економіки транспорту і промисловості. 2025. № 89. С. 174–184. DOI: <https://doi.org/10.18664/btie.89.330676>.
5. Попко О., Тивончук П. Стратегічний маркетинговий аналіз ринку ІТ-послуг в Україні. Економіка та суспільство. 2025. Вип. 71. DOI: <https://doi.org/10.32782/2524-0072/2025-71-58>.
6. Приймак В., Корж Б. Гнучкі моделі управління командною роботою інжинірингових проектів. Вісник Київського національного університету імені Тараса Шевченка. 2020. № 7(208). С. 21–27. DOI: <https://doi.org/10.17721/1728-2667.2019/207-6/3>.
7. Приймак В., Янчук В. Ефективне управління проектами. Сучасні проблеми управління підприємствами: теорія та практика : матеріали міжнар. наук.-практ. конф., м. Харків, 18–19 березня 2019 р. Харків : ФОП Панов А. М., 2019. С. 86–88.
8. Сметанюк О. А., Бондарчук А. В. Особливості системи управління проектами в ІТ-компаніях. Агросвіт. 2020. № 10. С. 105–111. DOI: <https://doi.org/10.32702/2306-6792.2020.10.105>.

9. Храпкін О., Кіндрат О., Чопей Р. Управління проєктами в ІТ-галузі: методики, інструменти та керування ризиками. Економіка та суспільство. 2023. № 55. DOI: <https://doi.org/10.32782/2524-0072/2023-55-110>.
10. Akhtar A., Bakhtawar B., Akhtar S. Extreme programming vs Scrum: a comparison of Agile models. International Journal of Technology, Innovation and Management. 2022. Vol. 2, No. 1. DOI: <https://doi.org/10.54489/ijtim.v2i1.77>.
11. Alaidaros H., Omar M., Romli R. The State of the Art of Agile Kanban Method: Challenges and Opportunities. Independent Journal of Management & Production. 2021. Vol. 12. DOI: <https://doi.org/10.14807/ijmp.v12i8.1482>.
12. Anderson D. J. Kanban: Successful Evolutionary Change for Your Technology Business. Sequim : Blue Hole Press, 2010. 278 p.
13. Anderson D. J., Carmichael A. Essential Kanban Condensed. Seattle : Lean Kanban University Press, 2016. 102 p.
14. Bass L., Weber I., Zhu L. DevOps: A Software Architect's Perspective. Boston : Addison-Wesley Professional, 2015. 352 p.
15. Beck K., Beedle M., van Bennekum A., Cockburn A., Cunningham W., Fowler M., Grenning J., Highsmith J., Hunt A., Jeffries R., Kern J., Marick B., Martin R. C., Mellor S., Schwaber K., Sutherland J., Thomas D. Principles behind the Agile Manifesto. URL: <https://agilemanifesto.org/principles.html>.
16. Ćirić D., Gračanin D. Agile Project Management Beyond Software Industry. XVII International Scientific Conference on Industrial Systems (IS'17), Novi Sad, Serbia, October 4–6, 2017. Novi Sad : University of Novi Sad, Faculty of Technical Sciences, 2017. P. 332–337. URL: https://www.researchgate.net/publication/351811453_Agile_Project_Management_Beyond_Software_Industry.
17. Cole R., Scotcher E. Brilliant Agile Project Management: A Practical Guide to Using Agile, Scrum and Kanban. Harlow : Pearson Education Limited, 2015.

18. Dong H., Dacre N., Baxter D., Ceylan S. What is Agile Project Management? Developing a New Definition Following a Systematic Literature Review. *Project Management Journal*. 2024. Vol. 55, Issue 6. P. 668–688.
19. Ebert C., Gallardo G., Hernantes J., Serrano N. DevOps. *IEEE Software*. 2016. Vol. 33, Issue 3. P. 94–100. DOI: <https://doi.org/10.1109/MS.2016.68>.
20. Engelhardt N. Comparison of agile and traditional project management: simulation of process models. *Acta Academica Karviniensia*. 2019. Vol. 19, Issue 2. P. 15–27. DOI: <https://doi.org/10.25142/aak.2019.011>.
21. Fitzgerald B., Stol K.-J. Continuous software engineering: A roadmap and agenda. *Journal of Systems and Software*. 2017. Vol. 123. P. 176–189. DOI: <https://doi.org/10.1016/j.jss.2015.06.063>.
22. Forsgren N., Humble J., Kim G. *Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations*. Portland : IT Revolution Press, 2018. 288 p.
23. Hoda R., Noble J., Marshall S. Self-organizing roles on agile software development teams. *IEEE Transactions on Software Engineering*. 2013. Vol. 39, Issue 3. P. 422–444. DOI: <https://doi.org/10.1109/TSE.2012.30>.
24. Holvitie J., Leppänen V., Hyrynsalmi S. Technical debt and agile software development practices and processes: An industry practitioner survey. *Information and Software Technology*. 2018. Vol. 96. P. 141–160. DOI: <https://doi.org/10.1016/j.infsof.2017.11.015>.
25. Humble J., Farley D. *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*. Boston : Addison-Wesley Professional, 2010. 512 p.
26. Ilja S., Kailanto J., Saanila-Sotamaa M. From pyramid to communities: How pharma company reinvented themselves using Scrum. *Agile Alliance*. 2021. URL: <https://www.agilealliance.org/wp->

content/uploads/2021/06/S.Lilja_.J.Kailanto.M.Saanila-Sotamaa.From-pyramid-to-communities.pdf.

27. Khan M., Khan A., Khan F., Khan M., Whangbo T. Critical Challenges to Adopt DevOps Culture in Software Organizations: A Systematic Review. *IEEE Access*. 2022. Vol. 10. DOI: <https://doi.org/10.1109/ACCESS.2022.3145970>.

28. Kim G., Humble J., Debois P., Willis J. *The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations*. 2nd ed. Portland : IT Revolution Press, 2021. 528 p.

29. Koi-Akrofi J. Y., Matey H. A., Koi-Akrofi G. Understanding the characteristics, benefits and challenges of agile IT project management: a literature based perspective. *International Journal of Software Engineering & Applications*. 2019. Vol. 10, Issue 5. P. 25–44. DOI: <https://doi.org/10.5121/ijsea.2019.10502>.

30. Leite G., Vieira R. E. P., Cerqueira L., Maciel R. S. P., Freire S., Mendonça M. Technical Debt Management in Agile Software Development: A Systematic Mapping Study. *Proceedings of the XXIII Brazilian Symposium on Software Quality*. 2024. DOI: <https://doi.org/10.1145/3701625.3701669>.

31. Mohammad N. Secure DevOps Practices for Continuous Integration and Deployment in Fintech Cloud Environments. 2024. Vol. 1. P. 11–26.

32. Paasivaara M., Durasiewicz S., Lassenius C. Distributed Agile Development: Using Scrum in a Large Project. *IEEE International Conference on Global Software Engineering*. 2008. P. 87–95. DOI: <https://doi.org/10.1109/ICGSE.2008.38>.

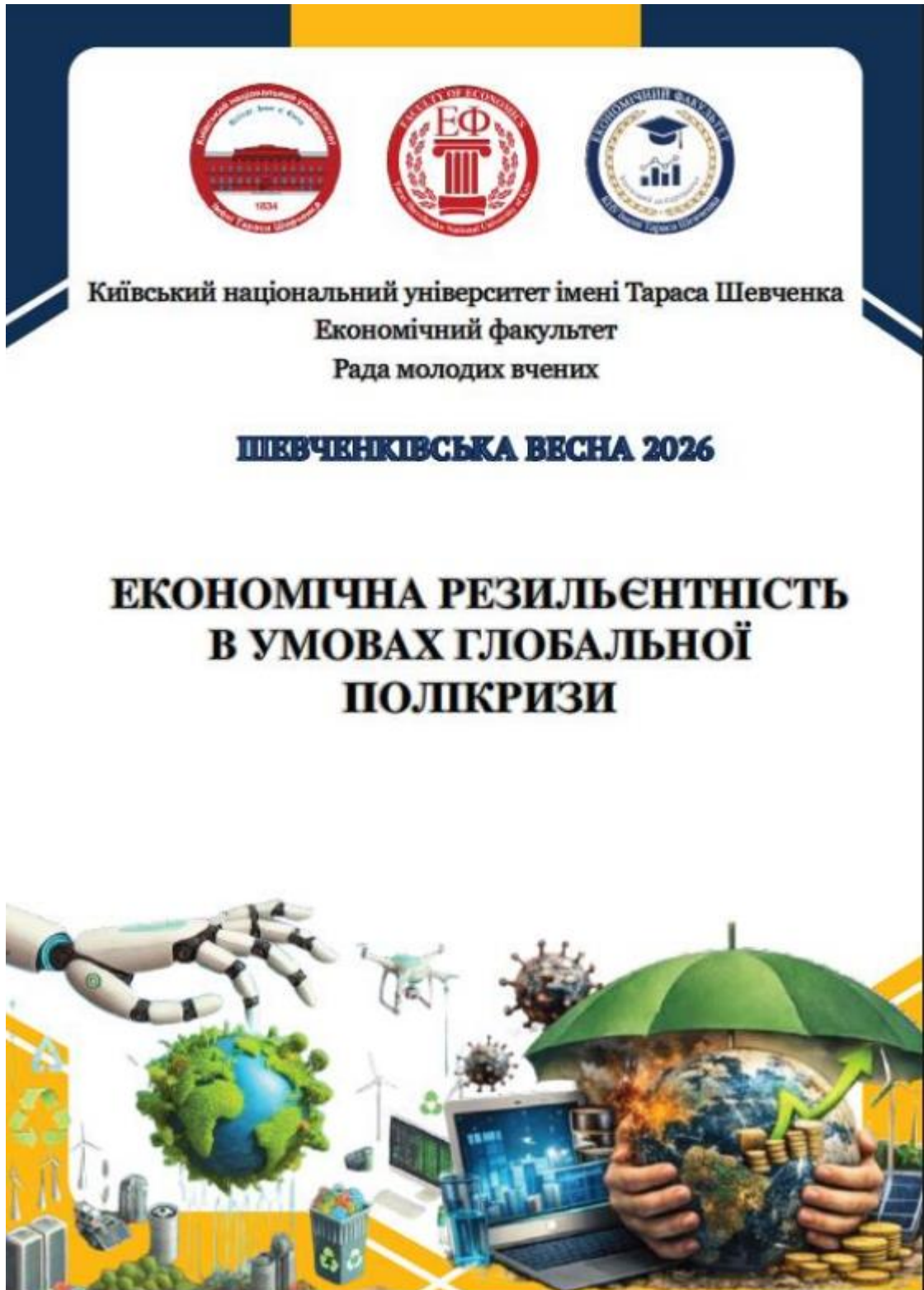
33. Pietrzak P. G., Cieciora M., Klimaszewski K. The use of traditional and agile Project Management methodologies in ICT. *Journal of Modern Science*. 2022. Vol. 49, Issue 2. P. 509–528. DOI: <https://doi.org/10.13166/jms/156463>.

34. Project Management Institute, Agile Alliance. *Agile Practice Guide*. Newtown Square : Project Management Institute, 2017. 210 p.

35. Rath S. P., Tripathy A., Naik S. A systematic literature review of agile software development. *Information and Software Technology*. 2025. DOI: <https://doi.org/10.1016/j.infsof.2025.107663>.
36. Sassa A., Almeida I., Pereira T., Oliveira M. Scrum: A Systematic Literature Review. *International Journal of Advanced Computer Science and Applications*. 2023. Vol. 14. DOI: <https://doi.org/10.14569/IJACSA.2023.0140420>.
37. Schwaber K., Sutherland J. *The Scrum Guide. The Definitive Guide to Scrum: The Rules of the Game*. 2020. URL: <https://scrumguides.org/scrum-guide.html>.
38. Shrivastava A., Jaggi I., Katoch N., Deepali, Gupta D., Gupta S. A Systematic Review on Extreme Programming. *Journal of Physics: Conference Series*. 2021. Vol. 1969. Article 012046. DOI: <https://doi.org/10.1088/1742-6596/1969/1/012046>.
39. Šmite D., Moe N. B., Hildrum J., Huerta J. G., Mendez D. Work-from-home is here to stay: Call for flexibility in post-pandemic work policies. *Journal of Systems and Software*. 2023. Vol. 195. Article 111552. DOI: <https://doi.org/10.1016/j.jss.2022.111552>.
40. Smith R. The Role of Agile in Pharma Digital Transformation. *Journal of Pharmaceutical Technology*. 2023. DOI: <https://doi.org/10.1177/10408313221117592>.
41. Stray V., Moe N. B., Hoda R. Autonomous agile teams: Challenges and future directions for research. *Proceedings of the 19th International Conference on Agile Software Development: Companion*. 2018. DOI: <https://doi.org/10.1145/3234152.3234182>.
42. Sutherland J., Sutherland J. J. *Scrum: The Art of Doing Twice the Work in Half the Time*. New York : Currency, 2014. 256 p.
43. Thomas D., Higgins D. A Model for Lean and Agile Integration in Pharmaceuticals. *International Journal of Operations Management*. 2019. DOI: <https://doi.org/10.1108/IJOM-10-2018-0379>.

44. Vacanti D. Actionable Agile Metrics for Predictability: An Introduction. Seattle : ActionableAgile Press, 2015. 314 p.
45. Vedsmand T., Kielgast S., Cooper R. G. Integrating agile with Stage-Gate®: How new Agile-Scrum methods lead to faster and better innovation. Innovation Management. 2016. URL: <http://www.innovationmanagement.se/2016/08/09/integrating-agile-with-stage-gate/>.
46. Waja G., Shah J., Nanavati P. Agile Software Development. International Journal of Engineering Applied Sciences and Technology. 2021. Vol. 5. DOI: <https://doi.org/10.33564/IJEAST.2021.v05i12.011>.
47. Wilkes B., Fox C., Ralph P. A Framework for Automating the Measurement of DevOps Metrics. IEEE International Conference on Software Maintenance and Evolution. 2022. URL: https://bw.codexwilkes.com/_media/icsme.pdf.
48. Agile Alliance. Remote Onboarding to Agile Teams. URL: <https://agilealliance.org/resources/experience-reports/remote-onboarding-to-agile-teams/>.
49. Agile Alliance. Webinar Recap: Tools for Effective Remote Agile Teams. 2020. URL: <https://agilealliance.org/webinar-recap-tools-for-effective-remote-agile-teams/>.
50. DevOps Institute. DevOps in Fintech. URL: <https://www.devopsinstitute.com/devops-in-fintech/>.
51. Digital.ai. 17th Annual State of Agile Report. 2023. URL: <https://digital.ai/resource-center/analyst-reports/state-of-agile-report>.
52. DORA. DORA's Software Delivery Performance Metrics. URL: <https://dora.dev/guides/dora-metrics/>.
53. Fowler M. Continuous Integration. 2024. URL: <https://martinfowler.com/articles/continuousIntegration.html>.

54. Fowler M. Technical Debt. 2019. URL: <https://martinfowler.com/bliki/TechnicalDebt.html>.
55. Google Cloud, DORA. Accelerate State of DevOps Report 2024. URL: <https://dora.dev/research/2024/dora-report/>.
56. JustMarkets. URL: <https://justmarkets.com>.
57. JustMarkets Careers. URL: <https://careers.justmarkets.com>.
58. JustMarkets на DOU. URL: <https://jobs.dou.ua/companies/justmarkets/>.
59. Kanban Guides. The Kanban Guide. 2020. URL: <https://kanbanguides.org/the-kanban-guide/>.
60. Kanban University. The Official Guide to the Kanban Method. URL: <https://kanban.university/kanban-guide/>.
61. Rehkopf M. Kanban vs. Scrum: Which agile are you? Atlassian. URL: <https://www.atlassian.com/agile/kanban/kanban-vs-scrum>.
62. Scrum.org. Kanban Guide for Scrum Teams. 2021. URL: <https://www.scrum.org/resources/kanban-guide-scrum-teams>.
63. Sutherland J., Scrum Inc. The Scrum@Scale Guide. Version 2.1. 2022. URL: <https://www.scrumatscale.com/scrum-at-scale-guide/>.
64. Selinder E. DevOps KPIs: Software Delivery Performance Metrics. Stockholm : KTH Royal Institute of Technology, 2024. URL: <https://www.diva-portal.org/smash/get/diva2:1936481/FULLTEXT01.pdf>.



Закопорожна Наталія Юрїївна	СТРАТЕГІЧНЕ УПРАВЛІННЯ РОЗВИТКОМ ПІДПРИЄМСТВА В УМОВАХ ВІЙНИ	61
Зімін Анатолій Володимирович	ІНКЛЮЗИВНИЙ МЕНЕДЖМЕНТ ЯК СТРАТЕГІЧНИЙ ІМПЕРАТИВ: РЕІНТЕГРАЦІЯ ВІТЕРАНІВ ТА ОСІБ З ІНВАЛІДНІСТЮ В УМОВАХ ВОЄННОЇ ЕКОНОМІКИ УКРАЇНИ	63
Зиченко Сергій Олександрович	ТРАНСФОРМАЦІЯ РИНКУ ПРАЦІ УКРАЇНИ В УМОВАХ ВІЙНИ: ВИКЛИКИ ТА СТРУКТУРНІ ДИСПРОПОРЦІЇ	66
Ігаско Олександра Олександрівна	МЕНЕДЖМЕНТ АГЕНЦІЙ НЕРУХОМОСТІ В УМОВАХ ВІЙНИ ТА ГЛОБАЛЬНОЇ ПОЛІКРИЗИ: ГАЛУЗЕВІ ТРЕНДИ ТА СТРАТЕГІЇ АДАПТАЦІЇ	68
Кафідов Валерій Валерійович	ШЛЯХИ ВДОСКОНАЛЕННЯ УПРАВЛІННЯ SEO-МАРКЕТИНГОВИМИ	69
Квік Олександр Іванович	ТРАНСФОРМАЦІЯ МОДЕЛЕЙ КОМАНДНОЇ ВЗАЄМОДІЇ ПІД ВПЛИВОМ БЕЗПЕКОВИХ ТА ІНФРАСТРУКТУРНИХ ВИКЛИКІВ ВОЄННОГО СТАНУ	71
Клюєцька В.В.	ТРАНСФОРМАЦІЯ ОПЕРАЦІЙНОЇ МОДЕЛІ УПРАВЛІННЯ МАРКЕТИНГОМ В УМОВАХ ВОЄННИХ РИЗИКІВ ТА ЕНЕРГОКРИЗИ	72
Коваленко Артем Анатолійович	МОДЕЛІ УПРАВЛІННЯ СТРАТЕГІЧНИМИ ЗМІНАМИ ІТ-ПІДПРИЄМСТВ В УМОВАХ ЦИФРОВОЇ ТРАНСФОРМАЦІЇ	73
Костик Ольга Василівна	СТРАТЕГІЧНЕ УПРАВЛІННЯ ДІДЖИТАЛІЗАЦІЄЮ ЯК ФАКТОР ІННОВАЦІЙНОГО РОЗВИТКУ ПІДПРИЄМСТВА	75
Котларенко Вероніка Анатоліївна	ОСОБЛИВОСТІ УПРАВЛІННЯ РЕКЛАМНОЮ ДІЯЛЬНІСТЮ В ІНТЕРНЕТІ ДЛЯ ПІДПРИЄМСТВ АГРАРНОГО СЕКТОРУ	76
Кощіа Ірина Миколаївна	УПРАВЛІННЯ МАРКЕТИНГОВОЮ ДІЯЛЬНІСТЮ ПОСЕРЕДНИЦЬКОЇ ПЛАТФОРМИ У СФЕРІ ТУРИЗМУ В КРИЗОВИХ УМОВАХ	78
Крикун Олександр Олександрович	УДОСКОНАЛЕННЯ ПРОЦЕСІВ ЗАДОВОЛЕННЯ ПОТРЕБИ У ПЕРСОНАЛІ ПІДПРИЄМСТВА НА ПРИКЛАДІ ТОВ «АРМА МОТОРС КИЇВ»	79
Кудель Юлія Іванівна	ТАЛАНТ-МЕНЕДЖМЕНТ ЯК МЕХАНІЗМ РОЗВИТКУ ТА УТРИМАННЯ ПЕРСОНАЛУ ПІДПРИЄМСТВА: НА ПРИКЛАДІ МХП	81
Куря Ярослав Андрійович	РЕЗУЛЬТАНТИСТІВ ЯК ОСНОВА МОТИВАЦІЇ ПЕРСОНАЛУ В УМОВАХ ВІЙНИ	83
Кулішова Марія Юрїївна	ЦИФРОВА ТРАНСФОРМАЦІЯ МЕНЕДЖМЕНТУ ВИРОБНИЧОГО ПІДПРИЄМСТВА В УМОВАХ ПОЛІКРИЗИ	85
Курієвський Дмитро Олександрович	ПАРТНЕРСЬКІ КОНФІГУРАЦІЇ ПІДПРИЄМСТВ АГРАРНОЇ ГАЛУЗІ УКРАЇНИ ЯК ІНСТРУМЕНТ МЕНЕДЖМЕНТУ В УМОВАХ ВІЙНИ ТА ГЛОБАЛЬНОЇ ПОЛІКРИЗИ	86
Лончар Сергій Сергійович	ВПЛИВ ДІДЖИТАЛІЗАЦІЇ НА РОЗВИТОК МАРКЕТИНГОВОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДПРИЄМСТВА	88
Мазанович Юлія Віталіївна	ОСОБЛИВОСТІ ПОБУДОВИ СИСТЕМИ КРІ ВИРОБНИЧОГО ПІДПРИЄМСТВА В УМОВАХ ВОЄННО-ЕКОНОМІЧНОЇ НЕСТАБІЛЬНОСТІ	89
Майборода Маргарита Миколаївна	ОСОБЛИВОСТІ ПРОГРАМ ОНБОРДІНГУ ПЕРСОНАЛУ В УМОВАХ ВІЙНИ	91
Мазарченко Вікторія Віталіївна	АКТИВАЦІЯ СТРАТЕГІЧНОГО ПОТЕНЦІАЛУ МОЛОКОПЕРЕРОБНИХ ПІДПРИЄМСТВ УКРАЇНИ ЯК НАПРЯМ ЗАБЕЗПЕЧЕННЯ ЇХ СТІЙКОСТІ В УМОВАХ ВОЄННОГО СТАНУ	92
Меженкова Дар'я Ігорівна	ВПЛИВ OTT-ПЛАТФОРМ НА ТРАНСФОРМАЦІЮ МЕДІАПРОСТОРУ ТА РЕКЛАМНОГО РИНКУ В УКРАЇНІ	93
Могилнич Анастасія Іванівна	УПРАВЛІННЯ ОРГАНІЗАЦІЙНИМИ ЗМІНАМИ В УМОВАХ ВІЙНИ ТА ГЛОБАЛЬНОЇ ПОЛІКРИЗИ	94
Мороз Олександр Віталійович	ОСОБЛИВОСТІ ПРОСУВАННЯ ІННОВАЦІЙНОЇ ПРОДУКЦІЇ НА РИНОК NORESA	95
Мушин Тетяна Леонідівна	ІНСТРУМЕНТИ УПРАВЛІННЯ ЯКІСТЮ АГРОПРОМИСЛОВОГО КОМПЛЕКСУ УКРАЇНИ В УМОВАХ ЦИФРОВІЗАЦІЇ ЕКОНОМІКИ	96
Наконечна Софія Ігорівна	УПРАВЛІННЯ БРЕНДОМ РОБОТОДАВЦЯ В КОНТЕКСТІ ЗАБЕЗПЕЧЕННЯ ОРГАНІЗАЦІЙНОЇ СТІЙКОСТІ ПІДПРИЄМСТВА В КРИЗОВИХ УМОВАХ	98
Недзківський Олександр Сергійович	ІНТЕГРАЦІЯ «ЗЕЛЕНИХ» ПРАКТИК ТА КОРПОРАТИВНОЇ СОЦІАЛЬНОЇ ВІДПОВІДАЛЬНОСТІ В СИСТЕМУ МОТИВАЦІЇ ЯК ДРАЙВЕР БЕЗПЕРЕРВНОГО ЕКОНОМІЧНОГО ЗРОСТАННЯ ОРГАНІЗАЦІЇ	100
Огер Даніло Костянтинівич	АДАПТАЦІЯ СИСТЕМИ МОТИВАЦІЇ ПЕРСОНАЛУ ОРГАНІЗАЦІЇ ДО УМОВ ВІЙНИ ТА ГЛОБАЛЬНОЇ ПОЛІКРИЗИ	101
Олексієнко Богдан Олександрович	СУЧАСНІ ТРЕНДИ ІННОВАЦІЙНОГО МЕНЕДЖМЕНТУ В УМОВАХ ЦИФРОВОЇ ТРАНСФОРМАЦІЇ ЕКОНОМІКИ	102
Onyshchenko Alina Volodymyrivna	THE IMPACT OF THE WAR ON THE PERSONNEL MANAGEMENT SYSTEM OF "SOFTSERVE" LLC	103
Палаш Олег Олександрович	ТРАНСФОРМАЦІЯ УПРАВЛІННЯ РЕКЛАМНИМИ ПРОЕКТАМИ АГЕНТСТВ НЕРУХОМОСТІ В УМОВАХ ВОЄННОГО СТАНУ В УКРАЇНІ	104
Паращич Христина Тарасівна	РАДІКАЛЬНА ІНКЛЮЗИВІА: СТРАТЕГІЯ ІНТЕГРАЦІЇ ВІТЕРАНІВ У БРЕНД РОБОТОДАВЦЯ ФІНАНСОВИХ УСТАНОВ ПІД ЧАС ВІЙНИ	105
Пісарев Ярослав Романович	СУЧАСНІ ПІДХОДИ ДО УПРАВЛІННЯ ПРОЕКТАМИ У ІТ-ПІДПРИЄМСТВАХ	107
Подолян Роман Васильович	ОСОБЛИВОСТІ МАРКЕТИНГУ В ІНДУСТРІЇ СТОРТУ ТА РОЗВАГ ЯК ІНТЕГРОВАНОЇ СИСТЕМИ УПРАВЛІННЯ ЦІННІСТЮ В УМОВАХ НЕВІЗНАЧЕНОСТІ	108
Подняков Іван Володимирович	СТРАТЕГІЧНИЙ КОНТРОЛІНГ ПІДПРИЄМСТВ: СУТНІСТЬ, ОСОБЛИВОСТІ, ІНСТРУМЕНТИ	109
Прилуцька Тетяна Юрїївна	ЖІНОЧЕ ЛІДЕРСТВО У ЗАБЕЗПЕЧЕННІ ЕКОНОМІЧНОЇ СТІЙКОСТІ БІЗНЕСУ ПІД ЧАС ВОЄННОГО СТАНУ	111

СУЧАСНІ ПІДХОДИ ДО УПРАВЛІННЯ ПРОЕКТАМИ У ІТ-ПІДПРИЄМСТВАХ

Пісарев Ярослав Романович

Київський національний університет імені Тараса Шевченка
 II курс магістратури, ОП «Менеджмент організацій і адміністрування»
 Науковий керівник: к. е. н. доц. Приймак В.М.

MODERN APPROACHES TO PROJECT MANAGEMENT IN IT ENTERPRISES

The study examines the use of agile methodologies in managing IT projects, focusing on frameworks such as Scrum, Kanban, Lean, Extreme Programming, and others. It analyzes their impact on team collaboration, iterative development, adaptability to changing requirements, and risk management. The research identifies how combining structured iterative approaches with practices aimed at improving quality and efficiency can optimize project workflows and enhance the competitiveness and resilience of IT companies operating in dynamic and uncertain environments.

Світовий ринок ІТ-послуг стрімко зростає, і його обсяг у 2024 році досягнув 1218,6 млрд доларів США. За прогнозами IMARC Group, до 2033 року ринок сягне 2289,3 млрд доларів, демонструючи середньорічне зростання на рівні 7,26% у період 2025–2033 років. Провідну позицію займає Північна Америка з часткою 36,5%, що зумовлено високим попитом на цифрові послуги у сферах охорони здоров'я, фінансів, ритейлу та активними інвестиціями у цифрову трансформацію.

Український ІТ-сектор, попри повномасштабну війну, залишається одним із найдинамічніших в економіці країни та забезпечує 38% експорту послуг. Вітчизняні ІТ-компанії демонструють високу стійкість до кризових умов, адаптуючись до змін через впровадження інноваційних управлінських і маркетингових рішень. Проте, водночас, галузь стикається з серйозними викликами — від релокації працівників і загроз інфраструктурі до зростаючої конкуренції з боку Індії та Польщі [2].

У цих умовах особливого значення набуває ефективне управління проектами в ІТ-компаніях. Швидко еволюція технологій, зокрема штучного інтелекту, кібербезпеки, блокчейну та великих даних, потребує впровадження гнучких та адаптивних підходів до управління, що здатні забезпечити стійкість і конкурентоспроможність українських ІТ-підприємств у глобальному середовищі.

Особливості управління ІТ-проектами пов'язані з високою складністю, динамічністю та невизначеністю процесів розробки в сфері інформаційних технологій. ІТ-проекти мають нестандартний життєвий цикл, що включає тестування, гарантійний та післягарантійний супровід, а також характерна необхідність чіткого визначення вимог на ранніх етапах, незважаючи на їхню рухливість і змінність. Важливою рисою є необхідність швидкого внесення змін під час розробки, що часто спричиняє відставання від початкових планів.

Для управління ІТ-проектами характерне використання гнучких методологій (Agile), зокрема Scrum, які дають змогу розбивати роботу на короткі ітерації, забезпечують швидку адаптацію до змін і тісну комунікацію між командою та замовником (табл. 1). Особлива увага приділяється управлінню ризиками через високу ймовірність змін у вимогах, технологіях і пріоритетах, що відрізняє ІТ-проекти від традиційних бізнес-проектів. Матрична організаційна структура і координація робіт сприяють ефективній взаємодії між різними фахівцями в команді [3].

Таблиця 1

Підходи до управління проектами у ІТ-галузі

Методологія	Основний підхід	Ключові характеристики	Переваги	Недоліки
Waterfall (Каскадна)	Послідовне виконання етапів	Жорстка послдовність фаз (аналіз, дизайн, розробка, тестування, впровадження)	Чітке планування, прозорі етапи	Слабка гнучкість, складність внесення змін
Agile	Ітеративний, гнучкий	Короткі ітерації (спринти), постійна комунікація з замовником, адаптація до змін	Висока адаптивність, швидке реагування	Можливість хаосу без чіткого управління
Scrum	Фреймворк Agile	Роль: Product Owner, Scrum Master, команда; спринти 2-4 тижні; шкелетні стандарти	Чітка організація роботи, швидкий фідбек	Потребує досвідченої команди
Kanban	Візуалізація потоку робіт	Дошка Kanban, ліміти на кількість завдань у роботі, без фіксованих ітерацій	Прозорість процесів, гнучкість	Може призводити до відсутності структури
Lean	Мінімізація витрат	Оптимізація процесів, усунення непотрібних дій, швидка доставка цінності	Ефективність, зниження витрат	Вимагає культури постійного покращення
Extreme Programming (XP)	Agile-практики з акцентом на якість коду	Парне програмування, тестування, часті реалізи	Висока якість, зниження дефектів	Потребує сильної дисципліни команди
Crystal	Гнучкий, орієнтований на команду	Різні варіанти (Crystal Clear, Orange тощо) залежно від розміру і складності проекту	Адаптується до розміру проекту і команди	Менш формалізований, потребує досвіду
Feature-Driven Development (FDD)	Орієнтація на функції	Моделювання, побудова списку функцій, ітеративна розробка	Чіткий фокус на функціональність	Менш підходить для проектів з невизначеними вимогами
DSDM (Dynamic Systems Development Method)	Agile з чіткою структурою	Фокус на бізнес-цінності, фіксований час і ресурси, ітеративна розробка	Чіткі ролі, баланс між гнучкістю і контролем	Вимагає залучення замовника

Джерело: складено автором на основі даних: [1, с. 92-93; 4]

Отже, ефективне управління ІТ-проектами передбачає застосування гнучких методологій, що забезпечують адаптивність і швидке реагування на зміни. Враховуючи складність і динамічність ІТ-середовища, ітеративні підходи застосовуються з орієнтацією на мінімізацію витрат і підвищення якості для підвищення продуктивності та конкурентоспроможності компаній. Вибір методології повинен відповідати особливостям проекту і рівню компетенції команди, що дасть змогу оптимізувати процеси розробки і забезпечити успішну реалізацію в умовах високої невизначеності.

1. Давидюк Н., Шулик Ю., Качан О. Сучасні підходи до управління проектною діяльністю ІТ-компаній. Наукові записки Національного університету «Острозька академія». Серія «Економіка». 2021. Вип. 22 (50). С. 88–94. ISSN 2311-5149.

2. Полюк О., Тивончук П. Стратегічний маркетинговий аналіз ринку ІТ-послуг в Україні. Економіка та суспільство. 2025. Вип. 71. DOI: <https://doi.org/10.32782/2524-0072/2025-71-58>

3. Сметанюк О. А., Бондарчук А. В. Особливості системи управління проектами в ІТ-компаніях. Агросвіт. 2020. № 10. С. 105–111. DOI: [10.32702/2306-6792.2020.10.105](https://doi.org/10.32702/2306-6792.2020.10.105).

4. Храпків О., Кіндрат О., Чопей Р. Управління проектами в ІТ-галузі: методика, інструменти та керування ризиками. Економіка та суспільство. 2023. № 55. С. 110. DOI: <https://doi.org/10.32782/2524-0072/2023-55-110>

Деталізоване оцінювання ефективності застосування гнучких методів управління проектами у JustMarkets Tech за підкритеріями

Б.1. Оцінювання продуктового планування

Підкритерій	Бал
Наявність дорожньої карти розвитку продукту	4
Наявність переліку вимог і завдань	4
Зв'язок вимог із потребами користувачів	4
Зв'язок вимог із бізнес-цілями	4
Регулярність перегляду пріоритетів	2

Розрахунок середнього бала $(4 + 4 + 4 + 4 + 2) / 5 = 3,6$

Середній бал за блоком продуктового планування становить 3,6.

Б.2. Оцінювання пріоритезації завдань

Підкритерій	Бал
Зрозумілість критеріїв відбору завдань	4
Врахування користувацької цінності	4
Врахування технічної складності	4
Врахування регуляторних обмежень	3
Баланс між новими функціями і технічним боргом	2

Розрахунок середнього бала: $(4 + 4 + 4 + 3 + 2) / 5 = 3,4$

Середній бал за блоком пріоритезації становить 3,4.

Б.3. Оцінювання міжфункціональної взаємодії

Підкритерій	Бал
Участь Product та Engineering у плануванні	4
Залучення Trading до змін торгової платформи	4
Участь Legal & Compliance у регуляторно чутливих змінах	4
Використання інформації від Customer Support	3
Участь Back Office в операційних змінах	3

Розрахунок середнього бала: $(4 + 4 + 4 + 3 + 3) / 5 = 3,6$

Середній бал за блоком міжфункціональної взаємодії становить 3,6.

Б.4. Оцінювання потоку завдань

Підкритерій	Бал
Середній час проходження завдання	3
Кількість заблокованих завдань	3
Кількість повернень на доопрацювання	3
Рівномірність навантаження	2
Обсяг незавершеної роботи	3

Розрахунок середнього бала: $(3 + 3 + 3 + 2 + 3) / 5 = 2,8$

Середній бал за блоком потоку завдань становить 2,8.

Б.5. Оцінювання технічного виконання

Підкритерій	Бал
Якість технічного оцінювання	4
Наявність тестування	4
Перегляд коду	4
Використання CI/CD	4
Контроль релізів	4
Стабільність після запуску	3

Розрахунок середнього бала: $(4 + 4 + 4 + 4 + 4 + 3) / 6 = 3,8$

Середній бал за блоком технічного виконання становить 3,8.

Б.6. Оцінювання результативності релізів

Підкритерій	Бал
Частота релізів	3
Час проходження зміни до запуску	3
Частка невдалих змін	2
Час відновлення після інциденту	3
Кількість помилок після запуску	3

Розрахунок середнього бала: $(3 + 3 + 2 + 3 + 3) / 5 = 2,8$

Середній бал за блоком результативності релізів становить 2,8.

Б.7. Оцінювання клієнтського результату

Підкритерій	Бал
Задоволеність користувачів	4
Кількість повторюваних звернень	3
Залученість користувачів	4
Утримання клієнтів	3
Конверсія	3
Якість взаємодії з продуктом	4

Розрахунок середнього бала: $(4 + 3 + 4 + 3 + 3 + 4) / 6 = 3,5$

Середній бал за блоком клієнтського результату становить 3,5.

Б.8. Оцінювання подальшого вдосконалення

Підкритерій	Бал
Аналіз результатів після релізу	4
Оновлення переліку завдань	3
Коригування дорожньої карти	3
Фіксація проблем процесу	2
Управлінські рішення за підсумками оцінювання	3

Розрахунок середнього бала: $(4 + 3 + 3 + 2 + 3) / 5 = 3,0$.

АНОТАЦІЯ

Пісарев Я. Р. Гнучке управління проєктами ІТ-підприємства.

Кваліфікаційна робота магістра зі спеціальності 073 «Менеджмент», освітньо-наукової програми «Менеджмент організацій і адміністрування». Київський національний університет імені Тараса Шевченка, Київ, 2026.

Метою роботи є обґрунтування теоретичних засад, аналіз практики застосування та розроблення напрямів удосконалення гнучкого управління проєктами ІТ-підприємства на прикладі JustMarkets Tech. Роботу присвячено питанням організації гнучкого управління ІТ-проєктами в умовах високої змінності вимог, швидкого оновлення цифрових продуктів, потреби у регулярному зворотному зв'язку та підвищенні якості програмних рішень. У роботі розкрито сутність, принципи, переваги, обмеження та бізнес-цілі гнучкого управління проєктами, охарактеризовано методичні підходи Scrum, XP, Kanban, Lean Agile і DevOps. Проаналізовано організаційну побудову системи управління ІТ-проєктами у JustMarkets Tech, проведено діагностику ефективності застосування гнучких методів за продуктовими, організаційними, технічними та клієнтськими критеріями. Визначено напрями оптимізації процесів гнучкого управління через удосконалення потоку завдань, релізних метрик, післярелізного аналізу, управління технічним боргом, клієнтського зворотного зв'язку та цифрової координації віддалених команд. Обґрунтовано впровадження інтегрованої моделі гнучкого управління ІТ-проєктами у JustMarkets Tech та оцінено економічну ефективність запропонованого проєкту.

Ключові слова: гнучке управління проєктами, Agile, Scrum, Kanban, Lean Agile, DevOps, ІТ-підприємство, JustMarkets Tech, релізні метрики, технічний борг.

ABSTRACT

Yaroslav R. Pisarev. Agile Project Management of an IT Enterprise.

Master's thesis in specialty 073 "Management", educational and scientific programme "Management of Organizations and Administration". Taras Shevchenko National University of Kyiv, Kyiv, 2026.

The purpose of the thesis is to substantiate the theoretical foundations, analyze the practical application, and develop directions for improving agile project management at an IT enterprise based on the case of JustMarkets Tech. The thesis addresses the organization of agile IT project management under conditions of changing requirements, rapid renewal of digital products, the need for regular feedback, and improved quality of software solutions. The study examines the essence, principles, advantages, limitations, and business objectives of agile project management, as well as methodological approaches such as Scrum, XP, Kanban, Lean Agile, and DevOps. The organizational structure of IT project management at JustMarkets Tech is analyzed, and the effectiveness of agile methods is assessed according to product, organizational, technical, and customer-related criteria. The thesis identifies directions for optimizing agile project management processes through the improvement of task flow, release metrics, post-release analysis, technical debt management, customer feedback, and digital coordination of remote teams. An integrated model of agile IT project management for JustMarkets Tech is substantiated, and the economic efficiency of the proposed project is evaluated.

Keywords: agile project management, Agile, Scrum, Kanban, Lean Agile, DevOps, IT enterprise, JustMarkets Tech, release metrics, technical debt.