

УДК: 004.4:005.8

DOI: <https://doi.org/10.30838/EP.201.30-36>**Гарматюк О.В.**

доктор філософії з економіки

Уманський державний педагогічний університет імені Павла Тичини

Garmatiuk Olena

PhD in Economic Sc.

Pavlo Tychyna Uman State Pedagogical University

<https://orcid.org/0000-0002-2589-3269>**Караман О.М.**

Уманський державний педагогічний університет імені Павла Тичини

Karaman Olesia

Pavlo Tychyna Uman State Pedagogical University

СУЧАСНІ СИСТЕМИ УПРАВЛІННЯ ПРОЄКТАМИ В ІТ-СЕКТОРІ

У сучасному цифровому середовищі розробка комп'ютерного програмного забезпечення входить до ТОП-5 галузей за формуванням потоку несировинного експорту для низки країн. ІТ-компанії активно впроваджують сучасні технології у свою діяльність, а ефективне управління проєктами є ключовим чинником успішної реалізації інноваційних розробок. Керування проєктами охоплює координацію всіх дій, що передують ініціації проєкту, а також супроводжують його реалізацію до завершення.

У статті розглядаються як традиційні (Waterfall, PRINCE2), так і гнучкі підходи до управління проєктами (Agile, SAFe). Особливу увагу приділено порівняльному аналізу каскадних та гнучких методологій. Основною метою порівняння є пошук відповідної системи керування для вирішення проблем, що виникають у проєктах, які потребують одночасного поєднання гнучкості та структурованого планування, особливо в умовах складної розробки програмного забезпечення.

Підкреслено роль культури управління проєктами, яка включає підтримку з боку вищого керівництва, залученість стейкхолдерів, комунікаційні механізми, а також процедури оцінювання результатів та трансферу знань.

Ключові слова: управління проєктами, каскадна модель, гнучка методологія, підхід SAFe, Скрам, розробка програмного забезпечення, ітерація.

MODERN PROJECT MANAGEMENT SYSTEMS IN THE IT SECTOR

In the context of the modern digital environment, the development of computer software has emerged as one of the top five industries driving non-resource export flows in numerous countries. This dynamic growth underscores the strategic importance of effective project management, particularly in the information technology (IT) sector, where the implementation of innovative solutions requires both flexibility and structured coordination. The main goal of the article is to explore the relevance and applicability of diverse project management approaches in response to the increasing complexity of software development initiatives. The study investigates traditional project management models, such as Waterfall and PRINCE2, alongside Agile methodologies, including Scaled Agile Framework (SAFe). The key objective is to identify the optimal methodology for projects that demand a balance between adaptive execution and detailed pre-planning.

To address the research objective, the methodology involved a comparative analysis of traditional and Agile project management practices. This approach enabled the authors to evaluate their operational efficiency, degree of stakeholder involvement, and responsiveness to change. Through this analysis, the article identifies the core advantages and limitations of each method, offering insights into the contexts in which they deliver the best results. The results of the research demonstrate that no universal solution exists; instead, the effective selection and implementation of a project management methodology depend on the unique requirements of each IT project. The study emphasizes that hybrid approaches are increasingly relevant in practice, allowing companies to tailor methodologies to suit specific technical challenges and organizational needs. Furthermore, the research highlights the significance of cultivating a robust project management culture that empowers teams, encourages continuous improvement, and aligns project objectives with broader business goals.

The practical value of this article lies in its guidance for IT companies seeking to enhance their project delivery capabilities through informed methodological choices. It offers strategic recommendations for adapting project

management frameworks in environments characterized by rapid technological change and evolving customer demands.

Keywords: project management, Waterfall, Agile methodology, SAFe approach, Scrum, software development, iteration.

JEL classification: D83, L86, M11, O32

Постановка проблеми. Протягом останнього десятиліття інформаційно-технологічний сектор в Україні посідає дедалі важливіше місце в структурі національної економіки, зокрема завдяки формуванню сталого потоку експорту продукції з високою доданою вартістю, що не належить до сировинного типу. Динаміка зростання експорту ІТ-послуг є надзвичайно показовою: якщо у 2010 році обсяги такого експорту становили лише десятки мільйонів доларів США за квартал, то вже у 2021 році цей показник досягнув рекордного рівня – 2,1 мільярда доларів США [1]. Подібне зростання свідчить про високу продуктивність та конкурентність ІТ-компаній на глобальному ринку та загальносвітовий тренд на розвиток та впровадження високо-технологічних продуктів в бізнес та повсякденне життя.

Одним із ключових чинників успішної діяльності суб'єктів підприємницької діяльності у сфері інформаційних технологій є ефективна система управління проектами, яка безпосередньо впливає на їх фінансову стабільність та стійкість до зовнішніх викликів. Процес управління проектами характеризується високим рівнем динамічності, що зумовлює потребу в адаптивному підході, раціональному розподілі ресурсів та впровадженні сучасних методологій керування.

Зважаючи на безперервний розвиток цифрових технологій та підвищену конкуренцію на світовому ринку, актуальність ефективного управління проектами постійно зростає. Застосування сучасних методологій керування дозволяє ІТ-компаніям оперативніше реагувати на зміни, скорочувати час виходу продукту на ринок (time to market) та забезпечувати високу якість кінцевого результату. Дослідження особливостей функціонування систем управління проектами в ІТ-компаніях є необхідною умовою для виявлення основних проблем та чинників, що визначають успішність реалізації проектів у цій сфері.

Аналіз останніх досліджень та публікацій. Теоретичні та практичні аспекти управління проектами є предметом численних наукових досліджень, які сформували міждисциплінарну галузь знань, що активно розвивається як на міжнародному, так і на національному рівнях. Серед провідних зарубіжних дослідників, які зробили вагомий внесок у розвиток теорії та методології управління проектами, слід відзначити Скотта Беркуна, Кена Швабера, Кіма Гелдмана, Гарольда Керцнера, Міка Керстена, Джеффа Сазерленда. У вітчизняному науковому середовищі проблематика управління проектами представлена працями таких науковців, як Бушуєв С.Д., Бушуєва Н.С., Петренко Н.О., Ноздріна Л.В., Крижановський Є.М. та інші.

Значний пласт знань акумульовано у напрацюваннях спеціалізованих професійних організацій, які

об'єднують фахівців з управління проектами. До таких належать Міжнародна асоціація з управління проектами (International Project Management Association, IPMA), професійні національні асоціації, а також Інститут управління проектами (Project Management Institute, PMI), який створив глобально визнану сертифікаційну систему знань – Project Management Body of Knowledge (PMBOK). Цей підхід охоплює такі аспекти, як багаторівневе структурування цілей, інтеграцію ІТ-проектів у бізнес-середовище підприємства, а також матричну організаційну структуру управління проектами, де важлива роль належить координатору або менеджеру проекту [2].

Окрему нішу в сучасних дослідженнях займає аналіз гнучких методологій управління проектами, зокрема таких, як Scrum та SAFe, які широко застосовуються в ІТ-секторі. У численних публікаціях підкреслюється ефективність Agile у реалізації ІТ-проектів завдяки його гнучкості, ітеративності та здатності швидко адаптуватися до змін. Водночас, актуальними залишаються питання управління ризиками, які суттєво впливають на результативність проектів, особливо в умовах високої динаміки та невизначеності ІТ-середовища. Зокрема, також наявний конфлікт між принципами гнучких методологій (які передбачають очні зустрічі та постійну комунікацію) та реаліями розподіленого середовища, де команди взаємодіють переважно віддалено. Цей виклик зумовлює необхідність подальшого глибокого аналізу та адаптації методологій до умов розподіленої проектною роботи та швидко змінних пріоритетів [3, С. 85].

Потребує додаткових досліджень аналіз каскадних та гнучких методологій та пошук відповідної системи керування для вирішення проблем, що виникають у проектах, які потребують одночасного поєднання гнучкості та структурованого планування, особливо в умовах складної розробки програмного забезпечення.

Мета дослідження є аналіз ефективності застосування традиційних та гнучких підходів до управління проектами в ІТ-секторі, зокрема в умовах складної розробки програмного забезпечення. Автори прагнуть визначити оптимальну систему керування, яка забезпечує баланс між необхідною гнучкістю та структурованим плануванням, а також підвищує результативність реалізації інноваційних ІТ-проектів.

Виклад основних результатів дослідження. Об'єктом даного дослідження було обрано ІТ-індустрію, насамперед у зв'язку з динамічним розвитком цифрових технологій та зростанням частки доходів держави, сформованих за рахунок надання послуг у сфері інформаційних технологій. Зокрема, лише ІТ-індустрія згідно даних Нацбанку за 2024 рік принесла до бюджету 6,45 млрд гривень завдяки експорту та надалі

залишається найбільшим експортером послуг в Україні. У загальній структурі експорту країни ІТ-сектор посідає друге місце з 11,5%, поступаючись лише продовольчим товарам [4].

Інформаційно-технологічні (ІТ) проекти є специфічним типом проектів, що вимагають особливого підходу до управління. Вони мають спільні риси з іншими проектами, але також містять суттєві відмінності, такі як складність, масштабність та різноманітність завдань. ІТ-проекти мають низку специфічних особливостей, які значно впливають на створення ефективної системи управління. Вони залежать від швидкого розвитку технологій, високої складності інформаційних систем та великої кількості зацікавлених сторін. Динамічний характер таких проектів зумовлений постійними змінами в ІТ-інфраструктурі, що потребує регулярного аналізу та внесення коректив. Крім того, ІТ-проекти характеризуються високим рівнем ризиків та невизначеності.

Ураховуючи вищезазначену специфіку, управління ІТ-проектами потребує врахування таких важливих факторів, як адаптивність, оперативність у прийнятті рішень, ефективна комунікація та співпраця в команді, а також здатність своєчасно керувати змінами та ризиками [3, С. 38].

У 60-х роках ХХ століття сфера управління проектами почала активно розширюватися. Пошук шляхів оптимізації витрат на організацію проектів призвів до розробки першої системи управління проектами компанією IBM. Згодом з'являються професійні організації з управління проектами, що об'єднують провідних фахівців Європи. Однією з таких була неприбуткова організація International Project Management Association (IPMA), заснована у 1965 році, яка займалася розробкою стандартів управління проектами [5].

У 1969 році виникає перший інститут управління проектами як неприбуткова організація, метою якої було сприяння розвитку професійного управління проектами та розробка методології щодо комунікацій всередині проекту. У 1980-х роках були розроблені низка стандартів проектного менеджменту, а у 1987 році був опублікований перший довідник РМВОК, в якому були зібрані основні практики для ефективного управління проектами в різних галузях. Це стало підґрунтям для формування управління проектами як окремої дисципліни [5].

Одним із найбільш традиційних підходів до управління проектами вважається методологія Waterfall (у перекладі з англійської – «водоспад»). Цей підхід був розроблений у 1970 році з метою оптимізації процесів розробки програмного забезпечення. Назва методології відображає її сутність – каскадну організацію виконання проекту, при якій кожен наступний етап розпочинається лише після завершення попереднього [6].

Методологія Waterfall є лінійним підходом до управління проектами, що передбачає поетапну реалізацію завдань відповідно до заздалегідь визначених вимог зацікавлених сторін. Цей підхід вирізняється

простотою у розумінні та передбачає розроблення детального плану, якого команда проекту має послідовно дотримуватися. На відміну від гнучких (agile) методологій, у яких вимоги та плани можуть змінюватися в процесі реалізації, Waterfall орієнтується на жорстке дотримання початково встановлених параметрів проекту.

Класична каскадна модель управління передбачає реалізацію п'яти основних фаз, кожна з яких починається лише після завершення попередньої:

1. *Фаза визначення вимог* – спрямована на фіксацію ключових потреб замовника, які ляжуть в основу подальших етапів планування та реалізації проекту;
2. *Фаза проектування* – полягає в створенні технічного проекту, формулюванні кінцевих цілей та визначенні шляхів їх досягнення;
3. *Фаза реалізації* – передбачає безпосередню розробку продукту відповідно до затвердженого плану;
4. *Фаза перевірки* – здійснюється з метою підтвердження відповідності кінцевого продукту раніше узгодженим вимогам замовника;
5. *Фаза технічного обслуговування* – забезпечує виявлення та виправлення дефектів, що виникають у процесі використання продукту, а також регулярне оновлення та підтримку працездатності системи [5].

Серед переваг методології Waterfall слід виокремити чітко визначений бюджет та графік реалізації проекту, адже його цілі та вимоги залишаються незмінними протягом усього життєвого циклу. Такий підхід сприяє структурованості процесу та підвищенню передбачуваності результату. Водночас до суттєвих недоліків цього підходу належить обмежена гнучкість: для внесення змін необхідно повертатися до попередніх етапів, що може призвести до подовження термінів реалізації та зростання витрат [2].

До каскадних моделей керування ІТ проектами також відноситься PRINCE2 (Projects IN Controlled Environments) – це одна з провідних у світі методологій управління проектами, що отримала міжнародне визнання: понад мільйон виданих сертифікатів та використання у більш ніж 150 країнах світу. У Великій Британії PRINCE2 вважається основним стандартом управління проектами. Методологія PRINCE2 бере свій початок від оригінального стандарту PRINCE, опублікованого урядом Великої Британії у 1989 році. Цей стандарт був модифікацією раніше розробленої методики PROMPT (Project Resource Organization Management Planning Techniques) у 1970-х роках. Спочатку PRINCE застосовувався переважно для ІТ-проектів та характеризувався надмірною жорсткістю, що ускладнювало його використання в інших сферах [7].

У 1996 році Центральне агентство з питань обчислювальної техніки та телекомунікацій (CCTA) опублікувало оновлену та більш універсальну версію – PRINCE2. Вона була спеціально розроблена для широкого застосування у проектах будь-якого типу. Відтоді методологія неодноразово оновлювалася з

урахуванням сучасних вимог та кращих практик управління. У 2023 році було опубліковано останню версію – PRINCE2 . Збереження назви «PRINCE2» (на відміну від «PRINCE3») підкреслює вірність базовим принципам методології.

Сутність PRINCE2 виходить за межі звичайної методики – це цілісна система управління проектами, яка забезпечує структурований та контрольований підхід до реалізації проектів різного масштабу й складності. На відміну від інших підходів (зокрема PMI або IPMA), PRINCE2 не концентрується на детальному описі виконання окремих завдань, а акцентує увагу на загальній організаційній структурі та контролі за ходом реалізації проекту [7].

Методологія PRINCE2 включає:

- чітко визначені процеси для управління всім життєвим циклом проекту;
- конкретні ролі та сфери відповідальності;
- стандартизовані управлінські шаблони (зокрема, бізнес-кейс, реєстр ризиків);
- фіксовані методи орієнтації на результат та контролю за виконанням проекту [7].

Однією з ключових особливостей PRINCE2 є її продукто-орієнтований підхід: основна увага зосереджена не на діяльності як такій, а на результатах – створених продуктах. Методологія також передбачає постійне обґрунтування бізнес-доцільності проекту: його реалізація має бути економічно виправданою та генерувати реальну цінність для бізнесу.

Базуючись на модульній структурі, PRINCE2 легко адаптується до потреб будь-якої галузі, організації чи проекту, незалежно від розміру. Завдяки цьому вона вважається міжнародно визнаним підходом до ефективного планування, реалізації та завершення високотехнологічних проектів, проте не обмежується ними [7].

У порівнянні з альтернативними підходами до управління проектами, PRINCE2 має низку важливих переваг:

- *Структурований та масштабований підхід* – на відміну від PMBOK, який є радше збірником знань, PRINCE2 пропонує конкретну методологію з чітко визначеними процесами та ролями;
- *Орієнтація на процеси* – забезпечує прозорі кроки реалізації й механізми ухвалення рішень;
- *Гнучкість* – PRINCE2 легко адаптується до специфіки конкретного проекту чи організації;
- *Бізнес-обґрунтування* – кожен проект повинен мати економічну доцільність, що підлягає постійному перегляду;
- *Інтегроване управління ризиками* – на відміну від багатьох гнучких методів, PRINCE2 враховує ризики з самого початку;
- *Сумісність з Agile-підходами* – методологія може ефективно комбінуватися з фреймворками Scrum або Kanban, зокрема через розширення PRINCE2 Agile;
- *Міжнародне визнання* – сертифікація PRINCE2 високо цінується в усьому світі як

підтвердження кваліфікації керівника проектів [6].

PRINCE2 базується на чітко визначеній структурі, яка включає сім основоположних принципів, сім ключових практик та сім груп процесів. Ця взаємодія забезпечує системне планування, управління та успішне завершення проектів. Основоположні принципи задають універсальні правила, яких слід дотримуватись у кожному проекті PRINCE2, тоді як ключові практики охоплюють основні аспекти управління: бізнес-кейс, організацію, управління ризиками тощо [7].

На противагу класичним моделям управління широкої популярності набули гнучкі методології управління проектами. Agile (від англ. *agile* – «гнучкий») є одним із найпопулярніших підходів у сучасному управлінні IT проектами, що пропонує революційний підхід до організації робочих процесів. Ця методологія відкидає жорстку ієрархію та послідовність етапів, замінюючи їх на гнучкі форми організаційної взаємодії [6].

Основні принципи Agile викладені в документі «Manifesto for Agile Software Development» (2001 р.), де акцент зроблено на таких ключових цінностях:

1. *Люди та взаємодія* важливіші за інструменти та процеси.
2. *Працюючий продукт* важливіший за вичерпну документацію.
3. *Співпраця із замовником* важливіша за формальне дотримання контракту.
4. *Готовність до змін* важливіша за дотримання жорсткого плану [8].

Agile передбачає поділ проекту на короткі цикли, які називаються ітераціями або спринтами. Кожна ітерація охоплює всі етапи життєвого циклу проекту та завершується створенням конкретного результату, інкременту, який може бути оцінений, доопрацьований або реалізований у наступних циклах.

Основні переваги Agile:

- *Гнучкість*. Команда може швидко реагувати на зміни, адаптуючи процеси до нових вимог замовника чи ринку.
- *Тісна співпраця*. Постійна комунікація між усіма зацікавленими сторонами забезпечує прозорість процесів.
- *Зосередженість на цінності*. Визначення короткострокових цілей сприяє швидкому досягненню результатів із високою бізнес-цінністю.
- *Оптимізація роботи команди*. Команда регулярно аналізує свою роботу, виявляє слабкі місця та вдосконалює процеси [8].

Згідно з дослідженнями Гвоздя М.Я. та Злидника Ю.О., ключові принципи Agile ґрунтуються на кількох важливих аспектах. По-перше, гнучкість, яка передбачає адаптацію до змін зовнішнього середовища, що дозволяє девелопмент командам швидко реагувати на виклики. По-друге, безперервність, яка забезпечує підтримання оптимального темпу роботи та стабільний розвиток проекту. Важливим є й принцип єдності, що акцентує увагу на тісній співпраці з клієнтом та

глибокому розумінні його потреб [9, С. 233].

Також значну роль відіграє мотивація, яка сприяє формуванню середовища взаємопідтримки та зміцненню корпоративної культури. Простота є ще одним ключовим принципом, адже забезпечує максимальну зрозумілість продукту та його зручність для користувачів. Окрім цього, Agile орієнтується на системність, що означає інтеграцію команди як єдиного організму, здатного ефективно досягати спільних цілей. Автономія є невіддільним елементом цього підходу, оскільки сприяє самоорганізації команди, дозволяючи їй ухвалювати рішення без зайвої бюрократії. Нарешті, відповідальність кожного учасника за якість виконаної роботи є основою ефективного функціонування команди в рамках Agile [9, С. 234].

Згідно з дослідженнями Крижанівського впровадження Agile у діяльність ІТ-компаній позитивно впливає на ефективність їхньої роботи. Зокрема, використання гнучких методологій сприяє підвищенню продуктивності команд, оскільки забезпечує краще планування, координацію та швидку адаптацію до змін. Крім того, Agile дозволяє значно зменшити обсяг непродуктивної роботи, усуваючи зайві процеси та оптимізуючи взаємодію між учасниками проекту. Важливою перевагою є також можливість виявлення та усунення помилок на ранніх етапах розробки, що мінімізує ризики та підвищує якість кінцевого продукту [10, С. 68].

Agile є надзвичайно ефективним інструментом управління проектами, особливо в умовах динамічних змін. Завдяки своїй гнучкості та орієнтації на результат, цей підхід став основою роботи багатьох успішних компаній у світі. Гнучкі методи розробки (Agile) набули широкої популярності в програмних організаціях з початку 2000-х років. Їхні принципи та практики спочатку були розроблені для малих мобільних команд (до 10 чоловік в команді, яка працює над розробкою одного продукту). Однак для того, щоб ефективно використовувати потенційні переваги Agile у великих підприємствах, виникла потреба в масштабуванні цих практик.

Масштабні трансформації на основі гнучких підходів стали актуальною темою досліджень, зокрема через необхідність додаткових координаційних механізмів та інтеграції з нефункціональними підрозділами організацій, такими як фінансовий департамент та відділ управління персоналом [10, С. 85].

Для підтримки процесу масштабування були розроблені спеціалізовані фреймворки, зокрема Scaled Agile Framework (SAFe), Large Scale Scrum (LeSS) та Disciplined Agile Delivery (DAD).

Найбільш поширеним серед масштабованих систем керування вважається Scaled Agile Framework (SAFe), розроблений Діном Леффінгвеллом, який був створений для впровадження гнучких практик у великих підприємствах. Цей підхід об'єднує елементи Scrum, Extreme Programming, Kanban та Lean та функціонує на чотирьох рівнях: командному, програмному,

портфельному та рівні потоків створення цінності [11].

На командному рівні діють автономні agile-команди. Для координації численних команд на програмному рівні вводиться поняття *Agile Release Trains (ART)* – структурованих об'єднань команд, що працюють спільно. ART працюють у спеціальних ітераціях HIP (Hardening, Innovation, Planning), результатом яких є потенційно постачальні інкременти (PSI) або програмні інкременти (PI). Планування PI здійснюється під час сесій реліз-планування.

Основні цінності SAFe включають:

- вбудовану якість (built-in quality),
- прозорість (transparency),
- узгодженість (alignment),
- ефективність виконання програм (program execution) [11].

Завдяки чітко прописаним ролям, артефактам та процесам, SAFe забезпечує високу передбачуваність та контроль над проектами, що полегшує управління на всіх етапах розробки. SAFe сприяє тому, щоб проекти завжди відповідали стратегічним цілям організації. Постійна перевірка бізнес-кейсу гарантує економічну доцільність проекту, що дозволяє організаціям орієнтуватися на реальну бізнес-цінність. Завдяки системному підходу до планування та вбудованим механізмам контролю, SAFe дозволяє організаціям ефективно знижувати ризики на всіх етапах проекту, що підвищує ймовірність успішного завершення [12].

Незважаючи на орієнтацію на гнучкість, SAFe може стати надмірно формалізованим через велику кількість процесів та ролей, що може суперечити основним принципам гнучкості та швидкої адаптації.

Чітко визначені PI-ітерації та процеси можуть обмежити автономію команд, особливо якщо вони звикли до більш гнучких та адаптивних підходів, таких як Scrum. Також ефективність SAFe значною мірою залежить від здатності організації забезпечити правильну синхронізацію між усіма рівнями [11].

Таким чином, SAFe є потужним фреймворком для масштабування Agile в великих організаціях, але він потребує значних зусиль для впровадження та може бути складним для малих або середніх команд через свою складну структуру.

Сучасні дослідження дедалі частіше зосереджуються на порівняльному аналізі класичних (Waterfall) та гнучких (Agile) підходів до управління проектами в сфері розробки програмного забезпечення. Зокрема, роботи авторів Бекера та Салама підкреслюють переваги гнучких методологій в умовах високої невизначеності, характерної для ІТ-проектів. Зазначається, що застосування Agile-методів підвищує адаптивність команди, скорочує час розробки та сприяє більш ефективній взаємодії із замовником [12].

Каскадні та гнучкі моделі мають принципово різні підходи до планування, управління проектами та взаємодії з клієнтами. Нижче представлена таблиця 1, яка порівнює ці дві моделі за ключовими аспектами.

Таблиця 1

Порівняльна характеристика каскадної та гнучкої моделей розробки програмного забезпечення

Аспект	Каскадні моделі (Waterfall)	Гнучкі моделі (Agile)
Основні цілі	Стабільність, висока надійність	Швидке надання цінності, реагування на зміни
Фундаментальна гіпотеза	Програмне забезпечення розробляється на основі детального та розширеного планування, яке передбачає заздалегідь запланований результат	Розробка ґрунтується на швидкому зворотному зв'язку та змінах, здійснюється невеликою командою з постійним удосконаленням дизайну та тестування
Вимоги користувачів	Деталізуються та визначаються до початку впровадження	Формуються ітеративно в процесі розробки
Метод комунікації	Формальний	Неформальний
Взаємодія з клієнтами	Присутні на місці, акцент на пріоритетність інкрементів	Потрібна постійна взаємодія з клієнтом до завершення контракту
Контроль якості	Складно планується, пізні тестування, контроль якості є важким	Тестування проводиться під час розробки, проєктування або зміни вимог
Вартість повторної розробки	Висока	Низька
Напрямок розвитку	Фіксований	Гнучкий
Розмір проєкту	Великі проєкти	Невеликі, середні проєкти
Тестування	В кінці етапу кодування	На кожному етапі

Джерело: побудовано автором

Висновки. Таким чином, гнучкі та каскадні моделі розробки програмного забезпечення відображають два різні підходи до управління проєктами. Кожна з них має як переваги, так і обмеження, що визначають сферу їх доцільного застосування. Agile підходи (Scrum, SAgile, тощо) базуються на ітеративному процесі, що забезпечує гнучкість у плануванні та виконанні завдань. Завдяки частому зворотному зв'язку з клієнтом та поетапній реалізації функціоналу, команди можуть швидко реагувати на зміну вимог, зменшуючи ризики невідповідності кінцевого продукту очікуванням замовника. Однак надмірна кількість змін без достатньої оцінки, обмежена доступність клієнта чи розподіленість команд можуть ускладнювати координацію та впливати на стабільність проєкту.

Каскадні моделі (Waterfall, PRINCE2), навпаки, передбачають чітку послідовність етапів та високу структурованість процесу, що знижує ризики на ранніх стадіях та забезпечує стабільність реалізації. Вони є зручними для проєктів з фіксованими вимогами та обмеженими ресурсами. Водночас, основним недоліком є низька гнучкість: внесення змін на пізніх етапах практично неможливе, а відкладене тестування може призвести до пізнього виявлення критичних помилок.

Таким чином, вибір між Agile та Waterfall залежить від масштабу проєкту, стабільності вимог та необхідного рівня адаптивності. У деяких випадках доцільним є використання гібридного підходу, який поєднує гнучкість Agile зі структурністю каскадної моделі.

Список використаних джерел:

1. Annual intra- and extra-EU exports of services from 2010 to 2022. URL: <https://www.statista.com/statistics/1387135/eu-international-trade-intra-extra-services-exports/>.
2. Project Management Institute. URL: <https://www.pmi.org/>.
3. Управління проєктами: навч. Посібник. ред. О.В Ульянченко, П.Ф. Цигігал. Харків: ХНАУ ім. В.В. Докучаєва, 2010. 522 с
4. Динаміка експорту послуг за видами. 2025. Національний Банк України. URL: https://bank.gov.ua/files/ES/Trade_y.pdf.
5. A Guide to the Project Management Body of Knowledge (PMBOK Guide). PA: Project Management Institute, 2022. 274 p.
6. Hareesh Kumar Rapolu. (2024). Comparative Analysis of Agile and Waterfall Methodologies in Modern Software Development. International Journal of Social Science Exceptional Research, No. 3(1). Pp. 172-176 URL: <https://www.allsocialsciencejournal.com/search?q=SER-2025-1-051&search=search>.
7. PRINCE 2. URL: <https://www.peoplecert.org/browse-certifications/project-programme-and-portfolio-management/PRINCE2->.

8. Manifesto for Agile Software Development. URL: <http://agilemanifesto.org>.
9. Гвоздь, М.Я., Злидник, Ю.О. (2018). Agile – нова методологія менеджменту: теоретичні аспекти. Інфраструктура ринку, Вип. 25. С. 230–235.
10. Крижановський Є.М. Ящолт А.Р., Жуков С.О., Козачко О.М. (2018). Моделювання бізнес (процесів та управління ІТ проєктами). Вінниця: ВНТУ, 91 с.
11. Scale Agile: Achieving Business Agility and Agile Planning. URL: <https://scaledagile.com/>.
12. Sutherland, J. (2001). Agile Can Scale: Inventing and Reinventing SCRUM in Five Companies. Cutter IT Journal, No. 14(12). Pp. 5–11.

References:

1. Annual intra- and extra-EU exports of services from 2010 to 2022. Retrieved from: <https://www.statista.com/statistics/1387135/eu-international-trade-intra-extra-services-exports/>.
2. Project Management Institute. Retrieved from: <https://www.pmi.org/>.
3. Ulianchenko, O.V., & Tsyhikalo, P.F. (2010). Upravlinnia proiektamy: navch. Posibny [Project Management: Training Manual]. Xarkiv: KhNAU im. V.V. Dokuchaieva. [in Ukrainian].
4. Dynamika eksportu posluh za vydamy [Dynamics of services exports by type]. (2025). National Bank of Ukraine. Retrieved from: https://bank.gov.ua/files/ES/Trade_y.pdf. [in Ukrainian].
5. A Guide to the Project Management Body of Knowledge (PMBOK Guide). PA: Project Management Institute, 2022. 274 p.
6. Hareesh Kumar Rapolu. (2024) Comparative Analysis of Agile and Waterfall Methodologies in Modern Software Development. International Journal of Social Science Exceptional Research, No. 3(1). Pp. 172-176 Retrieved from: <https://www.allsocialsciencejournal.com/search?q=SER-2025-1-051&search=search>.
7. PRINCE 2. Retrieved from: <https://www.peoplecert.org/browse-certifications/project-programme-and-portfolio-management/PRINCE2->.
8. Manifesto for Agile Software Development. Retrieved from: <http://agilemanifesto.org>.
9. Hvoz, M.Ia. & Zlydnyk, Yu.O. (2018). Agile – a new management methodology: theoretical aspect [Agile – nova metodolohiia menedzhmentu: teoretychni aspekty] Market infrastructure, Vol. 25. Pp. 230–235.
10. Kryzhanovskyi Ye.M. Yashcholt A. R., Zhukov S. O., & Kozachko O. M. (2018). Modeliuvannia biznes-protsesiv ta upravlinnia IT proiektamy [Business modeling (processes and IT project management)]. Vinnytsia: VNTU. [in Ukrainian].
11. Scale Agile: Achieving Business Agility and Agile Planning. Retrieved from: <https://scaledagile.com/>.
12. Sutherland, J. (2001) Agile Can Scale: Inventing and Reinventing SCRUM in Five Companies. Cutter IT Journal. No. 14(12). Pp. 5–11.