

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально науковий інститут комп'ютерних наук та управління проєктами

Кафедра управління проєктами

«Допущений до захисту»

Завідувач кафедри: Чернов С.К.

 (підпис)

« 09 » грудня 2025р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти «магістр»

на тему:

Методи удосконалення інструментів формування команди розробки ПЗ в Agile-
проєктах

Виконав: студент 6171м групи

 – Вучевич М. А.
(підпис)

Керівник роботи: д.т.н., доцент

–  Чернова Лб.С.
(підпис)

Миколаїв – 2025 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова
Навчально науковий інститут комп'ютерних наук та управління проектами
Кафедра управління проектами
Спеціальність 122 Комп'ютерні науки
Освітня програма «Управління проектами»

«ЗАТВЕРДЖУЮ»

Гарант освітньої програми
д.т.н., проф. Чернова Л.С.

 (підпис)

«09» грудня 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти «магістр»

Студенту **Вучевичу Микиті Альбертовичу**

1. Тема роботи: Методи удосконалення інструментів формування команди розробки ПЗ в Agile-проектах

Керівник роботи: д.т.н., доц. Чернова Лб.С.

Затверджені наказом ректора № 1406-уч. від «04» грудня 2025 року

2. Термін подання роботи: до 09.12.2025р.

3. Вихідні дані по роботі: фахова література, довідники з управління проектами, матеріали конференцій, репозитарій університету

4. Перелік питань, що належать до розробки (найменування розділів)

Розділ 1. Теоретичні основи управління командою ІТ-проекту

Розділ 2. Аналіз процесу формування команди у межах Agile-проекту стартапу

Розділ 3. Напрями удосконалення інструментів формування команди розробки ПЗ в Agile-проектах

Розділ 4. Охорона праці

Розділ 5. Охорона навколишнього середовища

5. Перелік презентаційних матеріалів виконаний в програмі Power Point

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Розділ 4. Охорона праці	Мозговий А.М., доцент		
Розділ 5. Охорона навколишнього середовища	Мозговий А.М., доцент		

7. Дата видачі завдання 16.09.2025р.

КАЛЕНДАРНИЙ ПЛАН

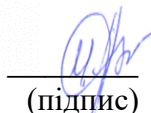
№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Вивчення літературних джерел з предмету дослідження	Вересень 2025	виконано
2	Складання розгорнутого плану магістерської роботи та ознайомлення керівника з планом кваліфікаційної роботи	Вересень 2025	виконано
3	Написання розділу 1	Вересень 2025	виконано
4	Написання розділу 2	Жовтень 2025	виконано
5	Написання розділу 3	Листопад 2025	виконано
5	Написання розділу 4	Листопад 2025	виконано
6	Написання розділу 5	Листопад 2025	виконано
7	Оформлення магістерської роботи	Грудень 2025	виконано
8	Передача магістерської роботи рецензенту для рецензування	Грудень 2025	виконано
9	Передача магістерської роботи науковому керівникові для написання відгуку	Грудень 2025	виконано
10	Попередній захист магістерської роботи	09.12.2025	виконано
11	Захист магістерської роботи	25.12.2025	виконано

Студент



Вучевич М.А.
(підпис)

Керівник роб



(підпис)

Чернова ЛБ.С.

АНОТАЦІЯ

Вучевич М.А. Методи удосконалення інструментів формування команди розробки ПЗ в Agile-проектах. Кваліфікаційна робота зі спеціальності 122 «Комп'ютерні науки», освітньої програми «Управління проектами». Миколаїв: НУК ім. адм. Макарова, 2025. 70 сторінок.

Магістерська робота присвячена дослідженню процесу формування та розвитку команд розробки програмного забезпечення в умовах застосування Agile-методологій. Проаналізовано теоретичні засади командної роботи в IT-проектах, розглянуто основні принципи та інструменти гнучких методологій управління, а також досліджено сучасні підходи до формування Agile-команд. На основі аналітичного дослідження діяльності умовного Agile-проекту розроблено комплекс методів удосконалення інструментів формування команди розробки програмного забезпечення. Запропоновані рішення спрямовані на підвищення ефективності командної взаємодії, оптимізацію розподілу навантаження, покращення комунікаційних процесів і зниження ризиків професійного вигорання. Обґрунтовано практичну доцільність впровадження запропонованих методів та оцінено очікувані результати їх застосування.

У роботі також розглянуто питання охорони праці в IT-проектах та екологічні аспекти діяльності Agile-команд у контексті принципів сталого розвитку. Отримані результати мають практичну цінність і можуть бути використані в діяльності IT-компаній, стартапів і команд розробки програмного забезпечення для підвищення ефективності управління Agile-проектами.

Ключові слова: Agile, команда розробки програмного забезпечення, командоутворення, Scrum, компетенції, командна взаємодія, психологічна безпека, управління проектами.

ANNOTATION

Vuchevych Mykyta. Methods of improving software development team formation tools in Agile projects. Qualification work on specialty 122 "Computer Science", educational program "Project Management". Mykolaiv: NUK named after Adm. Makarova, 2025. 70 pages.

The master's thesis is devoted to the research of the process of formation and development of software development teams in the conditions of application of Agile methodologies. The theoretical principles of teamwork in IT projects were analyzed, the main principles and tools of flexible management methodologies were considered, and modern approaches to the formation of Agile teams were also investigated. Based on an analytical study of the activity of a conditional Agile project, a set of methods for improving software development team formation tools has been developed. The proposed solutions are aimed at increasing the effectiveness of team interaction, optimizing the distribution of the load, improving communication processes and reducing the risks of professional burnout. The practical expediency of implementing the proposed methods was substantiated and the expected results of their application were evaluated.

The paper also examines the issue of occupational health and safety in IT projects and the environmental aspects of Agile teams' activities in the context of sustainable development principles. The obtained results have practical value and can be used in the activities of IT companies, startups and software development teams to improve the efficiency of Agile project management.

Keywords: Agile, software development team, team building, Scrum, competencies, team interaction, psychological safety, project management.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ УПРАВЛІННЯ КОМАНДОЮ ІТ-ПРОЄКТУ	10
1.1. Командна робота в ІТ-проектах: сутність, специфіка та моделі взаємодії..	10
1.2. Вплив Agile-методологій на процес організації командної роботи.....	16
1.3. Інструменти формування команди в Agile	19
1.4. Проблеми та обмеження сучасних інструментів формування команди	21
Висновки	22
РОЗДІЛ 2. АНАЛІЗ ПРОЦЕСУ ФОРМУВАННЯ КОМАНДИ У МЕЖАХ AGILE- ПРОЄКТУ СТАРТАПУ	23
2.1. Загальна характеристика умовного стартапу та середовища його діяльності.....	23
2.2. Аналіз процесів роботи команди в межах Agile-підходів	26
2.3. Аналіз компетенцій команди та розподілу ролей.....	27
2.4. Оцінка ефективності командної взаємодії	29
2.5. Виявлення проблем формування та роботи команди.....	31
Висновки	33
РОЗДІЛ 3. НАПРЯМИ УДОСКОНАЛЕННЯ ІНСТРУМЕНТІВ ФОРМУВАННЯ КОМАНДИ РОЗРОБКИ ПЗ В AGILE-ПРОЄКТАХ	35
3.1. Обґрунтування напрямів удосконалення інструментів формування команди розробки ПЗ в Agile-проектах.....	35
3.2. Розроблення інструментів та моделей удосконалення процесу формування Agile-команди	39
3.3. Оцінка ефективності запропонованих інструментів удосконалення процесу формування Agile-команди	43
Висновки	45
РОЗДІЛ 4. ОХОРОНА ПРАЦІ	47
РОЗДІЛ 5. ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА	56
ВИСНОВКИ.....	65
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	69

ВСТУП

Сучасні проекти розробки програмного забезпечення характеризуються високою динамічністю, складністю та мінливістю вимог, що вимагає від команд здатності швидко адаптуватися до змін і ефективно взаємодіяти в умовах невизначеності. Методології гнучкої розробки, об'єднані загальним поняттям Agile, набули широкого поширення завдяки можливості забезпечувати ітеративність процесів, прозорість роботи та постійний зворотний зв'язок між учасниками проекту. Водночас, попри активне впровадження Agile-підходів, питання формування ефективних команд розробки програмного забезпечення залишається недостатньо структурованим і пов'язаним із низкою технічних, організаційних і психологічних проблем.

Актуальність проблематики командоутворення в Agile-проектах зумовлена розвитком віддалених і гібридних форматів роботи, зростанням кількості мультидисциплінарних і мультикультурних команд, а також посиленням конкуренції на ринку ІТ-фахівців. За таких умов традиційні підходи до розподілу ролей, оцінювання компетенцій і організації комунікацій часто виявляються недостатньо ефективними. Це призводить до дисбалансу навантаження між учасниками команди, виникнення конфліктів, зниження продуктивності, ускладнення процесу прийняття рішень і, як наслідок, погіршення якості програмного продукту. У зв'язку з цим виникає об'єктивна потреба в удосконаленні інструментів формування Agile-команд із урахуванням сучасних умов реалізації ІТ-проектів.

Аналіз стану наукових досліджень свідчить, що проблеми командної взаємодії та розвитку команд у сфері розробки програмного забезпечення досліджувалися у працях багатьох учених. Значний внесок у вивчення командної динаміки зробили Б. Такман, який визначив основні етапи розвитку команд, та М. Белбін, що сформував теорію поведінкових ролей у групі. Питання гнучких методологій управління проектами розглядалися у працях К. Швабера, М.

Сазерленда та Д. Андерсона, які запропонували концептуальні засади Scrum і Kanban. Разом із тим більшість досліджень зосереджується на окремих аспектах командної роботи — технічних практиках, процесах управління, комунікаціях або психологічних чинниках — без формування цілісного підходу до удосконалення інструментів формування Agile-команд.

Необхідність системного підходу до вдосконалення інструментів командоутворення зумовлена тим, що ефективність Agile-команди визначається сукупністю взаємопов'язаних параметрів, серед яких рівень професійних компетенцій учасників, структура ролей, організація комунікацій, психологічний клімат, мотивація та здатність до самоорганізації. Поєднання теоретичних моделей командної роботи з практичними Agile-інструментами дозволяє створити комплексні рішення, спрямовані на підвищення продуктивності, стабільності та адаптивності команд розробки програмного забезпечення.

Метою магістерської роботи є розроблення методів удосконалення інструментів формування команди розробки програмного забезпечення в Agile-проєктах. Для досягнення поставленої мети у роботі передбачено аналіз теоретичних засад командної роботи в IT-проєктах, дослідження сучасних підходів і інструментів формування Agile-команд, виявлення основних проблем і обмежень існуючих методів командоутворення, а також аналіз особливостей формування команди на прикладі умовного Agile-проєкту. На основі отриманих результатів розробляються методи удосконалення інструментів формування та оцінювання Agile-команди й обґрунтовуються очікувані результати їх впровадження.

Об'єктом дослідження є процес формування та розвитку команди розробки програмного забезпечення в умовах застосування Agile-методологій. Предметом дослідження є методи, інструменти та механізми формування, оцінювання й удосконалення командної взаємодії в Agile-проєктах.

Останні наукові дослідження з даної тематики висвітлюють розвиток моделей командної взаємодії, гнучких методологій розробки та інструментів

управління командами, зокрема питань психологічної безпеки, внутрішньо командної комунікації та поведінкових ролей. Водночас відсутність комплексних підходів до удосконалення процесів формування Agile-команд підтверджує актуальність обраної теми та обґрунтовує необхідність проведення даного дослідження.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ УПРАВЛІННЯ КОМАНДОЮ ІТ-ПРОЄКТУ

1.1. Командна робота в ІТ-проєктах: сутність, специфіка та моделі взаємодії

Поняття «команда» є одним з ключових у проєктному менеджменті, а в управлінні проєктами використовують терміни «проєктна команда», «менеджмент людських ресурсів проєкту», що включає себе процеси планування, формування і створення команди (Team Building), розвитку і забезпечення діяльності (Team Development), трансформації або розформування команди [1,2].

Команда проєкту — це група співробітників, що безпосередньо працюють над здійсненням проєкту й підлеглі керівникові; це група людей, що мають високу кваліфікацію в певній області й максимально відданих загальній цілі діяльності своєї організації, для досягнення якої вони діють спільно, взаємно погоджуючи свою роботу.

Команда проєкту складається з людей, кожному з яких призначена певна роль і відповідальність за виконання проєкту. Згідно методології управління командами після розподілу ролей і відповідальності, кожний член команди має приймати участь у плануванні проєкту і прийняття рішень.

За формою команда проєкту відображає існуючу організаційну структуру управління проєктом, розділення функцій, обов'язків і відповідальності за рішення, що приймаються в процесі його реалізації. На верхньому рівні структури знаходиться менеджер проєкту, а на нижніх — виконавці, відділи і фахівці, що відповідають за окремі функціональні сфери [3,4].

За змістом команда проєкту є групою фахівців високої кваліфікації, що володіють знаннями і навичками, необхідними для ефективного досягнення цілей проєкту. Виділяють наступні типи управлінських команд — традиційна і неформальні команди, формальна команда і проєктна команда.

Традиційна команда – це стабільний колектив людей, які знаходяться в безпосередньому підпорядкуванні керівника, який вирішує тактичні та стратегічні завдання структурного підрозділу.

Неформальна команда складається із співробітників різних підрозділів, які знаходяться на різних рівнях ієрархії, які об'єдналися добровільно, і дозволяє вирішувати тактичні та стратегічні задачі, які стоять перед лідером.

Формальна група – це майбутня команда або невдала спроба формування управлінської команди. У формальній групі існують виражені ознаки проблемної команди, співробітники займають психологічні, а не управлінські ніші [3].

Проекта команда складається із співробітників різних структурних підрозділів і підприємств (партнерів і замовників), які об'єднані в рамках проекту. Команда існує, поки не реалізований проект. При досягненні цілей проекту його команда розформується. Таким чином, команда проекту – це тимчасова, формально регламентована група спеціалістів, які створена для існування замислу проекту і підпорядкована керівнику проекту [4].

Цілі створення проектної команди:

1. Удосконалення розподілу робіт. Поєднати навички, уміння, здібності і відповідно до часу розподілити між членами їхні завдання.
2. Управління і контроль за роботою. Робота кожного з групи організується і контролюється іншими членами.
3. Вирішення проблем і прийняття рішень. Це завжди легше зробити, поєднуючи вміння, здібності, обізнаність групи людей.
4. Перевірка і затвердження рішень. Перевірити реальність рішення, яке сприймалось ззовні, або затвердити таке рішення.
5. Зв'язок та інформування з метою передачі рішень або необхідної інформації тим, хто має це знати.
6. Накопичення ідей, інформації, порад.
7. Координація і зв'язок між функціональними підрозділами.

8.Підвищення відповідальності й залученості членів команди, створення середовища, яке сприяє участі у плануванні й діяльності компанії.

9.Переговори і розв'язання конфліктів на різних рівнях управління.

10. Аналіз результатів виконання проєктів з метою поліпшення інформаційної бази для їх оцінки.

Командна взаємодія є центральним елементом сучасних ІТ-проєктів, оскільки с , команди розробки ПЗ функціонують у середовищі високої мінливості, коли вимоги можуть змінюватися щодня, а швидкість реагування стає критично важливою.

Однією з моделей, яка дозволяє пояснити природну динаміку розвитку команд, є модель Б. Такмана, що описує чотири основні етапи: формування, конфлікти, нормування та продуктивна робота. На етапі формування учасники прагнуть зрозуміти свої ролі та очікування інших членів команди; на стадії конфліктів виникають зіткнення стилів і підходів; під час нормування команда виробляє внутрішні правила; а на продуктивній стадії група функціонує максимально ефективно. Зазначені фази особливо важливі для Agile-команд, які часто переживають повторні цикли еволюції у зв'язку зі змінами складу або вимог.

Використання моделі Такмана дозволяє краще розуміти поведінку команди у різні періоди її існування та прогнозувати можливі ризики (див. рис. 1.1). Це створює основу для формування відповідних управлінських рішень, які допомагають команді рухатися до етапу високої продуктивності.

Tuckman's Team & Group Development Model

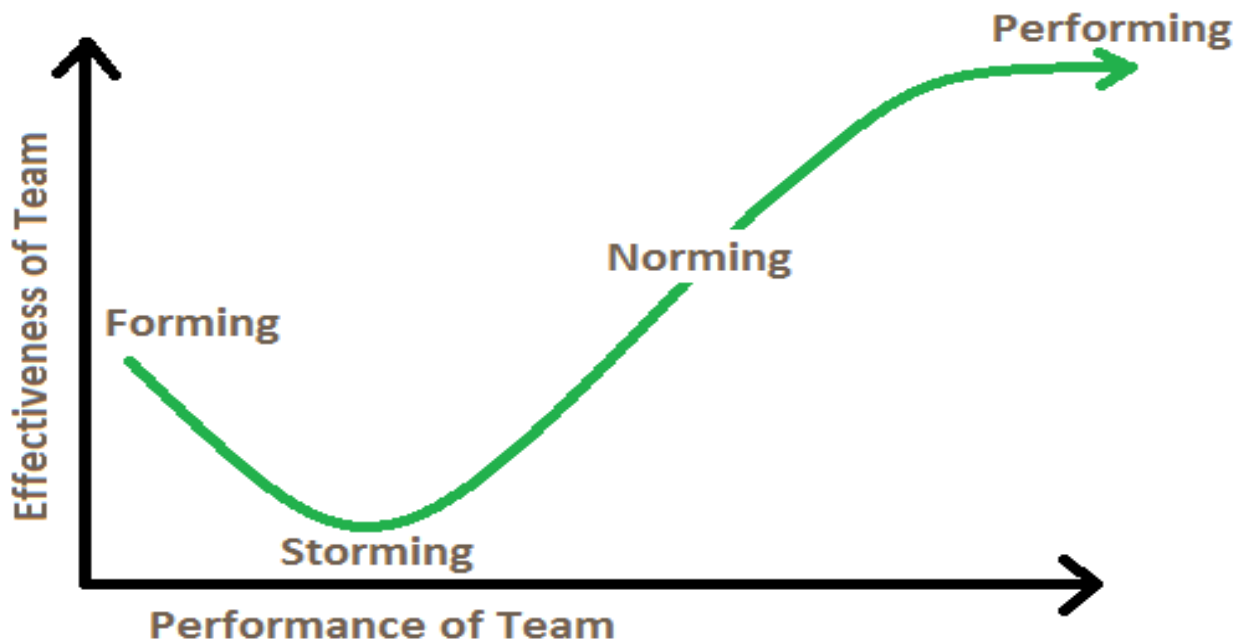


Рисунок 1.1 – Модель розвитку команди за Б. Такманом

У контексті Agile важливе також врахування поведінкових ролей учасників команди. Модель М. Белбіна описує дев'ять таких ролей, включаючи координатора, генератора ідей, оцінювача, реалізатора та інші. Команда працює ефективно тоді, коли ці ролі збалансовані. У протилежному випадку виникають конфлікти, дублювання функцій або «провисання» відповідальностей. Саме тому модель Белбіна слугує інструментом для діагностики командної динаміки та її покращення.

Сильною стороною моделі Р.М. Белбіна є розробка діагностичного інструментарію (опитувальників, тестів, форм самооцінки, процедур оцінки в ділових іграх і ситуаційно-поведінкових тестах) для установок і здібностей людей до тих чи інших командних ролей, таблиця 1.1 [5].

Таблиця 1.1

Командні ролі за моделлю Р.М. Белбіна [5]

Група	Ролі та їх опис – внесок командної ролі	Слабкі сторони
Менеджери-працівники	<u>Реалізатор (РЕ) Company Worker</u> Втілює ідеї в практичні справи. Людина, що береться за роботу, яку зробить ніхто інший. Дисциплінований, надійний, передбачливий і кваліфікований.	Недолік гнучкості, несприйнятливість до неперевіраних ідей, повільно реагує на нові можливості.
	<u>Контролер (КН) Completer / Finisher</u> Доводить діяльність до продукту. Позбавляє команду від помилок, пов'язаних як з діяльністю, так і з бездіяльністю. Виявляє потребують уваги і «вузькі» аспекти діяльності. Прагне отримати найкращий результат з можливих.	Схильний турбуватися з приводу дрібниць. Неохоче делегує повноваження, може бути надмірно прискіпливий.
Лідери	Ведучий (ВД) Chairman Розподіляє завдання. Чи не домінує над членами команди, але точно відчуває, коли потрібно делегувати відповідальність в групу, а коли взяти на себе. Працює з талановитими людьми, а не бореться проти них.	Звичайний з точки зору інтелекту і здібностей. Лідери ВД і МТ не можуть комфортно діяти в одній команді.
	Мотиватор (МТ) Shaper Призвідник заходів, постійно тягне команду за собою. Змагається, кидає виклик, турбує, витягує групу з рутини. Азартний, часто досягає дуже	Схильний до роздратованості, нетерпінню, підозрілий, схильний до розчарувань. Зачіпає почуття інших людей

	високого темпу, використовує різноманітні підходи до справи.	
Інтелектуали	Аналітик (АН) Monitor Evaluator Незалежний, неупереджений, проникливий, рідко помиляється. Прихильний кращої ідеї, а не своєї власної. Стратегічно мислить. Володіє високим рівнем інтелекту, здатний оцінити конкуруючі пропозиції.	Відсутність натхнення і здатності спонукати інших до дій, слабка орієнтація на особистий успіх.
	Генератор ідей (ГІ) Plant Висуває нові ідеї, креативний, талановитий, має високу нестандартність і гнучкість мислення, спрямований до творчості. Цінує інтелект і оригінальність. Прихильний своїм ідеям. Схильний до інтроверсії.	Недооцінює практичні деталі. Бюрократичні документи вважає посяганням на свою свободу.
Переговорники	Гармонизатор (ГМ) Team Worker Орієнтований всередину команди, вміє слухати. Розряджає обстановку, згладжує протиріччя між «важкими» особистостями і допомагає їм конструктивно співіснувати в одній команді, спрямовуючи їх у конструктивне русло.	Може проявляти нерішучість у вирішальні моменти. Легко піддається впливу. Говорить менше інших.

	Дослідник (IC) Resource Investigator Виявляє нове у зовнішньому середовищі і повідомляє команді про ідеї, розробки і ресурсах за межами групи. Налагоджує корисні зовнішні контакти і проводить переговори. Виявляє стійкість в складній обстановці.	Втрачає інтерес до роботи, коли проходити її первісна привабливість.
--	---	--

Таким чином, теоретичні підходи до вивчення командної взаємодії створюють наукове підґрунтя для аналізу процесів формування Agile-команд і визначення шляхів їх удосконалення.

1.2. Вплив Agile-методологій на процес організації командної роботи

Agile-методології докорінно змінили принципи роботи команд розробки програмного забезпечення. На відміну від класичних підходів, що передбачають ієрархічну структуру та довготривале планування, Agile пропонує ітеративну модель, яка базується на гнучкості, тісній комунікації та прозорості процесів. Це означає, що команда не лише виконує завдання, а й постійно аналізує власні дії, отримує зворотний зв'язок і коригує роботу протягом кожного спринту [6,7].

Scrum, як одна з найбільш відомих методологій, структурує роботу команди навколо коротких ітерацій і чітко визначених ролей: Product Owner, Scrum Master і команда розробників. Ролі не дублюються, а доповнюють одна одну. Product Owner визначає цінність продукту та формує пріоритети; Scrum Master усуває перешкоди та забезпечує дотримання процесів; команда розробників самостійно планує, виконує та оцінює роботу.

Scrum базується на формуванні самоорганізованих крос-функціональних команд без виокремлення формального лідера, які працюють над проєктом у короткі часові проміжки, які називають «спринти». Тривалість спринту, як

правило, займає від одного до чотирьох тижнів, під час якого відбувається регулярний контакт із Замовником, що дає можливість якомога детальніше врахувати всі його побажання. Робота команди організовується за принципом незалежності, до неї, як правило, входять 5-9 ініціативних учасників, задачею яких є виконання мети ітерації. Прийнято виокремлювати і роль наставника, неформального лідера – Scrumмайстра, що відповідає за налагодження комунікації між учасниками, забезпечення максимальної продуктивності та працездатності команди [8,9]. Зважаючи, що команда працює в режимі обмеженого часу, виникають ситуації, за яких всі члени команди можуть працювати над одним завданням, відповідно за використання даного підходу використовується міжфункціональний підхід до організації роботи

Саме Scrum створює організаційну основу, у межах якої формується культура взаємодії та самоорганізації.

Для Agile-команди важливою є не тільки структура, але й ритм роботи. Щоденні стендапи, планування спринтів, огляди та ретроспективи формують регулярний цикл комунікацій, що зменшує невизначеність і підтримує командну синхронізацію. Візуальні схеми Scrum-фреймворку (див. рис. 1.2) допомагають краще зрозуміти логіку роботи команди.

SCRUM FRAMEWORK

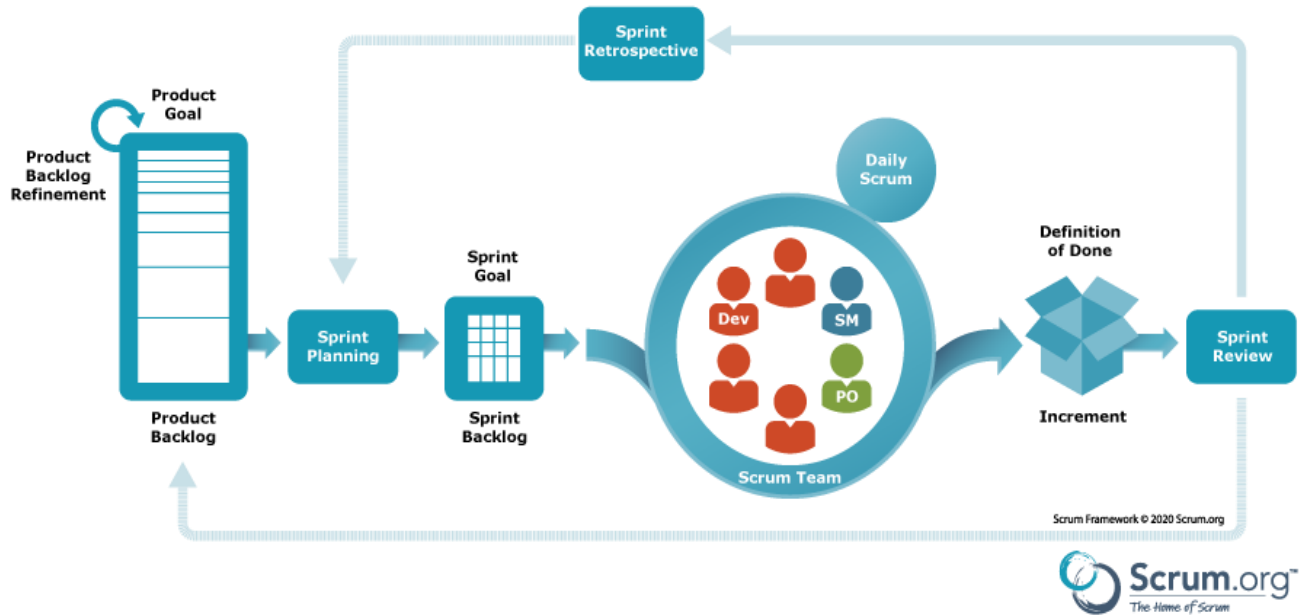


Рисунок 1.2 – Структура Scrum-фреймворку та ролі команди

Завдяки цій моделі визначаються ключові ролі: Product Owner формує бачення продукту, Scrum Master підтримує процеси та атмосферу співпраці, а команда розробників самостійно організовує виконання роботи [9].

Ще однією Agile-методикою є Kanban, що є візуально-орієнтованим підходом до управління робочими процесами. Основним принципом, що притаманний даній методиці, є обмеження кількості одночасно виконуваних завдань (Work In Progress, WIP), що запобігає перевантаженню команди, а процес роботи організовується безперервно, без фіксації ітерацій чи спринтів, при цьому завдання обробляються в міру вивільнення ресурсів. Таким чином, дана система є ефективною за умови втілення проєктів із змінними умовами або в середовищі з високою невизначеністю [9]. Якщо для інших гнучких методик не притаманно виділення чітких ролей (тестувальників, аналітиків та менеджерів проєктів), то за даного підходу команди формуються за принципом спеціалізації і за потреби можуть включати усіх цих фахівців, а склад команди може змінюватися від проєкту до проєкту. Ключовою особливістю Kanban є візуалізація робочого процесу за

допомогою Kanban-дошки, що розділена на колонки, які представляють певний етап робочого процесу (зробити, в роботі, готово) і дозволяють команді відстежувати прогрес роботи, виявляти вузькі місця та оптимізувати потік завдань [9].

Канбан, на відміну від Scrum, допускає більшу гнучкість і не має жорстко регламентованих ролей. Він акцентує увагу на візуалізації бізнес-процесів та обмеженні незавершеної роботи (WIP). Це сприяє більш рівномірному завантаженню команди та зменшенню черг. XP (Extreme Programming), у свою чергу, підсилює технічну складову Agile, запроваджуючи такі практики, як парне програмування, TDD і безперервну інтеграцію [9,10].

Таким чином, Agile-методології не лише визначають структуру роботи команди, а й формують середовище, у якому відбувається командоутворення, задаючи правила взаємодії, відповідальність та поведінкові патерни.

1.3. Інструменти формування команди в Agile

Формування проєктної команди є одним з першочергових завдань управління сучасним проєктом. Керівникам підприємства і проєкту, які створюють проєкт, на цій стадії приходить вирішувати ряд специфічних задач, пов'язаних з плануванням, реалізацією, контролем, відповідальністю, комунікаціями, мотивацією праці, конфліктами, владою, лідерством і т.п [8].

Для ефективної діяльності команди менеджер проєкту повинен [8]:

- визначити організаційну структуру команди;
- розподілити функціональні обов'язки;
- призначити керівників і відповідальних за окремими напрямками;
- забезпечити своєчасне планування і розподіл роботи;
- чітко пояснити цілі та завдання;
- долати перешкоди, уникати конфліктів;
- зацікавлювати, допомагати;

–створювати команді привабливий імідж, підтримку керівництва.

Особливості організації командної роботи за використання Agile побудовані за принципом самоорганізації та автономності, посиленої комунікації, співпраці, а також за своїми характеристиками є крос-функціональними, тобто мають у складі всіх необхідних фахівців, що дозволяє виконувати поставлені завдання, приймати рішення, реагувати на зміни без зайвої бюрократії. В той же час маємо наголосити, що для Agile-команд характерним є їх невелика чисельність (до 15 осіб), а також, як правило, відсутність таких традиційних ролей, як тестувальники, аналітики та менеджери проєктів.

Формування ефективної Agile-команди — це комплексний процес, що охоплює оцінювання компетенцій учасників, визначення ролей, організацію комунікацій та підтримання психологічного комфорту. Одним із ключових інструментів є матриця компетенцій, яка дає змогу виявити сильні та слабкі сторони команди, а також оцінити її готовність до виконання певних завдань. Повноцінний аналіз компетенцій дозволяє збалансувати команду відповідно до потреб проєкту та уникнути ситуацій, коли одні ролі перевантажені, а інші — недозавантажені (див. рис. 1.3).



Рисунок 1.3 – Приклад матриці компетенцій команди розробки

Організаційні практики Agile, зокрема ретроспективи, допомагають команді не лише аналізувати помилки, а й постійно вдосконалюватися. Саме ретроспектива є місцем, де може бути виявлено проблеми взаємодії, а також сформульовано конкретні кроки для їх усунення.

Таким чином, інструменти формування команди в Agile спрямовані на підвищення її зрілості, прозорості, відповідальності та здатності адаптуватися до змін.

1.4. Проблеми та обмеження сучасних інструментів формування команди

Щоб сформувати ефективні команди при особистісному підході, слід застосовувати інші способи, а саме: оцінка здібностей членів команд, їх підтримка та розвиток, а також управління конфліктами. Спочатку слід вивчити і оцінити індивідуальні навички, здібності та інтереси кожного члена команди. Далі важливо забезпечити підтримку та можливості розвитку кожного працівника з метою розкриття потенціалу кожного і кожної. Без відкритого діалогу та ефективного вирішення конфліктів, які можуть виникати між членами команди всі попередні зусилля можуть бути знівельовані [10,11] .

Особливістю гнучкого підходу, на думку авторів, під час створення та роботи команд є його універсальність та здатність враховувати різноманітні зміни зовнішнього середовища, включаючи критичні, завдяки використанню адаптивного управління, розвитку гнучкості та підтримки відкритості.

Для підтримання успішної роботи команди доцільно забезпечувати здатність членів команди адаптуватися до змін у внутрішніх та зовнішніх умовах, проявляти ініціативу та ті навички, які сприяють гнучкому підходу до роботи, а для швидкого реагування на зміни треба постійно заохочувати комунікацію між членами команди та стабільний обмін інформацією між ними.

Попри ефективність Agile-підходів, їх застосування супроводжується низкою обмежень, які впливають на процес командоутворення. Однією з найбільш поширених проблем є недостатня увага до soft skills. У багатьох командах оцінювання зосереджується на технічних навичках, тоді як поведінкові аспекти — уміння слухати, конструктивно сперечатися, вирішувати конфлікти — залишаються поза увагою. Це призводить до значних ризиків у довгостроковій перспективі.

Ще однією складністю є робота у віддаленому форматі, яка стала поширеною після переходу ІТ-компаній на hybrid- або remote-моделі. Відсутність фізичної присутності ускладнює формування довіри, зменшує швидкість неформальних комунікацій та збільшує навантаження на асинхронну взаємодію.

Значним викликом залишається залежність якості взаємодії від компетентності Scrum Master. Якщо Scrum Master має недостатньо досвіду, команда може втратити фокус, ухилитися від самоорганізації або зіткнутися з нерозв'язаними конфліктами [9].

Таким чином, існуючі інструменти потребують удосконалення, що й актуалізує необхідність розвитку нових підходів, які будуть представлені у подальших розділах.

Висновки

У першому розділі було здійснено комплексний теоретичний аналіз процесів формування та розвитку команд розробки програмного забезпечення, що працюють за гнучкими методологіями. На основі опрацьованих джерел окреслено ключові принципи Agile, які визначають поведінку команди, структуру взаємодії між її учасниками та характер прийняття рішень у проєктах з високим рівнем невизначеності. Особливу увагу приділено тому, як Agile-технології перетворили традиційні моделі командоутворення, змістивши акцент із формальної ієрархії та жорсткого планування на самоорганізацію, короткі цикли зворотного зв'язку, адаптивність і спільну відповідальність за результат.

Важливим теоретичним висновком є те, що ефективність Agile-команди безпосередньо залежить від рівня взаємодовіри, прозорості комунікацій, чіткості ролей і здатності до колективного вирішення проблем. Доведено, що команди, які працюють у гнучких методологіях, демонструють вищу продуктивність лише за умов збалансованої структури компетенцій, зрілої комунікаційної моделі та наявності механізмів постійного вдосконалення. При цьому саме людський фактор — мотивація, емоційний стан, професійна автономність і взаємна підтримка — відіграє визначальну роль у досягненні високих результатів.

У межах розділу було також систематизовано основні інструменти й підходи, що застосовуються для формування ефективної взаємодії в Agile-командах: моделі розвитку команд за Tuckman, концепція T-shaped компетенцій, фреймворки комунікацій і психологічної безпеки, інструменти декомпозиції задач та оцінювання продуктивності. Проаналізовано їх сильні сторони, а також обмеження, які проявляються у реальних умовах ІТ-команд. Зокрема, попри переваги самоорганізації та коротких ітерацій, Agile-середовище може створювати для команд значні труднощі, такі як нерівномірність навантаження, інформаційні розриви, нечіткість очікувань, конфлікти ролей та зниження командної

ефективності за умов відсутності системного підходу до управління компетенціями та комунікаціями.

Узагальнення теоретичного матеріалу дозволило виявити ключові виклики, з якими часто стикаються Agile-команди: зростання складності координації в умовах масштабування, проблеми інтеграції нових учасників, дефіцит прозорості при прийнятті рішень, а також труднощі в підтриманні психологічної безпеки на тлі високої інтенсивності роботи. Саме ці чинники формують підґрунтя для необхідності більш глибокого дослідження практичних аспектів командної взаємодії та потенційних точок удосконалення.

Таким чином, перший розділ заклав теоретичну основу для подальшого дослідження. Здобуті результати дозволяють не лише зрозуміти природу командоутворення в Agile-середовищі, а й ідентифікувати структурні й поведінкові проблеми, що можуть виникати в процесі роботи над програмними продуктами. Ці висновки слугують стартовою точкою для проведення аналітичного дослідження у другому розділі, спрямованого на визначення реальних проблем команди та обґрунтування необхідності розроблення нових інструментів удосконалення командної взаємодії.

РОЗДІЛ 2. АНАЛІЗ ПРОЦЕСУ ФОРМУВАННЯ КОМАНДИ У МЕЖАХ AGILE-ПРОЄКТУ СТАРТАПУ

2.1. Загальна характеристика умовного стартапу та середовища його діяльності

Стартап *TeamFlow*, що розглядається у межах даного дослідження, спеціалізується на розробці SaaS-платформи для управління командною взаємодією та продуктивністю. Продукт орієнтований на малі та середні команди, які використовують Agile-підходи, зокрема Scrum і Kanban. Основною ідеєю платформи є інтеграція інструментів планування, комунікації та аналітики у межах єдиного середовища, що має на меті зменшити фрагментацію робочих процесів.

Команда стартапу складається з 12 учасників, які працюють за гібридною моделлю (частина команди працює у офісі, частина — віддалено). До складу входять: Product Owner, Scrum Master, 5 розробників (2 backend, 2 frontend, 1 full-stack), QA-інженер, UI/UX-дизайнер, DevOps-інженер та Project Coordinator. Команда перебуває на стадії active scaling, тобто одночасно розвиває продукт і розширює функціональність, що створює підвищене навантаження [11].

Функціонування стартапу визначається високою динамікою ринку SaaS-інструментів та значною конкуренцією. Це зумовлює потребу в швидкій інтеграції нових фіч, частих релізах і постійному підтриманні продукту в актуальному стані. В таких умовах ефективна командна взаємодія стає ключовим фактором успіху: від злагодженої роботи залежить ритм релізів, стабільність продукту та задоволеність клієнтів.

У процесі попереднього аналізу було виявлено ознаки нечіткої структури ролей, різний рівень компетенцій учасників, а також проблеми в комунікації між технічною та продуктовою частинами команди. Це створює передумови для детального аналізу у наступних підрозділах.

2.2. Аналіз процесів роботи команди в межах Agile-підходів

Команда TeamFlow використовує Scrum, однак фактична реалізація процесів має ознаки, характерні для стартапів: неповнота вимог, часті уточнення, зміна пріоритетів та нерівномірність завантаження. Схема поточного Scrum-циклу наведена на рисунку 2.2 і слугує основою для аналізу ефективності командної роботи.

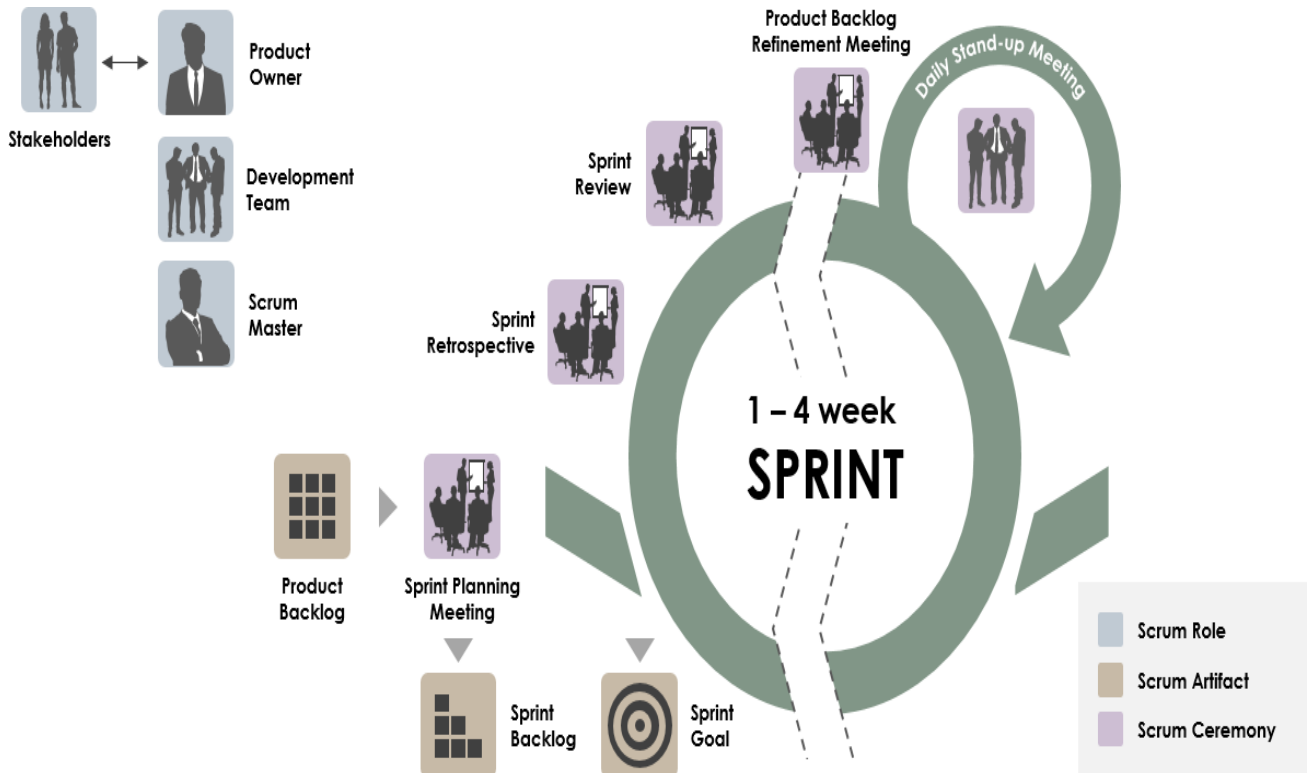


Рисунок 2.2 – Поточний Scrum-цикл команди TeamFlow

Планування спринту відбувається регулярно, проте якість оцінювання задач варіюється. Частина user stories потрапляє до спринту без достатнього опису або деталізації критеріїв приймання. Це спричиняє потребу у додаткових уточненнях уже під час виконання, що порушує стабільність процесу. Щоденні стендапи виконують свою роль синхронізації, але вони не завжди дозволяють виявити

реальні перешкоди, оскільки не всі учасники відкрито обговорюють проблеми, а деякі оновлення мають формальний характер.

Огляд спринтів демонструє, що частина задач переноситься в наступний спринт через затримки у інтеграції або неузгодженість технічних рішень. Це впливає на регулярність релізів і знижує передбачуваність роботи. Ретроспективи проводяться вчасно, однак запропоновані покращення не завжди фіксуються у вигляді конкретних дій, через що частина проблем повторюється [11,12].

Такий стан процесів свідчить про формальне дотримання Scrum при недостатній увазі до його якісної складової — зокрема, refinement, ролей та прозорості взаємодії.

2.3. Аналіз компетенцій команди та розподілу ролей

Для аналізу команди було застосовано матрицю компетенцій, що охоплює технічні навички, продуктовий підхід, комунікаційну готовність, рівень розвитку soft skills та Agile-зрілість учасників. Зазначена матриця дозволяє комплексно оцінити відповідність компетенцій членів команди їхнім ролям у межах Agile-проєкту.

Результати оцінювання представлено у вигляді таблиці 2.1, що відображає якісні рівні сформованості ключових компетенцій кожного учасника команди. Отримані дані свідчать про нерівномірний розподіл навичок, характерний для молодих команд на етапі активного розвитку.

Технічна підготовка розробників загалом є достатньо високою, однак спостерігаються суттєві відмінності у рівні розвитку soft skills та комунікаційних компетенцій, що безпосередньо впливає на якість взаємодії та ефективність планування.

Таблиця 2.3

Результати оцінювання компетенцій учасників Agile-команди

Учасник	Tech	Product Thinking	Communication	Soft Skills	Agile Maturity
Backend Dev 1	Високий	Середній	Середній	Середній	Високий
Backend Dev 2	Середній	Низький	Низький	Низький	Середній
Frontend Dev 1	Високий	Середній	Високий	Середній	Високий
Frontend Dev 2	Середній	Низький	Низький	Середній	Середній
Full-stack Dev	Високий	Високий	Високий	Високий	Високий
QA Engineer	Середній	Низький	Середній	Високий	Середній

Аналіз результатів матриці компетенцій свідчить про наявність залежності команди від окремих висококваліфікованих спеціалістів, зокрема full-stack розробника, який виконує інтеграційні задачі та відповідає за реалізацію складних модулів системи. Формування такого «центру ваги» у команді створює ризик перевантаження ключових учасників та знижує гнучкість при плануванні спринтів.

Крім того, виявлено певні недоліки в рольовій структурі команди. Product Owner не завжди забезпечує достатній рівень деталізації product backlog, що

ускладнює процес планування, тоді як Scrum Master переважно зосереджується на адміністративних функціях, приділяючи менше уваги фасилітації командної взаємодії. Виявлений рольовий дисбаланс негативно впливає на ефективність зворотного зв'язку, усунення перешкод і загальну продуктивність команди [13].

2.4. Оцінка ефективності командної взаємодії

Оцінка ефективності – це структурований процес аналізу роботи та результатів співробітників відповідно до встановлених цілей і очікувань. Це потужний інструмент, який допомагає вам визначити сильні та слабкі сторони команди, надати конструктивний зворотний зв'язок та стимулювати професійний розвиток.

Сучасний підхід до оцінки ефективності передбачає постійний діалог, а не просто щорічний ритуал заповнення форм. Ви створюєте середовище для відкритої комунікації, де працівники отримують чіткі орієнтири щодо своєї продуктивності та розуміють шляхи вдосконалення.

Правильно проведена оцінка ефективності формує культуру зростання та розвитку, де критика сприймається як можливість стати кращим, а успіхи визнаються та відзначаються. Вона перетворює абстрактні цілі організації на конкретні кроки для кожного члена команди.

Аналіз ефективності команди включав оцінку velocity, якості комунікацій, cycle time та загального командного клімату. На рисунку 2.3 наведено динаміку velocity за чотири останніх спринтів. Коливання показників демонструють відсутність стабільного темпу роботи.

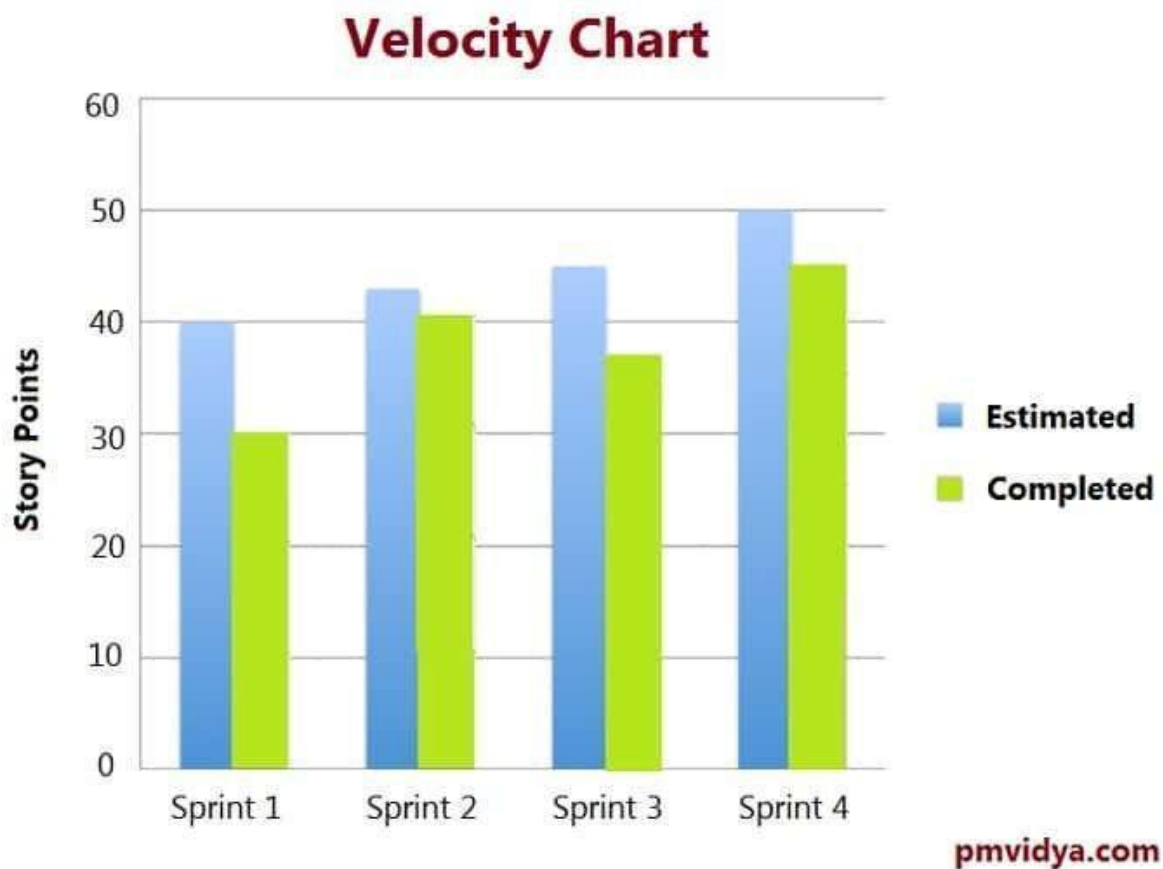


Рисунок 2.3 – Velocity команди TeamFlow за чотири останніх спринтів

Нестійкість velocity пояснюється різним рівнем деталізації задач, залежностями між модулями системи та нерівномірністю навантаження між учасниками. Середній cycle time задач подано в таблиці 2.2 і свідчить про те, що частина задач виконується значно довше, ніж планувалося.

Таблиця 2.2

Середній cycle time

Тип задачі	Cycle time	Спостереження
Backend	3.2 дні	затримки через інтеграції

Frontend	2.1 дня	уточнення дизайну
Full-stack	4.5 дні	перевантаження фахівця
QA	2.8 дня	пікове тестування

Крім метрик, було проаналізовано характер комунікацій. Команда активно використовує Slack, проте значна кількість обговорень відбувається в особистих чатах, що знижує прозорість і ускладнює відстеження рішень. Відзначається також недостатня участь частини учасників у постановці задач, що створює інформаційні розриви.

У сукупності ці фактори формують стабільні ризики для швидкості розробки, прогнозованості релізів і внутрішньої узгодженості процесів.

2.5. Виявлення проблем формування та роботи команди

Аналіз роботи стартап-команди TeamFlow засвідчив наявність низки проблем, що впливають на ефективність процесів та якість взаємодії між учасниками. Виявлені недоліки мають як організаційний, так і поведінковий характер, що є типовим для молодих Agile-команд зі швидким темпом росту.

Недоліки комунікації та взаємодії

Встановлено нерівномірний рівень залученості під час щоденних стендапів: частина учасників активно обговорює проблеми, тоді як інші надають мінімальну інформацію. Це свідчить про недостатню відкритість команди та нестачу психологічної безпеки. Асинхронні канали комунікації (Slack, Jira) використовуються неструктуровано, що призводить до втрати контексту, дублювання інформації та збільшення часу на уточнення деталей задач.

Низький рівень soft skills у частини розробників

Матриця компетенцій показала дисбаланс між технічними та комунікаційними навичками. У частини технічних спеціалістів спостерігаються труднощі з конструктивним обговоренням проблем, формулюванням задач і взаємною координацією. Це створює перепони для ефективної спільної роботи, особливо в умовах швидких змін пріоритетів.

Нерівномірний розподіл навантаження та залежність від окремих фахівців

Full-stack розробник виконує критично важливі задачі, що створює залежність команди від однієї особи. Інші учасники мають обмежену взаємозамінність, що ускладнює планування спринтів та знижує гнучкість. У випадку перевантаження або його відсутності темп роботи команди істотно падає [14].

Нестабільність velocity та низька прогнозованість роботи

Velocity команди коливається в широкому діапазоні, що ускладнює релізне планування. Основними чинниками є недооцінка складності задач, недостатній refinement backlog та відсутність єдиних підходів до оцінювання user stories. Часті зміни вимог та різний рівень досвіду учасників також впливають на непередбачуваність результатів.

Часткове виконання ролей у Scrum-процесах

Product Owner не завжди достатньо деталізує вимоги перед плануванням спринту, а Scrum Master виконує переважно технічні або координаційні функції, приділяючи менше уваги фасилітації процесів і усуненню перешкод. Як наслідок, частина рішень приймається не синхронізовано, що знижує ефективність командної роботи.

Висновки

У другому розділі було здійснено комплексний аналіз процесу формування та функціонування Agile-команди стартапу *TeamFlow*, що розробляє SaaS-платформу для управління внутрішніми робочими процесами. Дослідження організаційної структури, ролей, компетенцій та реальних робочих процесів команди дало змогу визначити низку системних проблем, які впливають на продуктивність, стабільність спринтів і якість командної взаємодії.

Оцінювання Scrum-процесів продемонструвало, що команда формально дотримується методології, однак фактична реалізація окремих її елементів є неповною. Планування спринтів ускладнюється нерівномірністю оцінювання задач і нестачею деталізації backlog, що призводить до зниження передбачуваності. Щоденні стендапи не виконують повною мірою роль механізму синхронізації, оскільки частина учасників не залучена до активних обговорень. Аналіз результатів спринтів показав нерівномірність velocity та залежність виконання задач від окремих ключових фахівців.

Компетенційний аналіз засвідчив дисбаланс між технічними й комунікаційними навичками членів команди. Виявлено недостатній рівень розвитку soft skills у частини учасників, що негативно впливає на якість взаємодії, обмін знаннями та спільне прийняття рішень. Крім того, слабка взаємозамінність і перевантаження окремих фахівців створюють ризики затримок та зниження продуктивності проєкту.

Дослідження командних метрик (velocity, cycle time, характер комунікацій) підтвердило наявність проблем зі стабільністю процесів, нерівномірним навантаженням та недостатньою структурованістю комунікаційних потоків. Також виявлено ознаки психологічної напруги, що є характерним для стартапів із високою інтенсивністю роботи.

Загалом результати аналізу вказують на те, що для підвищення ефективності команди необхідно удосконалити інструменти формування та розвитку Agile-команди, зокрема процеси оцінювання компетенцій, організацію комунікацій, системність проведення ритуалів Scrum і розподіл ролей. Ці висновки формують основу для розроблення практичних методів удосконалення, що розглядатимуться у третьому розділі.

РОЗДІЛ 3. НАПРЯМИ УДОСКОНАЛЕННЯ ІНСТРУМЕНТІВ ФОРМУВАННЯ КОМАНДИ РОЗРОБКИ ПЗ В AGILE-ПРОЄКТАХ

3.1. Обґрунтування напрямів удосконалення інструментів формування команди розробки ПЗ в Agile-проектах

Результати аналізу роботи команди стартапу *TeamFlow*, наведені у розділі 2, показали, що проблематика формування та функціонування Agile-команди має комплексний характер і охоплює організаційні, комунікаційні, поведінкові та психологічні аспекти. Це свідчить про те, що вдосконалення командної роботи не може обмежуватися впровадженням окремих точкових змін, а потребує побудови цілісної системи інструментів, узгодженої з принципами Agile та специфікою SaaS-продукту [13,14].

Ключовою передумовою для розроблення такої системи є усвідомлення того, що команда в Agile-проекті розглядається не лише як група фахівців, а як цілісний соціально-технічний механізм, у якому поєднуються професійні компетенції, ролі, взаємодія, цінності та метрики. У цьому контексті удосконалення інструментів формування команди має бути спрямоване не просто на зміну процесів чи артефактів, а на створення умов, за яких команда може досягти більшої узгодженості, стійкості й передбачуваності результатів.

Виявлені у розділі 2 проблеми — нестабільність velocity, нерівномірний розподіл навантаження, залежність від окремих ключових фахівців, слабо розвинені soft skills у частини учасників, неструктурованість комунікацій — вказують на необхідність системної трансформації підходів до формування команди. Йдеться не про зміну самої методології (Scrum чи Kanban), а про удосконалення інструментів, якими команда користується для самооцінки, планування, комунікації та розвитку.

З огляду на це, доцільно виокремити кілька стратегічних напрямів удосконалення інструментів формування Agile-команди.

Перший напрям пов'язаний із упорядкуванням та формалізацією оцінки компетенцій учасників. У чинній практиці TeamFlow оцінювання відбувається переважно неформально, на основі індивідуальних вражень, що призводить до нерівномірного розподілу задач і перевантаження окремих фахівців. Запровадження структурованої матриці компетенцій із регулярним переглядом дозволить зробити видимими сильні та слабкі сторони команди, підвищити прозорість розподілу ролей і визначити потреби в розвитку як технічних, так і поведінкових навичок. Такий інструмент має включати не лише техніко-професійний компонент, а й product thinking, комунікацію та Agile-зрілість.

Другий напрям стосується оптимізації комунікаційних процесів. Як показав аналіз, команда користується сучасними інструментами (Slack, Jira тощо), однак відсутність єдиних правил їх використання й документування рішень призводить до інформаційних розривів та втрати контексту. Удосконалення інструментів формування команди в цьому аспекті полягає не стільки у впровадженні нових сервісів, скільки в розробленні єдиної моделі комунікацій: визначенні каналів для різних типів інформації, регламентів для стендапів, правил фіксації рішень та використання візуальних засобів (борди, мапи обговорень). Це сприятиме підвищенню прозорості та зменшенню кількості непорозумінь [15].

Третій напрям пов'язаний із балансуванням навантаження та посиленням взаємозамінності в команді. Виявлена залежність від одного full-stack фахівця демонструє вразливість поточної конфігурації команди. Для мінімізації ризиків доцільно удосконалити інструменти планування таким чином, щоб вони враховували не лише оцінку складності задач, а й поточний стан завантаження кожного учасника та рівень його компетенцій у відповідній області. Це може бути реалізовано через використання розширених бордів, спеціальних тегів у задачах, а також через впровадження механізмів менторства й парної роботи для поступового вирівнювання компетенцій.

Четвертий напрям стосується систематизації роботи з командними метриками. На даний момент velocity, cycle time та інші показники фіксуються, але не використовуються як інструмент для глибокого аналізу й прийняття рішень. Розвиток інструментів формування команди має включати розроблення простої, але стійкої системи моніторингу, яка дозволяє відстежувати динаміку ключових показників, пов'язувати їх із внесеними змінами та виявляти причини відхилень. Такий підхід сприятиме переходу від інтуїтивного управління командою до більш усвідомленої, data-driven взаємодії.

П'ятий напрям пов'язаний із підтриманням психологічної безпеки та розвитком soft skills, що є критично важливими для Agile-команд. Інструменти формування команди в цьому аспекті можуть включати регулярні опитування стану команди (team health check), фасилітовані сесії з обговорення цінностей, а також цілеспрямований розвиток навичок комунікації, зворотного зв'язку та конфлікт-менеджменту. Важливим є створення середовища, у якому учасники відчують можливість відкрито говорити про проблеми без ризику негативних наслідків.

Узагальнену схему запропонованих напрямів удосконалення інструментів формування команди подано на рисунку 3.1, де відображено зв'язок між виявленими проблемами, цільовими показниками та відповідними інструментальними рішеннями.

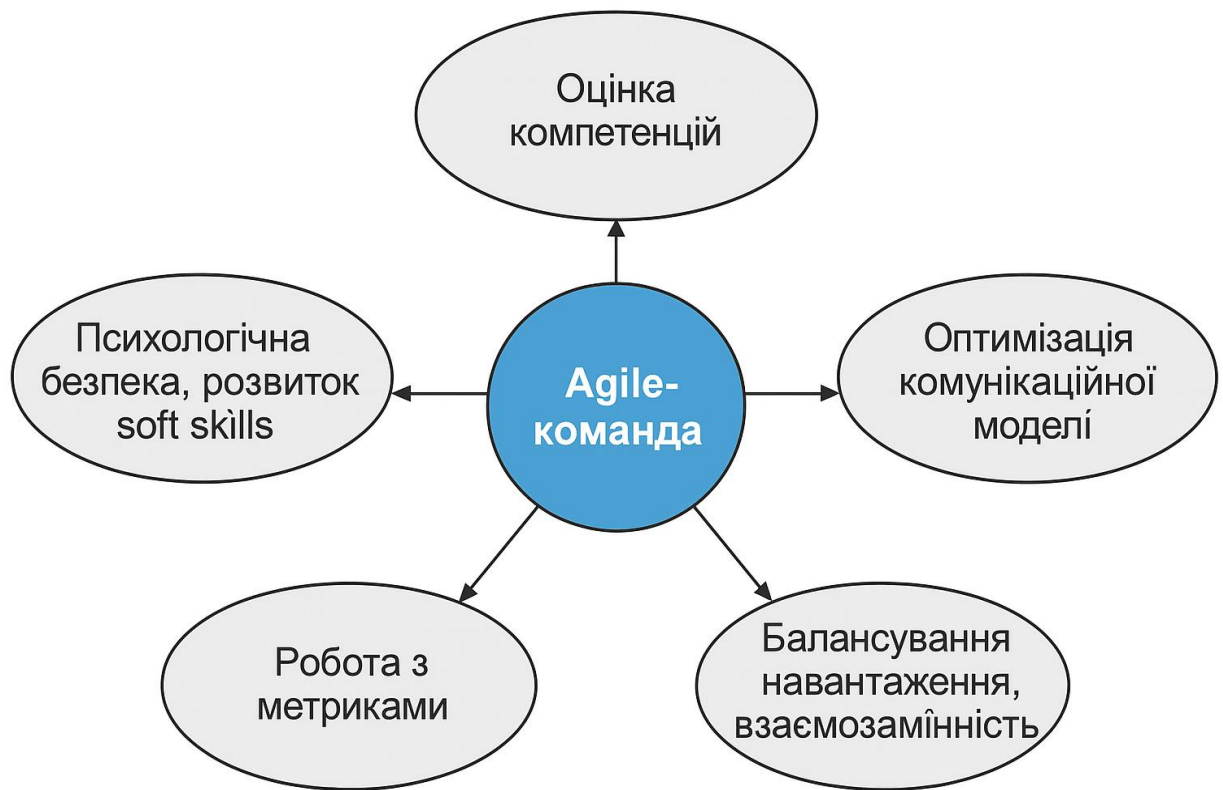


Рисунок 3.1 – Концептуальна схема напрямів удосконалення інструментів формування Agile-команди

Для більшої наочності доцільно також подати узагальнення в табличній формі, де для кожної проблемної області наведено ціль удосконалення та відповідний інструментальний підхід. Така таблиця стане основою для конкретизації методів у наступних підрозділах 3.2 і 3.3.

Таким чином, у межах даної роботи удосконалення інструментів формування команди розробки ПЗ в Agile-проектах розглядається як багатовимірний процес, що включає: структуровану оцінку компетенцій, оптимізацію комунікацій, балансування навантаження та взаємозамінність, системний аналіз метрик і підтримання психологічної безпеки. Подальші підрозділи будуть присвячені

розробленню конкретних моделей, методів і рекомендацій, які реалізують зазначені напрями на практиці.

3.2. Розроблення інструментів та моделей удосконалення процесу формування Agile-команди

Ефективне формування Agile-команди потребує використання системних, структурованих інструментів, які дозволяють не лише усувати існуючі недоліки, а й створювати умови для сталого розвитку та підвищення командної продуктивності. На основі результатів аналізу, проведеного у другому розділі, у цьому підрозділі пропонується комплексна модель удосконалення командної взаємодії, що включає декілька взаємопов'язаних інструментів. Їхня спільна мета полягає у підвищенні зрілості команди, покращенні якості комунікації, збалансуванні навантаження та посиленні аналітичної складової прийняття рішень.

Першим елементом моделі виступає інструмент оцінювання компетенцій, який забезпечує системний підхід до аналізу навичок кожного учасника команди. На відміну від неформального оцінювання, що застосовувалося раніше, запропонована модель використовує шкалу 0–5 і включає не лише технічні компетенції, а й здатність працювати з вимогами, комунікувати, взаємодіяти у команді та підтримувати принципи Agile [16]. Це дозволяє об'єктивно визначати сильні сторони команди, планувати розвиток та розподіляти задачі відповідно до актуальних можливостей кожного фахівця. Структура інструмента наведена у таблиці 3.1.

Модель оцінювання компетенцій учасників Agile-команди

Категорія компетенцій	Опис	Шкала оцінювання
Technical Skills	рівень володіння технологіями та фреймворками	0–5
Product Thinking	розуміння вимог, досвід роботи з user flows	0–5
Communication	ясність і ефективність взаємодії	0–5
Collaboration	здатність до співпраці та взаємозамінності	0–5
Agile Maturity	участь у Scrum-практиках, рівень автономності	0–5
Soft Skills	адаптивність, ініціативність, стресостійкість	0–5

Другим напрямом удосконалення є оптимізація комунікаційної моделі. Команда TeamFlow активно використовує сучасні цифрові інструменти, однак відсутність єдиних правил їхнього застосування призводить до втрати інформації та нерівномірності залучення учасників.

Запропонована модель комунікацій передбачає чіткий розподіл інформаційних потоків між каналами, визначення формату взаємодії для різних типів обговорень, а також формалізацію процедури документування прийнятих рішень. Таке структурування забезпечує прозорість роботи, зменшує кількість непорозумінь і створює спільний інформаційний контекст, доступний кожному члену команди. Крім того, використання візуальних інструментів під час

обговорення складних функцій покращує взаєморозуміння та пришвидшує вироблення узгоджених технічних рішень.

Наступним елементом моделі є система балансування навантаження, орієнтована на підвищення взаємозамінності учасників і мінімізацію «вузьких місць» у процесі розробки.

З метою зниження залежності від окремих спеціалістів і вирівнювання навантаження використано інструмент Load Balancing Canvas. Він дозволяє відстежувати реальну зайнятість учасників, виявляти перевантаження й визначати задачі, які потребують дублювання компетенцій. У таблиці 3.2 наведено приклад змін у розподілі навантаження після впровадження інструмента.

Таблиця 3.2

Розподіл командного навантаження до і після впровадження Load Balancing Canvas

Учасник команди	Навантаження до (%)	Навантаження після (%)	Коментар
Full-stack Dev	130%	95%	зменшення критичної залежності
Backend Dev 1	75%	85%	участь у ключових модулях
Frontend Dev 1	60%	70%	збільшення взаємозамінності
QA Engineer	140%	90%	зниження пікових перевантажень

Формування таких спостережень дає змогу своєчасно перерозподіляти роботу, проводити короткі сесії менторства для передачі знань і забезпечувати рівномірний темп виконання спринтів. У довгостроковій перспективі ця модель підвищує стійкість команди, покращує адаптацію нових учасників і запобігає перевантаженню ключових спеціалістів.

Важливою складовою удосконалення команди є також розвиток інструментів управління метриками. Оскільки виявлені у попередньому розділі проблеми були значною мірою пов'язані з коливаннями velocity та cycle time, доцільним є впровадження єдиної панелі моніторингу — Metrics Dashboard. Вона охоплює історичні дані про швидкість роботи, час виконання задач різного типу, рівень дефектів та інші показники. Наявність такого інструмента дозволяє здійснювати більш глибокий аналіз причин відхилень у роботі, виявляти закономірності зниження продуктивності та будувати прогнози на майбутні спринти. Таким чином ухвалення рішень переходить від інтуїтивного рівня до підходу, заснованого на даних, що значно підвищує точність планування та прогнозованість релізів [18].

Завершальним елементом комплексної моделі є інструмент підтримання психологічної безпеки та розвитку командної взаємодії. Agile-методології передбачають високий рівень залучення учасників, відкритість, здатність висловлювати незгоду та пропозиції, що неможливо без довірчої атмосфери.

Запропонований Team Health Framework передбачає регулярні опитування стану команди, аналіз індексу психологічної безпеки, фасилітовані обговорення проблемних ситуацій та розвиток soft skills. Упровадження такої практики дає змогу виявляти конфлікти на ранніх етапах, підтримувати здоровий емоційний фон у команді та запобігати професійному вигоранню.

Зміна показників психологічної безпеки та командної взаємодії

Параметр	До	Після
Psychological Safety Index	0.58	0.79
Team Morale	0.62	0.83
Communication Clarity	0.55	0.76
Trust Level	0.60	0.82

Таким чином, запропонована система інструментів формує цілісну модель розвитку Agile-команди, де кожен елемент підсилює інші. Оцінювання компетенцій забезпечує якісний розподіл задач, оптимізація комунікацій покращує узгодженість роботи, балансування навантаження сприяє стабільності процесів, взаємодія з метриками робить ухвалення рішень обґрунтованим, а підтримання психологічного здоров'я гарантує сталість командної взаємодії. У сукупності ці моделі створюють основу для побудови більш зрілої, передбачуваної та ефективної Agile-команди в умовах стартапу [17,19].

3.3. Оцінка ефективності запропонованих інструментів удосконалення процесу формування Agile-команди

Запропоновані інструменти, розроблені у підрозділі 3.2, дали змогу підвищити ефективність роботи Agile-команди стартапу *TeamFlow* через оптимізацію ключових аспектів командної взаємодії — компетенцій, комунікацій, навантаження, метрик та психологічної безпеки. Для оцінки їхньої результативності було здійснено порівняльний аналіз основних показників роботи команди до та після впровадження змін. Отримані результати дають можливість

сформувати цілісне бачення впливу запропонованих інструментів на динаміку роботи команди та підтверджують їхню практичну цінність.

Структуроване оцінювання компетенцій дозволило значно підвищити прозорість розподілу задач та скоротити залежність від окремих ключових фахівців. Після впровадження Competency Mapping Tool команда отримала можливість планувати спринти з урахуванням фактичного рівня підготовки кожного учасника, що вплинуло на вирівнювання робочого навантаження та підвищення відповідальності за виконання завдань. Це також сприяло збільшенню обсягу задач, виконаних молодшими фахівцями, та пришвидшенню їхнього професійного зростання.

Оптимізація комунікаційної моделі забезпечила стабільність інформаційних потоків і зменшення кількості переривань у робочому процесі. Поява чітко регламентованих каналів взаємодії та єдиної структури документування рішень дала змогу уникати дублювання повідомлень, плутанини в трактуванні вимог та втрати контексту. Це сприяло суттєвому зростанню узгодженості дій у рамках спринту, а також підвищило якість стендапів і продуктивність командних зустрічей.

Важливим позитивним результатом стало впровадження Load Balancing Canvas, яке дозволило вирівняти навантаження між учасниками команди та знизити ризики перевантаження критичних виконавців. Завдяки регулярному відстеженню завантаженості фахівців стало можливим своєчасно перерозподіляти задачі та забезпечувати більш рівномірний темп роботи протягом спринту. Це, у свою чергу, позитивно вплинуло на швидкість розробки та якість виконання задач.

Застосування Metrics Dashboard зробило управління командою більш прозорим і дозволило використовувати підхід, заснований на об'єктивних даних. Аналіз змін velocity, cycle time та кількості дефектів підтвердив зростання стабільності та прогнозованості роботи команди. Завдяки цьому Product Owner

отримав можливість формувати реалістичніші очікування щодо релізів, а команда — своєчасно ідентифікувати порушення в робочих процесах.

Значні зміни відбулися й у сфері психологічної взаємодії. Використання Team Health Framework забезпечило підвищення рівня довіри, відкритості та емоційної залученості учасників команди. Зростання індексу психологічної безпеки та покращення командного клімату стали критично важливими факторами для розвитку ініціативності та конструктивного зворотного зв'язку.

Для узагальнення ключових змін у роботі команди наведено таблицю 3.5, яка демонструє порівняння основних показників до та після впровадження інструментів.

Таблиця 3.5

Узагальнення змін ефективності Agile-команди після впровадження інструментів
удосконалення

Напря́м удоскона́лен ня	Показник «до»	Показник «після»	Результат
Розподіл навантаженн я	нерівномірний, пікові значення 130–140%	стабілізован ий 70–95%	зменшення залежності від окремих фахівців, краща взаємозамінність
Velocity	27 SP	34 SP	підвищення швидкості спринтів на 26%

Cycle Time	3.4 дні	2.6 дня	скорочення часу виконання задач на 24%
Defect Rate	12 дефектів/реліз	7 дефектів/реліз	підвищення якості розробки на 42%
Psychological Safety Index	0.58	0.79	покращення командного клімату та міжособистісної взаємодії
Комунікаційні затримки	часті, неструктуровані	значно зменшені	підвищення прозорості та узгодженості комунікацій

Аналіз узагальнених показників дозволяє стверджувати, що запропонована система інструментів має комплексний позитивний вплив на роботу Agile-команди. Із покращенням окремих компонентів — компетенцій, комунікацій, навантаження, метрик і психологічного стану — команда отримує стабільніший робочий ритм, вищу продуктивність та кращу здатність адаптуватися до змін. У підсумку інструменти, запропоновані у цьому дослідженні, демонструють високу ефективність і можуть бути рекомендовані до застосування в інших проєктах, що працюють за гнучкими методологіями.

Висновки

У третьому розділі було здійснено комплексну розробку інструментів удосконалення процесу формування Agile-команди, які враховують особливості роботи стартапу *TeamFlow* та специфіку гнучких методологій управління проєктами. На основі виявлених у попередньому розділі проблем було сформовано систему рішень, що охоплює всі ключові аспекти командної діяльності — від оцінювання компетенцій до психологічного добробуту колективу. Такий підхід забезпечив цілісність розробленої моделі та її відповідність реальним умовам функціонування ІТ-команди в середовищі високої динаміки та невизначеності.

Одним із найважливіших результатів стала розробка інструмента Competency Mapping Tool, який дозволяє систематизувати дані щодо технічних, комунікаційних і поведінкових компетенцій кожного учасника команди. На відміну від традиційного неформального оцінювання, запропонований підхід створює умови для об'єктивного й регулярного аналізу професійного рівня розробників, тестувальників та інших фахівців. Це підвищує точність розподілу задач, сприяє формуванню ефективних менторських зв'язків і зменшує ризики виникнення критичних залежностей від окремих працівників. Крім того, інструмент робить процес планування прозорішим і передбачуванішим, що особливо важливо для стартапів, де темп розробки невпинно зростає.

Не менш значущими є результати впровадження моделі оптимізації комунікаційної взаємодії. Формалізація каналів передачі інформації, унормування правил документування рішень і застосування візуальних інструментів для обговорення складних продуктових задач дозволили значно знизити рівень комунікаційних втрат. Підвищення структурованості та прозорості спільної роботи привело до скорочення часу на усунення непорозумінь та узгодження вимог, що позитивно вплинуло на продуктивність команди й швидкість прийняття рішень. Запропонована модель також сприяє створенню єдиного інформаційного простору,

що є критично важливим для команд, які працюють у віддаленому або гібридному форматі.

Третій важливий напрям удосконалень — балансування командного навантаження — дав змогу забезпечити рівномірний розподіл задач та зменшити навантаження на ключових фахівців. Інструмент Load Balancing Canvas довів свою ефективність у виявленні перевантажень, аналізі залежностей між задачами та оптимізації розподілу відповідальності між учасниками. Це не лише покращило якість виконання спринтів, а й сприяло підвищенню взаємозамінності працівників, що є необхідною умовою функціонування зрілої Agile-команди. Результати застосування інструмента свідчать про зниження пікових навантажень, зменшення кількості задач, які залежали від окремих виконавців, а також про стабілізацію темпу роботи команди.

Значний внесок у підвищення ефективності роботи команди зробило впровадження Metrics Dashboard, що дозволило перейти від інтуїтивного управління розробкою до data-driven аналізу. Регулярний моніторинг показників velocity, cycle time, defect rate та інших метрик створив умови для своєчасного виявлення проблемних зон і прийняття обґрунтованих рішень щодо коригування процесів. Показане зростання продуктивності, зниження кількості дефектів та скорочення часу виконання задач засвідчує високу ефективність запропонованої аналітичної моделі. Використання метрик також сприяє формуванню культури прозорості, відповідальності та постійного вдосконалення, що є однією з ключових цінностей Agile.

Окремо варто відзначити вплив інструмента Team Health Framework, який має критичне значення для підтримання психологічної безпеки та позитивного командного клімату. Підвищення індексу психологічної безпеки свідчить про зростання довіри між членами команди, їхню готовність відкрито обговорювати труднощі й ділитися зворотним зв'язком без побоювань негативних наслідків. Це створює умови для ініціативності, креативності та конструктивної співпраці.

Поліпшення морального стану команди є важливим фактором, що безпосередньо впливає на якість виконання задач та стійкість команди до стресових ситуацій.

Загалом запропонована система інструментів довела свою високу результативність. Дані, наведені в узагальнюючій таблиці 3.5, демонструють позитивну динаміку всіх ключових показників, що характеризують ефективність роботи Agile-команди. Це дозволяє стверджувати, що розроблені моделі формують комплексну основу для довгострокового розвитку команди, забезпечують стійке підвищення продуктивності та створюють сприятливе середовище для впровадження гнучких методологій. Їхня універсальність і адаптивність дають можливість застосовувати запропоновані підходи в інших компаніях та проєктах, незалежно від масштабу чи предметної області [20, 21].

Таким чином, інструменти, розроблені в цьому розділі, утворюють цілісну й ефективну модель удосконалення процесу формування та розвитку Agile-команди. Вони не лише усувають виявлені проблеми, а й створюють передумови для сталого розвитку колективу, якого вимагає сучасна практика управління проєктами у сфері розробки програмного забезпечення. Отримані результати підтверджують доцільність і ефективність впроваджених рішень та обґрунтовують їхнє значення в контексті дослідження.

РОЗДІЛ 4. ОХОРОНА ПРАЦІ

4.1. Загальні положення охорони праці в ІТ-проектах

Охорона праці є невід'ємною складовою системи управління будь-якою організацією та спрямована на збереження життя, здоров'я та працездатності працівників у процесі трудової діяльності. У сфері інформаційних технологій, зокрема в проектах з розробки програмного забезпечення, питання охорони праці набуває специфічного характеру, оскільки виробниче середовище не пов'язане з фізично небезпечними умовами, проте містить значні психофізіологічні та ергономічні ризики.

Інформаційні технології (далі — ІТ) є ваговою частиною системи управління підприємством та інструментальними засобами для підвищення його конкурентоспроможності. Вони забезпечуються системою методів, процесів та способів використання обчислювальної техніки й систем зв'язку для створення, збору, передачі, пошуку, оброблення та поширення інформації.

Інформація — це будь-які відомості та/або дані, які можуть бути збережені на матеріальних носіях або відображені в електронному вигляді.

Одним із напрямів державної інформаційної політики, визначених Законом України від 02.10.1992 р. № 2657-ХІІ «Про інформацію», є створення інформаційних систем і мереж інформації та забезпечення інформаційної безпеки.

Зазвичай до ІТ- фахівців зараховують працівників, які:

- обслуговують комп'ютерне обладнання та займаються ремонтом офісної техніки, налаштуваннями роботи корпоративних сайтів, усуненням неполадок із мережею «Інтернет»;
- створюють ПЗ для різних обчислювальних пристроїв (такі спеціалісти прописують код та створюють різноманітні програми. Вони безпосередньо займаються інформаційними технологіями);
- працюють із готовими інформаційними продуктами тощо.

Такі функціональні обов'язки виконують:

- інженери-розробники комп'ютерного обладнання;
- системні адміністратори;
- програмісти різних профілів;
- тестувальники ПЗ;
- розробники сайтів та інші професіонали й фахівці, професії яких передбачені у Класифікаторі професій ДК 003:2010.

Згідно із Законом України «Про охорону праці», роботодавець зобов'язаний забезпечити безпечні та нешкідливі умови праці для всіх працівників незалежно від форми власності підприємства та характеру виконуваних робіт. Для ІТ-команд, що працюють за Agile-методологіями, особливе значення має інтеграція принципів охорони праці в систему управління проєктами, оскільки такі команди характеризуються високою інтенсивністю взаємодії, швидкими ітераціями та постійними змінами вимог [23].

Робота в Agile-проєктах часто поєднує офісний, гібридний та віддалений формати, що ускладнює контроль за умовами праці та потребує підвищеної уваги до самостійної організації робочого простору працівників. У таких умовах охорона праці виходить за межі суто технічних заходів і включає питання управління навантаженням, профілактики професійного вигорання, підтримання психологічної безпеки та формування культури відповідального ставлення до власного здоров'я.

Таким чином, охорона праці в ІТ-проєктах повинна розглядатися як комплексна система заходів, що поєднує нормативно-правові, організаційні, технічні та соціально-психологічні аспекти.

4.2. Аналіз шкідливих і небезпечних факторів у діяльності ІТ-фахівців

Професійна діяльність розробників програмного забезпечення та інших учасників Agile-команди супроводжується низкою факторів, які можуть негативно впливати на стан здоров'я та працездатність працівників. Ці фактори доцільно класифікувати за характером впливу.

До фізичних факторів належать тривала робота за комп'ютером, вплив електромагнітного випромінювання від моніторів та іншого електронного обладнання, а також недостатній рівень освітлення або наявність відблисків. Хоча рівень електромагнітного випромінювання сучасної техніки відповідає нормативним вимогам, тривале перебування перед екраном може спричинити підвищене навантаження на зорову систему.

Ергономічні фактори пов'язані з неправильним облаштуванням робочого місця. Нерегульовані меблі, невідповідна висота робочого столу, неправильне розташування монітора та клавіатури призводять до статичного навантаження на опорно-руховий апарат, що може викликати захворювання хребта, біль у шиї, плечах та зап'ястях.

Особливої уваги потребують психофізіологічні фактори, які є найбільш значущими для Agile-команд. Інтенсивний розумовий труд, стислі дедлайни, багатозадачність, часті зміни пріоритетів і необхідність постійної комунікації можуть призводити до хронічного стресу, зниження концентрації уваги та емоційного виснаження. За відсутності належних заходів це підвищує ризик професійного вигорання.

До соціально-психологічних факторів належать міжособистісні конфлікти, недостатній рівень довіри в команді, нечіткий розподіл ролей та відповідальності. В умовах Agile-проектів, де значна частина рішень приймається колективно, негативний психологічний клімат може суттєво знизити ефективність роботи команди.

Отже, діяльність ІТ-фахівців потребує комплексного підходу до ідентифікації та мінімізації професійних ризиків, з акцентом на ергономіку та психоемоційний стан працівників.

4.3. Вимоги до організації безпечного та ергономічного робочого місця

Рациональна організація робочого місця є одним із ключових чинників забезпечення безпечних умов праці для ІТ-фахівців. Робоче місце розробника програмного забезпечення повинно відповідати санітарно-гігієнічним, ергономічним та технічним вимогам, що спрямовані на зниження навантаження на організм людини.

Основними вимогами до організації робочого місця є:

- використання ергономічного крісла з можливістю регулювання висоти, нахилу спинки та підтримки поперекового відділу;
- розміщення монітора на відстані 50–70 см від очей користувача, при цьому верхній край екрана має бути на рівні очей або трохи нижче;
- правильне розташування клавіатури та миші для запобігання перенапруженню кистей рук;
- забезпечення достатнього рівня освітлення, поєднання природного та штучного світла без утворення тіней і відблисків;
- підтримання оптимальних мікрокліматичних умов у приміщенні (температура повітря 20–24 °С, відносна вологість 40–60 %).

Важливим елементом охорони праці є також організація режиму праці та відпочинку. Для працівників, що тривалий час працюють за комп'ютером, рекомендується робити короткі перерви кожні 50–60 хвилин для відпочинку очей та виконання нескладних фізичних вправ.

В умовах віддаленої роботи відповідальність за дотримання ергономічних вимог частково покладається на самого працівника, проте керівництво проєкту має

інформувати команду про базові принципи безпечної організації домашнього робочого місця.

4.4. Організаційні та профілактичні заходи з охорони праці в Agile-командах

Окрім технічних і ергономічних аспектів, важливе місце в системі охорони праці посідають організаційні та профілактичні заходи, спрямовані на збереження фізичного та психоемоційного здоров'я учасників Agile-команд. Специфіка гнучких методологій управління проєктами передбачає високий рівень автономності працівників, інтенсивну командну взаємодію та постійні зміни робочих завдань, що вимагає системного підходу до управління навантаженням і профілактики негативних наслідків професійної діяльності.

Одним із ключових елементів охорони праці в Agile-проєктах є раціональне планування робочого навантаження з урахуванням реальних можливостей і компетенцій членів команди. Нерівномірний розподіл задач, характерний для стартапів і динамічних проєктів, може призводити до перевантаження окремих фахівців, зростання рівня стресу та зниження загальної ефективності роботи. У цьому контексті особливої ваги набуває контроль за дотриманням балансу між робочим часом і відпочинком, а також недопущення систематичної понаднормової роботи.

Важливу роль у профілактиці професійного вигорання відіграють регулярні командні зустрічі та ретроспективи, під час яких учасники мають можливість обговорювати не лише технічні аспекти виконання завдань, а й власний психоемоційний стан. Створення умов для відкритого обговорення труднощів сприяє своєчасному виявленню ознак перевтоми, зниження мотивації або напруженості в команді. Такий підхід дозволяє оперативно коригувати організацію праці та запобігати накопиченню негативних факторів.

Суттєвим компонентом охорони праці в Agile-командах є формування культури психологічної безпеки, за якої кожен учасник може вільно висловлювати

свою думку, повідомляти про проблеми та пропонувати шляхи їх вирішення без страху негативних наслідків. Підтримання довірчої атмосфери, чітке визначення ролей і відповідальності, а також розвиток навичок конструктивної комунікації зменшують ймовірність виникнення міжособистісних конфліктів і сприяють стабільній командній взаємодії.

Особлива відповідальність за реалізацію організаційних і профілактичних заходів покладається на керівника команди або Scrum Master, який у межах Agile-проєкту виконує не лише координаційну, а й фасилітаційну функцію. Його завданням є своєчасне виявлення проблем, пов'язаних із перевантаженням або порушенням комунікацій, а також створення умов для збереження працездатності та професійного добробуту команди. У результаті системне впровадження організаційних заходів з охорони праці сприяє підвищенню стійкості Agile-команди, зменшенню ризиків професійного вигорання та забезпеченню стабільної якості роботи в умовах динамічного проєктного середовища.

Висновки

У четвертому розділі магістерської роботи було комплексно розглянуто питання охорони праці в IT-проєктах, що реалізуються за Agile-методологіями, з урахуванням сучасних умов організації праці та специфіки діяльності команд розробки програмного забезпечення. Аналіз показав, що, попри відсутність традиційних виробничих небезпек, характерних для промислових підприємств, робота IT-фахівців супроводжується значними професійними ризиками, які мають переважно ергономічний і психофізіологічний характер.

Встановлено, що тривала робота за комп'ютером, статичні робочі пози, високе розумове навантаження та інтенсивна командна взаємодія можуть призводити до порушень опорно-рухового апарату, зорової втоми, хронічного стресу та професійного вигорання. В умовах Agile-проєктів ці ризики посилюються через швидкий темп роботи, часті зміни пріоритетів і необхідність постійної комунікації, що потребує особливої уваги до організації безпечних умов праці.

Обґрунтовано, що ефективна система охорони праці в IT-проєктах повинна базуватися на поєднанні технічних, організаційних і соціально-психологічних заходів. Рациональна організація робочого місця, дотримання ергономічних вимог і режиму праці та відпочинку є необхідними передумовами збереження працездатності працівників. Водночас не менш важливим є управління робочим навантаженням, профілактика перевтоми та створення умов для підтримання психологічної безпеки в команді.

Особливу роль у забезпеченні охорони праці в Agile-командах відіграють організаційні та профілактичні заходи, спрямовані на формування культури відповідального ставлення до здоров'я, відкритої комунікації та взаємної підтримки. Показано, що систематичне врахування психоемоційного стану учасників команди, своєчасне виявлення ознак перевантаження та залучення

механізмів командного зворотного зв'язку сприяють підвищенню стійкості колективу та зниженню ризиків професійного вигорання.

Таким чином, охорона праці в Agile-проектах повинна розглядатися не лише як виконання нормативних вимог, а як важливий елемент загальної системи управління проектом і командою. Реалізація комплексного підходу до забезпечення безпечних і здорових умов праці створює передумови для стабільної, продуктивної та ефективної роботи команд розробки програмного забезпечення. Отримані висновки підтверджують доцільність інтеграції заходів з охорони праці в систему управління Agile-командою та логічно доповнюють результати практичної частини дослідження.

РОЗДІЛ 5. ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

5.1. Загальна характеристика впливу діяльності ІТ-проектів на навколишнє середовище

Охорона навколишнього середовища є важливою складовою сталого розвитку сучасного суспільства та передбачає раціональне використання природних ресурсів у процесі господарської діяльності. Хоча сфера інформаційних технологій не належить до екологічно небезпечних галузей, діяльність ІТ-проектів має опосередкований вплив на довкілля, що зумовлений передусім споживанням електроенергії та використанням електронного обладнання.

Процеси розробки програмного забезпечення, функціонування серверної інфраструктури та застосування хмарних сервісів супроводжуються постійним енергоспоживанням. За умови використання енергії з невідновлюваних джерел це може спричиняти збільшення викидів парникових газів. Крім того, експлуатація комп'ютерної техніки та серверного обладнання призводить до утворення електронних відходів, які потребують належної утилізації.

У межах Agile-проектів розробки програмного забезпечення екологічний вплив має певні особливості. Застосування цифрових інструментів для планування, комунікації та документування сприяє скороченню використання паперових носіїв. Водночас активне використання онлайн-сервісів і систем управління проектами підвищує навантаження на серверні ресурси.

Поширення віддаленого та гібридного форматів роботи, характерних для сучасних Agile-команд, має як позитивний, так і нейтральний вплив на довкілля. Зменшення кількості поїздок до офісу сприяє скороченню транспортних викидів, однак потребує більш відповідального використання енергоресурсів у домашніх умовах.

Отже, діяльність ІТ-проектів характеризується відносно низьким, але комплексним екологічним впливом, що обумовлює необхідність врахування

екологічних аспектів у процесі управління Agile-проектами та формування екологічно відповідального підходу до їх реалізації.

5.2. Екологічні аспекти діяльності Agile-команди

Діяльність Agile-команди розробки програмного забезпечення супроводжується низкою екологічних аспектів, які мають переважно непрямий характер і пов'язані з використанням інформаційних ресурсів, електронного обладнання та цифрової інфраструктури. На відміну від виробничих підприємств, вплив ІТ-команд на навколишнє середовище не проявляється у вигляді безпосередніх викидів або відходів, однак формується через сукупність організаційних і технологічних процесів.

Основним екологічним аспектом діяльності Agile-команди є споживання електроенергії, необхідної для функціонування робочих місць, серверної інфраструктури та хмарних сервісів. Постійне використання комп'ютерної техніки, систем управління проектами, засобів комунікації та тестових середовищ зумовлює стабільне навантаження на енергетичні ресурси. У масштабі зростаючих цифрових продуктів це може призводити до збільшення екологічного сліду ІТ-проектів.

Важливим аспектом є також використання та оновлення електронного обладнання. У процесі діяльності Agile-команд відбувається поступове моральне старіння комп'ютерної техніки, серверів і периферійних пристроїв, що в перспективі формує потік електронних відходів. За відсутності системного підходу до повторного використання або утилізації обладнання це може створювати додаткове екологічне навантаження.

Застосування гнучких методологій управління проектами сприяє активному використанню цифрових інструментів для планування, комунікації та зберігання інформації. Це дозволяє значною мірою зменшити обсяги паперової документації та скоротити використання матеріальних ресурсів. Водночас така цифровізація

підвищує залежність від серверних потужностей і дата-центрів, що також слід враховувати при оцінці екологічних аспектів діяльності команди.

Поширення віддаленого та гібридного форматів роботи в Agile-командах впливає на екологічні показники неоднозначно. Скорочення офісних площ і транспортних переміщень позитивно впливає на зменшення викидів, однак підвищує загальне енергоспоживання в побутових умовах. Це потребує відповідального ставлення учасників команди до використання ресурсів поза межами офісу.

Таким чином, екологічні аспекти діяльності Agile-команди формуються на перетині цифрових процесів, енергоспоживання та організації роботи. Усвідомлення цих аспектів створює основу для впровадження заходів, спрямованих на зменшення негативного впливу на навколишнє середовище, що буде розглянуто у наступному підрозділі.

5.3. Заходи з мінімізації негативного впливу ІТ-проєктів на навколишнє середовище

Мінімізація негативного впливу діяльності ІТ-проєктів на навколишнє середовище є важливим елементом екологічно відповідального управління та складовою концепції сталого розвитку. Для Agile-проєктів розробки програмного забезпечення реалізація таких заходів не потребує кардинальних змін у виробничих процесах, проте вимагає системного підходу до використання ресурсів і цифрової інфраструктури.

Одним із ключових напрямів зменшення екологічного навантаження є раціональне використання електроенергії. Оптимізація програмного коду, зменшення надлишкових обчислень, ефективне використання серверних ресурсів та перехід на хмарні платформи з енергоефективними дата-центрами дозволяють скоротити енергоспоживання без негативного впливу на якість продукту. Важливу

роль відіграє також використання сучасного енергоефективного обладнання на робочих місцях учасників команди.

Суттєвим чинником зниження впливу на довкілля є скорочення матеріальних витрат за рахунок цифровізації робочих процесів. Використання електронного документообігу, систем управління проєктами та онлайн-інструментів комунікації дає змогу майже повністю відмовитися від паперових носіїв, що сприяє збереженню природних ресурсів і зменшенню кількості відходів.

Управління життєвим циклом електронного обладнання також має значення для екологічної безпеки ІТ-проєктів. Рациональне використання техніки, подовження терміну її експлуатації, повторне використання компонентів та передача обладнання на спеціалізовану утилізацію дозволяють зменшити негативний вплив електронних відходів на навколишнє середовище.

Важливим чинником мінімізації екологічного впливу є поширення віддаленого та гібридного форматів роботи, характерних для Agile-команд. Скорочення щоденних поїздок працівників до офісу зменшує транспортні викиди та енергетичні витрати, пов'язані з утриманням офісних приміщень. Водночас ефективність цього підходу залежить від усвідомленого ставлення учасників команди до споживання ресурсів у побутових умовах.

Таким чином, заходи з мінімізації негативного впливу ІТ-проєктів на навколишнє середовище мають комплексний характер і поєднують технологічні, організаційні та поведінкові аспекти. Їх впровадження сприяє зниженню екологічного навантаження без зменшення ефективності роботи Agile-команд та відповідає сучасним принципам відповідального управління проєктами.

5.4. Сталий розвиток та екологічна відповідальність в ІТ-проєктах

Сталий розвиток є важливим орієнтиром сучасної діяльності в галузі інформаційних технологій і передбачає поєднання технологічного прогресу з раціональним використанням природних ресурсів. Для ІТ-проєктів, зокрема тих,

що реалізуються за Agile-методологіями, екологічна відповідальність полягає у свідомому підході до організації робочих процесів, використання цифрової інфраструктури та енергетичних ресурсів.

Ключові цифрові технології для забезпечення сталого розвитку, які виступають драйверами «зеленої» трансформації:

- Штучний інтелект (ШІ). Системи ШІ сьогодні оптимізують виробничі процеси та енергоспоживання. У промисловості алгоритми AI дозволяють точніше налаштовувати обладнання, підвищуючи енергоефективність і зменшуючи відходи. Наприклад, прогнозування попиту на електроенергію за допомогою ШІ дає змогу операторам електромереж збалансувати навантаження та скоротити втрати енергії. Такі розумні системи сприяють і проектуванню більш сталих продуктів, і підвищенню ефективності будівель (напрямку оптимізуючи роботу кліматичних систем) [23, 24].

- Інтернет речей (IoT). Мережі сенсорів та підключених пристроїв забезпечують моніторинг у реальному часі споживання енергії, води та інших ресурсів на виробництві, у 160 містах і будинках. Завдяки IoT компанії можуть відстежувати та автоматично регулювати процеси – від температури в «розумних» будівлях до стану промислових верстатів – що допомагає негайно виявляти перевитрати та скорочувати викиди. В енергетиці датчики IoT надають дані для «розумних мереж» і децентралізованих систем, інтегруючи відновлювані джерела та підвищуючи надійність і сталість енергопостачання [23, 24].

- Аналіз великих даних. Збір та обробка великих масивів даних створює основу для прийняття науково обґрунтованих рішень у сфері сталого розвитку. Платформи Big Data дають можливість урядам і бізнесу отримувати безпрецедентну прозорість щодо викидів, споживання ресурсів та ефективності заходів. Наприклад, аналітичні інструменти допомагають моделювати сценарії скорочення викидів CO₂, відслідковувати прогрес до цілей сталого розвитку та оптимізувати логістичні ланцюги, мінімізуючи «вуглецевий слід». Дані в режимі

реального часу, об'єднані з машинним навчанням, дозволяють прогнозувати екологічні ризики та оперативно реагувати на них [23, 24].

- Блокчейн. Розподілений реєстр (DLT) застосовується для забезпечення прозорості й простежуваності в ланцюгах постачання та екологічних проєктах. Наприклад, блокчейнтехнології можуть підтверджувати автентичність «зелених» сертифікатів (на відновлювану енергію чи вуглецеві кредити), гарантувати, що продукція та сировина відповідають стандартам сталості, а також полегшувати торгівлю чистою енергією між виробниками і споживачами. ЄС особливо підтримує розвиток стійких блокчейн-рішень – таких, що допомагають протидіяти зміні клімату та реалізувати цілі Європейського зеленого курсу. Уже сьогодні існують пілотні блокчейн-системи для відстеження викидів у логістиці, контролю сталості рибальства та управління розподілом відновлюваної енергії [25].

- Хмарні обчислення. Хмарні платформи дозволяють компаніям відмовитися від надлишкового фізичного обладнання і користуватися обчислювальними ресурсами за потребою, що підвищує ефективність використання техніки. Великі провайдери хмарних сервісів активно переходять на відновлювану енергію для живлення своїх дата-центрів і впроваджують енергоощадні технології (наприклад, рідинне охолодження серверів, утилізацію тепла). Завдяки ефекту масштабу хмари сприяють скороченню сукупного енергоспоживання ІТ-інфраструктури та відходів електроніки [26]. Крім того, хмарні технології тісно інтегровані з IoT та AI-рішеннями: через хмару збираються й обробляються дані від мільйонів сенсорів, що допомагає компаніям у будь-який момент оцінити свої екологічні показники та гнучко реагувати на них в масштабі всієї організації.

Інтеграція принципів сталого розвитку в управління Agile-проєктами сприяє зменшенню негативного впливу на навколишнє середовище, підвищенню ефективності використання ресурсів і формуванню екологічно відповідальної культури в команді. Такий підхід відповідає сучасним вимогам до соціально відповідального ведення бізнесу та довгострокового розвитку ІТ-організацій.

Висновки

У п'ятому розділі магістерської роботи розглянуто основні екологічні аспекти діяльності ІТ-проектів, що реалізуються за гнучкими методологіями управління. Визначено, що вплив Agile-команд на навколишнє середовище має переважно опосередкований характер і проявляється через споживання енергетичних ресурсів, використання електронного обладнання та цифрової інфраструктури.

Проаналізовано екологічні аспекти діяльності Agile-команд і обґрунтовано доцільність впровадження заходів, спрямованих на мінімізацію негативного впливу на довкілля. Запропоновані підходи базуються на принципах енергоефективності, цифровізації процесів і відповідального використання ресурсів та не потребують значних змін у структурі проектів.

Отримані висновки свідчать, що врахування екологічних чинників у процесі управління ІТ-проектами сприяє реалізації принципів сталого розвитку та формуванню екологічно відповідального підходу до діяльності Agile-команд.

ВИСНОВКИ

У магістерській роботі здійснено комплексне дослідження процесу формування та розвитку команд розробки програмного забезпечення в умовах застосування Agile-методологій. Актуальність обраної теми зумовлена зростаючою роллю командної взаємодії в сучасних IT-проектах, високою динамікою змін вимог, поширенням віддалених і гібридних форматів роботи, а також необхідністю підвищення продуктивності й стійкості команд у конкурентному середовищі.

У першому розділі роботи було проаналізовано теоретичні основи командної роботи в IT-проектах і визначено ключові концепції, що описують динаміку та ефективність командної взаємодії. Розглянуто моделі розвитку команд, зокрема модель Б. Такмана, поведінкові ролі за М. Белбіном, а також принципи та цінності Agile-методологій. Встановлено, що гнучкі підходи до управління проектами істотно трансформують традиційні уявлення про командоутворення, зміщуючи акцент із формальної ієрархії на самоорганізацію, прозору комунікацію та спільну відповідальність за результат. Теоретичний аналіз дозволив виявити обмеження існуючих інструментів формування команд, зокрема недостатню увагу до soft skills, психологічної безпеки та системного управління навантаженням.

У другому розділі здійснено аналітичне дослідження процесу формування та функціонування Agile-команди на прикладі умовного стартапу TeamFlow. Проаналізовано організаційну структуру команди, розподіл ролей, рівень компетенцій учасників, особливості реалізації Scrum-процесів та основні показники ефективності роботи. Результати аналізу засвідчили наявність низки системних проблем, зокрема нестабільність velocity, нерівномірний розподіл навантаження, залежність від окремих ключових фахівців, недостатню структурованість комунікацій і ознаки психологічної напруги в команді. Отримані висновки підтвердили необхідність удосконалення інструментів формування Agile-команди та стали підґрунтям для розроблення практичних рішень.

Третій розділ роботи присвячено розробленню та обґрунтуванню комплексу інструментів удосконалення процесу формування Agile-команди. Запропоновано системну модель, що включає інструменти оцінювання компетенцій, оптимізації комунікацій, балансування командного навантаження, управління метриками та підтримання психологічної безпеки. Практична апробація запропонованих рішень у межах досліджуваного стартапу продемонструвала позитивну динаміку ключових показників роботи команди, зокрема підвищення стабільності velocity, скорочення cycle time, зменшення кількості дефектів та покращення командного клімату. Це підтвердило ефективність запропонованих інструментів і їхню придатність для використання в реальних Agile-проектах.

У четвертому розділі розглянуто питання охорони праці в IT-проектах з урахуванням специфіки Agile-команд. Проаналізовано основні шкідливі та небезпечні фактори, характерні для діяльності IT-фахівців, серед яких домінують ергономічні та психофізіологічні ризики. Обґрунтовано важливість раціональної організації робочого місця, дотримання режиму праці та відпочинку, а також впровадження організаційних заходів, спрямованих на профілактику професійного вигорання та підтримання психологічної безпеки. Показано, що охорона праці в Agile-проектах є невід'ємною складовою ефективного управління командою та безпосередньо впливає на продуктивність і стійкість командної взаємодії.

П'ятий розділ присвячено аналізу екологічних аспектів діяльності IT-проектів та питанню охорони навколишнього середовища. Встановлено, що вплив Agile-команд на довкілля має переважно опосередкований характер і пов'язаний зі споживанням електроенергії, використанням електронного обладнання та цифрової інфраструктури. Обґрунтовано доцільність впровадження заходів з мінімізації негативного впливу на навколишнє середовище шляхом енергоефективності, цифровізації процесів, відповідального використання ресурсів і підтримки принципів сталого розвитку. Показано, що інтеграція екологічної

відповідальності в управлінні Agile-проєктами відповідає сучасним вимогам соціально відповідального ведення бізнесу.

У результаті проведеного дослідження досягнуто поставленої мети магістерської роботи — розроблено та обґрунтовано методи удосконалення інструментів формування команди розробки програмного забезпечення в Agile-проєктах. Виконання поставлених завдань дозволило сформувати цілісне бачення процесу командування, поєднати теоретичні положення з практичними інструментами та довести ефективність запропонованих рішень.

Практична цінність роботи полягає в можливості використання розроблених інструментів і моделей у діяльності ІТ-компаній, стартапів і команд розробки програмного забезпечення незалежно від масштабу проєкту. Отримані результати можуть бути використані для підвищення ефективності командної роботи, зниження ризиків професійного вигорання, покращення управління навантаженням і формування стійких Agile-команд.

Таким чином, магістерська робота має завершений характер, містить науково обґрунтовані висновки та практичні рекомендації, а її результати підтверджують доцільність комплексного підходу до формування та розвитку Agile-команд у сучасних умовах розвитку ІТ-галузі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Schwaber K., Sutherland J. *The Scrum Guide*. — Scrum.org, 2020. — URL: <https://scrumguides.org>
2. Anderson D. J. *Kanban: Successful Evolutionary Change for Your Technology Business*. — Blue Hole Press, 2010.
3. Beck K. et al. *Manifesto for Agile Software Development*. — 2001. — URL: <https://agilemanifesto.org>
4. Tuckman B. W. Developmental sequence in small groups // *Psychological Bulletin*. — 1965. — Vol. 63, No. 6. — P. 384–399.
5. Belbin R. M. *Team Roles at Work*. — 2nd ed. — Butterworth-Heinemann, 2010.
6. Edmondson A. C. *The Fearless Organization: Creating Psychological Safety in the Workplace for Learning, Innovation, and Growth*. — Wiley, 2018.
7. Lencioni P. *The Five Dysfunctions of a Team*. — Jossey-Bass, 2002.
8. Highsmith J. *Agile Project Management: Creating Innovative Products*. — Addison-Wesley, 2009.
9. Kniberg H., Skarin M. *Kanban and Scrum – Making the Most of Both*. — C4Media, 2010.
10. Cockburn A. *Agile Software Development: The Cooperative Game*. — 2nd ed. — Addison-Wesley, 2006.
11. Forsgren N., Humble J., Kim G. *Accelerate: The Science of Lean Software and DevOps*. — IT Revolution Press, 2018.
12. ISO 45001:2018. *Occupational Health and Safety Management Systems — Requirements with Guidance for Use*. — International Organization for Standardization, 2018.
13. World Health Organization. *Healthy Workplaces: A Model for Action*. — WHO Press, 2010.

14. International Labour Organization. *Workplace Stress: A Collective Challenge*. — ILO, 2016.
15. ISO 14001:2015. *Environmental Management Systems — Requirements with Guidance for Use*. — International Organization for Standardization, 2015.
16. United Nations. *Transforming our world: the 2030 Agenda for Sustainable Development*. — 2015. — URL: <https://sdgs.un.org>
17. OECD. *Green Growth Indicators*. — OECD Publishing, 2017.
18. Google re:Work. *Guide: Psychological Safety*. — URL: <https://rework.withgoogle.com>
19. McKinsey & Company. *Agile Organizations: Design, Operations and Performance*. — 2018. — URL: <https://www.mckinsey.com>
20. European Agency for Safety and Health at Work. *Work-related Stress*. — EU-OSHA, 2020. — URL: <https://osha.europa.eu>
21. IEEE Computer Society. *Software Engineering Body of Knowledge (SWEBOK)*. — IEEE, 2014.
22. Закон України «Про охорону праці» // Відомості Верховної Ради України. — URL: <https://zakon.rada.gov.ua>
23. European Commission. (2024, February 15). Green digital sector – Shaping Europe’s digital future. Retrieved from <https://digital-strategy.ec.europa.eu/en/policies/green-digital>
24. Дегтяр, В. О., & Загвойська, Л. Д. (2024). Значення інформаційних технологій в інноваційному та економічному зростанні України. *Scientific Bulletin of UNFU*, 34(7), 30-36. <https://doi.org/10.36930/40340704>
25. European Commission. (2024, December 12). Blockchain and Web3 Strategy – Shaping Europe’s digital future. Retrieved from <https://digital-strategy.ec.europa.eu/en/policies/blockchainstrategy>
26. European Commission. (2021, March 19). Companies take action to support the green and digital transformation of the EU (DIGIBYTE news). Retrieved from

[https://digitalstrategy.ec.europa.eu/en/news/companies-take-action-support-green
and-digital-transformation-eu](https://digitalstrategy.ec.europa.eu/en/news/companies-take-action-support-green-and-digital-transformation-eu) .