

#### **4.12 Методологічні підходи управління проєктами**

На сьогодні проєктний менеджмент є одним з найбільш ефективних засобів адаптації підприємств та організацій до швидких змін навколишнього середовища, своєрідною філософією управління, яка змінює акцент на реалізацію більш диференційованих короткотермінових завдань, що забезпечують максимізацію використання наявних у розпорядженні виробничих і людських ресурсів, мінімізацію фінансових витрат. Все більше організацій є проєктно-керованими і світовий досвід підтверджує, що саме проєктне управління дозволило багатьом країнам вирішувати найскладніші задачі перехідного характеру в умовах жорстко обмежених фінансових ресурсів та часу.

Відповідно до Project Management Body of Knowledge Guide (PMBok Guide), управління проєктами (Project Management) це «застосування знань, навичок, інструментів і методів для задоволення або перевищення вимог проєкту» [320]. Проєктний менеджмент надає стандартні вказівки для забезпечення того, щоб проєкти були завершені вчасно, в межах бюджету та проводилися керовано та послідовно, використовуючи специфічні підходи до управління кожним аспектом проєкту у вигляді загальних і галузевих процедур, правил й положень, що встановлюють стандарти для забезпечення якості та контролю [321, 322].

Методологія управління проєктами (Project Management Methodology) є «системою практик, прийомів, процедур і правил, які використовуються тими, хто працює в дисципліні» [320]; це «основа, яка допомагає керувати та організовувати проєкт таким чином, щоб він працював найкращим чином» [323]; це «чітко визначена та науково доведена комбінація логічно пов'язаних практик, методів і процесів, які визначають, як найкраще планувати, розробляти, контролювати, виконувати та доводити проєкт до успішного завершення» [325]. Тобто, це стандартизована система управлінських дій щодо реалізації проєктів та управління проєктною командою.

У роботі «Classifying & Defining Project Management Methodologies» зазначено, що наразі існує плутанина в науковій літературі та з боку практиків проєктного менеджменту щодо того, що є методологією. Як наслідок, авторами було проведено дослідження, за результатами якого було визначено, що методологія управління проєктами це «повний набір найкращих практик, інструментів і методів, які є динамічними, гнучкими, адаптивними та адаптованими до різних проєктів в певному середовищі» [322]. Це в свою чергу породжує різноманіття методологічних підходів та методів управління проєктами.

Вибір методологічного підходу визначає як працюватиме та взаємодіятиме команда: враховує особливості проєкту, встановлює принципи управління, командної роботи, форму контролю, перевірки та оцінки результату. Від методології управління залежить якість планування та виконання проєкту. Обраний методологічний підхід управління проєктами повинен надати команді набір процесів, які можна масштабувати або замінювати відповідно до кожного проєкту, щоб допомогти керувати проєктами протягом усього їхнього життєвого циклу. Використовуючи методологічні підходи управління проєктами, команди проєкту чітко розумітимуть пріоритети, методи і сферу своєї роботи, а окремі учасники - місце та роль у команді, особистий внесок у проєкт в цілому.

Відповідно до досліджень, проведених науковцями Університету Ноттінгема, методологічні підходи управління проєктами класифіковано за рівнями [322]:

L1 – Найкраща практика, стандарти та рекомендації (Project Management Institutes Body of Knowledge (PMBOK) [320], the Association of Project Managers Body of Knowledge [335] and Project in Controlled Environment Version 2 (PRINCE2) [328]);

L2 – Секторальна методологія (виробничий сектор – класичні підходи, зазначені у L1; IT сфера - методології розробки програмного забезпечення (наприклад, Agile, SCRUM, RAD));

L3 – Специфічна індивідуальна методика організації (Microsoft розробив та використовує у своїй діяльності методику Microsoft Solution Framework (MSF); IBM - Rational Unified Process (RUP); Ericsson – PROPS тощо).

L4 – Методологія конкретного проєкту.

L5 – Індивідуальна методологія.

В свою чергу McConnell роботі «Project Management Methodology: Definition, Types, Examples» зазначає, що усі види методологій управління проєктами можна розподілити за двома підходами: традиційна (каскадна, водоспадна) - Traditional approach; гнучка (адаптивна) методологія – Modern Approach [325]. У вітчизняній науковій літературі зазначається, що усі методи управління проєктами можна розподілити на жорсткі та гнучкі. Жорсткі методи застосовуються в умовах суворого формального управління проєктом, зі зниженою довірою й підвищеною відповідальністю замовника і виконавця. Гнучкі ж методи управління проєктами базуються на поділі управлінської праці, що носить горизонтальний характер: розподіл конкретних керівників на чолі окремих підрозділів [324].

Project Management Institute (PMI) у своїх дослідженнях [333] виділяє 3 типи:

- Waterfall /каскадний підхід – класичне проєктне управління;
- Agile / гнучкий підхід до проєктного управління;
- Hybrid / гібридний підхід.

Таких класифікаційних ознак будемо дотримуватись і ми цій роботі. Визначимо та згрупуємо методи управління проєктами відповідно до каскадного, гнучкого та гібридного підходів, розглянемо особливості їх використання.

Waterfall (модель водоспаду, каскадна модель) – це послідовний процес розробки проєкту, в якому прогрес розглядається як плавне низхідне проходження (як водоспад) через фази збору вимог, аналізу, проєктування, розробки, модульного тестування, тестування інтеграції, тестування приймання та подальшого випуску. Тобто Waterfall полягає в розбивці завдань на

підзавдання або фіксовані етапи та передбачає їх виконання у суворому порядку. Модель розвитку водоспаду виникла і досі застосовується в тих галузях промисловості, в яких постфактум зміни є надмірно дорогими, якщо взагалі неможливими.

Основою каскадного підходу можна визначити РМВoК – звіт знань з управління проєктами – стандарти, конвенції, процеси, передового досвіду, термінологія та рекомендації, які прийняті як галузеві стандарти управління проєктами. Слід зазначити, що це більше не методологічний підхід, а основоположний стандарт управління проєктами, що базується на процесному підході і в якому повністю визначено взаємодію між усіма процесами, що включені у сферу знань управління проєктами. РМВoК, є єдиним стандартом у галузі проєктного менеджменту, який відповідає ISO 9001. Незважаючи на це, структуру РМВoК від РМІ часто вважають традиційною методологією. Рекомендації, наведені у РМВoК, ґрунтуються на досвіді, найкращих практиках управління проєктами та визначають головні процеси, яких керівник повинен дотримуватися, щоб надати високоякісний продукт. Сфери знань охоплюють: управління інтеграцією; управління обсягом; керування розкладом; управління витратами; управління якістю; управління людськими ресурсами; управління комунікаціями; управління ризиками; управління закупівлями; управління зацікавленими сторонами. Процеси управління проєктами, відповідно до РМВoК, згруповані за стадіями: ініціація, планування, розробка, реалізація, моніторинг – після чого йде завершення проєкту.

До процесно-орієнтованих стандартів управління проєктами також відносять і PRINCE2 – це метод управління проєктами, розроблений агентством ССТА (Central Computer and Telecommunications Agency) в 1989 р. як урядовий стандарт Великобританії для управління проєктами в інформаційних технологіях [326]. Метод PRINCE 2 є також процесно-орієнтованим з фокусом на продукт (product-based) і основну увагу приділяє організації та контролю усього проєкту, від початку до кінця. Це означає, що проєкти ретельно сплановані перед початком, кожен етап процесу чітко структурований. Його

відмінною рисою є роль ради проєкту, яка відповідає за контроль якості. Рада проєкту – це комісія, до складу якої входять клієнт, основний користувач і головний спеціаліст, який на кожному етапі перевіряє проєкт й за результатами приймає рішення. PRINCE2 концентрується на управлінських сторонах проєкту, виражених в семи принципах, семи процесах і семи темах проєкту [324, 328]:

- 7 принципів визначають загальні правила управління проєктами відповідно до PRINCE2, визначають базу методології (проєкти повинні мати бізнес-обґрунтування; команди повинні вчитися на кожному етапі; чітко визначені ролі та відповідальність; робота планується поетапно; правління проєкту «керують за винятком»; команди постійно зосереджуються на якості; підхід індивідуальний для кожного проєкту);

- 7 процесів визначають кроки просування по проєктному циклу;

- 7 тем – аспекти, за якими проводиться контроль для досягнення успіху проєкту (бізнес-кейс, організація, якість, плани, ризики, зміни, прогрес).

PRINCE2 може бути адаптований для проєктів будь-якого масштабу і будь-якої предметної області. Стандарт пропонує конкретні рекомендації щодо зміни життєвого циклу проєкту, рольової моделі і набору обов'язкових документів відповідно до потреб проєкту [328].

Project and Program Management for Enterprise Innovation (P2M) – проєктний та програмний менеджмент для інновацій підприємства – розроблений в Японії стандарт управління програмами і проєктами в рамках організації з використанням інноваційних технологій на рівні підприємств у нестабільному середовищі. Відмінність цієї каскадної методології полягає в орієнтованості не на продукт, а на покращання організації у результаті виконання проєктів [326]. P2M проєкт – це зобов'язання створити цінність, що базується на місії проєкту, яка має бути завершена в певний період у рамках узгоджених часу, ресурсів та умов експлуатації. Рамкова методологія P2M базується на «трилемі»: складність, цінність та опір (Complexity, Value and Resistance), що становлять так званий «залізний» трикутник контекстних обмежень, в рамках яких здійснюється інноваційна діяльність.

Якщо, все ж таки звертатися до методів, то до каскадного проєктного управління відносять такі:

*Critical Path Method (CPM)* – це метод управління проєктами з чітко охарактеризованими залежностями між видами діяльності. Метод містить перелік заходів і використовує структуру розбивки робіт (WBS), графік виконання, а також залежності, етапи та результати. Він окреслює критичні та некритичні дії, обчислюючи «найдовший» (на критичному шляху) і «найкоротший» (плаваючий) час для виконання завдань, щоб визначити, які види діяльності є критичними, а які ні. CPM ідеально підходить для планування, розуміння ризику розкладу та моніторингу проєкту.

*Critical Chain Project Management (CCPM)* є альтернативою моделі CPM. Якщо CPM зосереджується на термінах, CCPM робить акцент на ресурсах (людські ресурси, обладнання, обладнання тощо). Принципами CCPM є: відсутність багатозадачності; концентрація на критичних завданнях; час на створення проєкту розраховується виходячи з 100% завантаженості спеціаліста; вбудовування буферів у послідовності дій, щоб зменшити ризик перевиконання графіка.

*Event Chain Methodology (ECM)* є ще більш складною розробкою CPM – метод мережевого аналізу, що зосереджений на ідентифікації та управлінні подіями та взаємозв'язками між ними (ланцюжками подій), які впливають на розклад проєкту. Це метод моделювання графіка невизначеності. Метод ланцюга подій є розширенням кількісного аналізу ризиків проєкту за допомогою моделювання Монте-Карло. Методологія ланцюга подій намагається пом'якшити вплив мотиваційних і когнітивних упереджень при оцінці та плануванні. Це покращує точність оцінки ризику та допомагає сформувати більш реалістичні графіки проєкту з коригуванням ризиків [336].

*Six Sigma* – процесно-орієнтований підхід до управління проєктами зосереджений на кращому розумінні вимог клієнтів, покращенні бізнес-систем та фінансових показників організації. Це досягається шляхом аналізу основних процесів, а також зменшення або усунення дефектів і відходів. Сенс цього проєктного управління у тому, що постійно підвищується якість продукту,

завдяки безперервному процесу аналізу та поліпшення всіх аспектів. Для цього виділяють кілька сигм [329]:

- Визначення: цілі та обсяг проєкту визначені. Збирається відповідна інформація про процес і клієнта.
- Вимірювання: збираються дані про поточну ситуацію та показники процесу.
- Аналіз: зібрані дані аналізуються, щоб знайти першопричину(и) проблеми.
- Розробка: розроблено та впроваджено рішення проблеми.
- Контроль: реалізовані рішення оцінюються та впроваджуються механізми для утримання вигравів, які можуть включати стандартизацію.

Цю методологію часто називають ініціалами: DMAIC (Define, Measure, Analyze, Improve, Control)

*Projects integrating Sustainable Methods (PRiSM)* – метод управління проєктом, який поєднує принципи сталого розвитку із загальноприйнятою практикою управління проєктами і зосереджується на впливі проєкту на навколишнє середовище. PRiSM розглядає життєвий цикл, який виходить за межі введення проєкту в експлуатацію та виведення з експлуатації. Мета полягає в тому, щоб зосередити визначення, проєктування, планування та впровадження на стійкість та мінімізацію впливу на навколишнє середовище. Методологія PRiSM, розроблена компанією GPM Global, використовується в основному для масштабних будівельних проєктів, таких як забудова нерухомості, де несприятливий вплив на навколишнє середовище становить дуже реальну небезпеку.

*Extreme Project Management (XPM)* є перехідним методом між каскадним і адаптивним управлінням проєктами й спрямований на усунення недоліків традиційних методологій управління проєктами в їх жорсткості, але зберігає принципи планування та бюджетування. Згідно XPM, з метою забезпечення гнучкості, план, бюджет, ресурси та послідовність випуску можуть змінюватися протягом усього проєкту, щоб врахувати зовнішні зміни та переваги

користувачів. Тобто XPM це традиційне управління проектами з переміщенням до центру уваги керування змінами.

Таким чином, основні переваги використання каскадного підходу полягають у тому, що у цьому випадку передбачено дотримання інструкцій та правил по всьому процесу, завдяки цьому є чітка визначеність у термінах, бюджеті, якісних показниках тощо. Методологія каскадної моделі (класичне проектне управління) найкраще підходить у випадках, коли чітко відомо, який продукт необхідно отримати у результаті; є багато часу та ресурсів на проект; потрібна детальна документація щодо всіх процесів розробки; створення продукту проекту будується на строгій послідовності етапів. Окрім цього, великою перевагою методології Waterfall – це фіксована вартість (fixed cost). Виконавець одразу розуміє собівартість проекту, трудовитрати, які очікуються і терміни його виконання – всі етапи чітко визначені. Замовник ініціює проект, розуміючи і терміни, і вартість. Однак цей підхід має багато недоліків, а саме: складна адаптація до змін вимог клієнтів; витрачання часу на виконання регламентованих функцій, забезпечення процесів кожної стадії розробки проекту; відсутність зворотного зв'язку до закінчення реалізації проекту тощо.

Agile – сімейство гнучких ітеративно-інкрементальних методів управління проектами, об'єднаних цінностями та принципами Agile-маніфесту. Відповідно до цього підходу проект розбивається не на послідовні фази, а на маленькі підпроекти, які потім об'єднуються в готовий продукт [337]. У цьому випадку необов'язково чекати завершення одного завдання, щоб розпочати робити інше. Тобто використовується більш гнучкий метод. Управління проектами по Agile передбачає формалізацію побажань замовника в Product Backlog. При цьому процес управління вимогами в Agile-проектах можна умовно розділити на дві фази: збір та формалізація вимог і проведення циклу ітерацій (спринтів). Ініціація і верхньорівневе планування проводяться для всього проекту, а етапи (розроблення, тестування та інші) проводяться для кожного міні-проекту окремо. Це дає змогу передавати результати цих міні-проектів (інкремент) швидше, а



приступаючи до нового підпроєкту (ітерацій), в нього можна ввести зміни без великих витрат і впливу на інші частини проєкту [324].

Agile став основою для цілого ряду гнучких методів, серед яких найбільш відомі: Scrum, Kanban, Scrumban, Lean, Extreme Programming (XP), Agile Unified Process (AUP), Rapid Application Development (RAD), Dynamic Systems Development Method (DSDM), Disciplined Agile Delivery (DAD), Feature-Driven Development (FDD), Test-Driven Development (TDD), Scaled Agile Framework (SAFe). Розглянемо особливості використання кожного з вище перелічених:

*Scrum.* Життєвий цикл проєкту по Scrum – це набір ітерацій, кожна з яких включає міні-версію розробки проєкту за методом Waterfall. Над кожним проєктом працює універсальна команда фахівців, до якої приєднується ще двоє людей: власник продукту та scrum-майстер. Перший з'єднує команду з замовником та стежить за розвитком проєкту; це не формальний керівник команди, а скоріше куратор. Другий допомагає першому організувати бізнес-процес, проводить загальні збори. Відповідно до принципів Agile, Scrum розбиває проєкт на частини (project backlog), які відразу можуть бути використані замовником для отримання цінності (product backlog). І потім ці частини пріоритезуються власником продукту – представником замовника в команді. Після завершення першого блоку підпроєктів підприємець уже може використати продукт, який нехай і частково працює. Потім виконуються інші завдання, що доповнюють продукт і дають можливість повноцінно функціонувати.

Scrum як метод ґрунтується на таких принципах: люди (члени команди) важливіші за процеси; продукти, що дійсно працюють важливіші за документування їхніх цілей; співпраця з клієнтами важливіша за переговори з ними; реакція на зміни зовнішнього середовища важливіша за дотримання плану (етапів). Таким чином, кінцевим результатом застосування Scrum є висока продуктивність.

*Lean.* Цей метод управління проєктами дуже схожий на попередній, але є більш удосконаленим варіантом. Він додає до принципів Agile схему потоку

операцій (workflow) для того, щоб кожна з ітерацій виконувалася однаково якісно. У Lean, так само, як і в Scrum, робота розбивається на невеликі підпроекти, що реалізуються окремо і незалежно. Але в Lean для розроблення кожного підпроекту є потік операцій з етапами. Як і в класичному проектному менеджменті, це можуть бути етапи планування, розроблення, виробництва, тестування і поставки або будь-які інші необхідні для якісної реалізації проектів етапи. Завдяки тому, що всі підпроекти проходять контрольні етапи, проектний менеджер може бути впевненим, що все буде реалізовано якісно.

*Kanban.* Канбан є найменш суворим методом проектного управління, завдяки цьому він набув великої популярності у багатьох підприємств. Для роботи з Kanban необхідно визначити етапи потоку операцій (workflow). У Kanban вони зображуються як стовпці, а завдання позначають спеціальні картки. Картка переміщається по етапах подібно до деталі на заводі, що переходить від верстата до верстата, і на кожному етапі відсоток завершення стає вищим[324]. Kanban дозволяє залишити завдання незакінченим на якомусь етапі, якщо з'явилося більш пріоритетне завдання. На виході – готовий до постачання замовнику елемент продукту.

*Scrumban* – це буквально поєднання Scrum і Kanban. Завдяки методу Scrum команди працювали в кожному спринті з відставанням, тоді як Scrumban має функцію «витягування», що дозволяє командам працювати виключно на основі своїх можливостей, подібно до методології Kanban [323].

*Extreme Programming (XP)* – це гнучкий метод управління проектами, що використовується при розробці програмного забезпечення. XP охоплює стандартні принципи Agile, такі як короткі ітерації; вимоги нечітко визначені за допомогою історій користувачів, поки вони не знадобляться; негайне та регулярне тестування; тісне залучення зацікавлених сторін на місці за допомогою швидкого зворотного зв'язку; і відповідальність команди за успіх чи невдачу. Це схоже на Scrum, тільки XP є набагато більш директивним. На відміну від Scrum, який в першу чергу займається організацією команди, XP в першу чергу займається кодом: його простотою, тим, як він написаний і розгорнутий,

як працюють розробники тощо. Це означає, що можливо застосовувати XP, лише для компанії, яка розробляє програмне забезпечення.

*Agile Unified Process (AUP)* – це метод управління проєктами, що об'єднує Rational Unified Process (RUP) з Agile методами (AM). AUP – це ітераційно-інкрементний процес, що складається з робочих процесів і фаз. Пряме порівняння між RUP і AUP показує, що AUP поєднує бізнес-моделювання, вимоги, аналіз і проєктування в єдиний робочий процес. Робочий процес AUP складається з моделювання, впровадження, тестування, розгортання, керування конфігурацією, керування проєктами та середовища. Фази AUP складаються з початку, розробки, будівництва та переходу [330].

*Rapid Application Development (RAD)* – метод управління проєктами, популярний в розробці програмного забезпечення. Ключовою перевагою підходу RAD є швидке виконання проєкту завдяки зосередженню на мінімізації етапу планування та максимальному розробці прототипу. Зменшуючи час планування та наголошуючи на ітераціях прототипів, RAD дозволяє керівникам проєктів і зацікавленим сторонам точно вимірювати прогрес, спілкуватися в режимі реального часу щодо проблем або змін, що розвиваються. Це сприяє підвищенню ефективності, швидшому розвитку та ефективному спілкуванню.

*Dynamic Systems Development Method (DSDM)* – це гнучка структура реалізації проєкту, яка спочатку використовувалася як метод розробки програмного забезпечення. DSDM Agile Project Framework – це ітераційний і поетапний метод, який охоплює принципи Agile-розробки, включаючи безперервну участь користувачів/замовників, фіксує вартість, якість і час, використовує пріоритети MoSCoW щодо обсягу робіт, щоб налаштувати результат проєкту відповідно до встановленого обмеження часу.

Вперше випущений у 1994 році, DSDM спочатку намагався забезпечити певну дисципліну методу швидкої розробки додатків (RAD). У пізніших версіях DSDM Agile Project Framework був переглянутий і став загальним підходом до управління проєктами та прийняття рішень і може використовуватися для проєктів, які не є IT. Agile Project Framework DSDM охоплює широкий спектр

заходів протягом усього життєвого циклу проєкту та передбачає управлінські заходи, які відрізняють його від деяких інших методів Agile [331].

*Disciplined Agile Delivery (DAD)* – це гнучкий метод розробки IT-рішень, орієнтований на людей, насамперед, на навчання. DAD розглядає всі аспекти повного життєвого циклу доставки, підтримуючи різні способи роботи (WoW), які можна адаптувати до контексту. DAD охоплює всі аспекти гнучкої розробки програмного забезпечення надійним, прагматичним і керованим чином [332].

*Test-Driven Development (TDD)* – метод управління проєктами розробки програмного забезпечення, що передбачає повторення короткого циклу розробки: спочатку розробник пише (спочатку невдалий) автоматизований тестовий приклад, який визначає бажане покращення або нову функцію, потім створює мінімальну кількість коду, щоб пройти цей тест, і реорганізує новий код до прийнятних стандартів.

*Scaled Agile Framework (SAFe)* – метод, що виник з метою адаптації та Agile методології для реалізації великих проєктів і передбачає наявність кількох команд, які працюють або над різними модулями більшої інтегрованої системи (продукт), або кількома різними проєктами, що мають загальний набір бізнес-цілей (програма). Далі відбувається координування розробки та випуску кількох окремих продуктів (або програм), що мають взаємозалежність, спільні ресурси, залежність від підвищеного ризику та пов'язане управління для прийняття рішень, або просто загальну організаційну структуру звітності [327].

Перевагами гнучких підходів є ранній зворотний зв'язок, безперервна інтеграція, безперебійне спілкування та міждисциплінарні проєктні команди. Тому, методологія Agile є найкращим підходом, якщо: є велика ймовірність змін при реалізації проєкту; відсутнє чітке уявлення щодо мети проєкту; умови реалізації проєкту вимагають швидкого реагування на зміни та можливості, що важливіше, ніж ідеальні результати; зацікавлені сторони потребують (або хочуть) бути залученими на кожному етапі. У випадку, якщо необхідно реалізувати великий проєкт, який потребує суворої регламентації та документації дій (наприклад, якщо залучаються сторонні організації під час

проєкту), є потреба у передбачуваному результаті і чіткому уявленні результатів на початку, механізм управління змінами ускладнений, є суворі терміни або результати, яких необхідно дотримуватися, слід скористатися каскадним підходом, або обрати гібридну модель.

Hybrid - гібридний підхід являє собою поєднання методологій Waterfall та Agile, де фази проєкту плануються з використанням підходу водоспаду, а виконання фаз проєкту (фактичних завдань) виконується методом Agile [327]. Він включає найкращі частини цих двох методологій та утворює гнучкий і структурований підхід, що можна використовувати в великих проєктах. Поєднуючи атрибути Waterfall та Agile, гібридний метод (який іноді називають «Структурований Agile») дає найкраще з обох методологій і фокусується на зборі та аналізі вимог, дає можливість швидких ітерацій.

Існує багато різних гібридних методологій, які намагаються застосувати механізми зворотного зв'язку до традиційної моделі водоспаду, щоб технічні та функціональні недоліки оригінального дизайну, які були виявлені під час розробки, можна було швидше врахувати. Деякі з найбільш популярних гібридних моделей включають [327]: V-Model /V-модель; Spiral Model / Спіральна модель; Iterative and Incremental / Ітеративний і інкрементальний підходи.

*V-модель* розробки проєкту використовує модифікований водоспад, щоб забезпечити послідовну методологію розробки, яка має механізми зворотного зв'язку між фазами життєвого циклу. Замість того, щоб рухатися вниз лінійним шляхом, етапи процесу згинаються вгору після кожної фази, щоб утворити типову V-форму. V-модель демонструє взаємозв'язки між кожною фазою життєвого циклу розробки та пов'язаною з нею фазою тестування.

*Спіральна модель* – це генератор моделей процесів, орієнтованих на ризики. Ключові відмінні характеристики спіральної моделі полягають у тому, що кожна ітерація системи відповідає чотирьом ключовим фазам, які призначені для виявлення та зменшення ризиків: визначення цілей, планування обсягу проєкту; створення прототипів, експерименти та дослідження для виявлення та

вирішення потенційних ризиків (технічних, концептуальних тощо); проектування, розробка та тестування; моніторинг та зворотний зв'язок для планування наступної ітерації. Далі, на основі виявлених ризиків проекту, спіральна модель спрямовує команду працювати обираючи модель процесів (інкрементальна, каскадна або еволюційна).

Ітераційний характер моделі Spiral робить її прикладом гібридної Waterfall-Agile методології та відповідає багатьом характеристикам (прототипи, експерименти), які існують в інших новітніх чистих методологіях Agile, таких як Scrum, XP та AUP.

*Інкрементальний* підхід можна назвати підвидом Waterfall. Він полягає в тому, що великий проєкт розділяється на незалежні частини, які запускаються окремо. Використання інкрементальної розробки зменшує основні недоліки методологій водоспаду/послідовної розробки, оскільки весь проєкт розбивається на менші кроки або ітерації, що можуть застосувати досвід, отриманий з попередньої ітерації.

*Ітеративний* підхід використовується у ситуації, коли частини проєкту сильно інтегровані і неможливо реалізувати його частинами, однак потрібен швидкий запуск мінімального функціоналу з подальшим його доопрацюванням та розвитком.

Хоча ітеративна/інкрементальна розробка проєкту на перший погляд може виглядати як Agile-підхід, однак є кілька ключових відмінностей, головна з яких полягає в тому, що у цьому випадку розробка зазвичай виконується за тими ж кроками, що і водоспад, просто відбувається за менші одиниці часу, і що з Agile маємо потенційно придатний для доставки продукт завжди, тоді як це не обов'язково стосується ітеративної / інкрементальної розробки.

До різновидів ітеративно-інкрементального підходу можна віднести *Integrated Project Management (IPM)* – це методологія управління проєктами, яка найчастіше використовується в креативних індустріях. Цей цілісний стиль управління проєктами поєднує попередні вимоги до документування методу Waterfall з підходом Agile до робочого процесу, що включає впровадження з

подальшим зворотним зв'язком та ітерацією. Як і інші методології, він заохочує обмін інформацією та стандартизацію в усій організації. Інтеграція тут означає об'єднання багатьох невеликих проєктів в одну структуру для створення узгоджених кампаній. Він також спрямований на зменшення впливу функціональних і відомчих відділів в організації.

*Feature-Driven Development (FDD)* – це ітераційний і інкрементний процес розробки програмного забезпечення, який слідує принципам маніфесту Agile. Ідея полягає в тому, щоб розробити високорівневі функції, область і об'єктну модель (спочатку визначається загальна місія, обсяг і бізнес-цілі системи, що розробляється), а потім використовувати їх для планування, проєктування, розробки та тестування конкретних вимог і завдань на основі загальної функції, до якої вони належать [327].

Таким чином, кожен з вище перелічених варіантів має свої особливості, характеристики та специфічні методи (табл. 1).

**Таблиця 1.**  
Методологія управління проєктами

| Методологічний підхід, характеристика                                                                                                                       | Ключові переваги                                                                                                                                                                                                      | Недоліки                                                                                                                                                                         | Сфера використання                                                                                                                                                                                                                                         | Стандарти/Методи                         |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------|
| Waterfall – послідовний процес розробки проєкту і полягає в розбивці завдань на підзавдання / фіксовані етапи та передбачає їх виконання у суворому порядку | Детальна документація; Чіткі процеси та терміни; Наявність інструкцій та правил; Визначеність у термінах та бюджеті; Відсутність додаткових витрат на комунікацію; Зниження відхилень внаслідок ретельного планування | Тривалий запуск; Жорсткі вимоги уповільнюють внесення змін; Низька гнучкість змін у виконанні проєктних робіт; Кінцевий результат замовник отримує наприкінці реалізації проєкту | Середні та великі проєкти; Проєкти з декількома командами; Проєкти із зрозумілими цілями на початку; Проєкти впровадження рішень, що тиражуються; Проєкти, функціонування та запуск яких зрозумілі та стандартизовані; Проєкти, які вимагають «fixed cost» | PMBok                                    |
|                                                                                                                                                             |                                                                                                                                                                                                                       |                                                                                                                                                                                  |                                                                                                                                                                                                                                                            | PRINCE 2                                 |
|                                                                                                                                                             |                                                                                                                                                                                                                       |                                                                                                                                                                                  |                                                                                                                                                                                                                                                            | P2M                                      |
|                                                                                                                                                             |                                                                                                                                                                                                                       |                                                                                                                                                                                  |                                                                                                                                                                                                                                                            | Critical Path Method (CPM)               |
|                                                                                                                                                             |                                                                                                                                                                                                                       |                                                                                                                                                                                  |                                                                                                                                                                                                                                                            | Critical Chain Project Management (CCPM) |
|                                                                                                                                                             |                                                                                                                                                                                                                       |                                                                                                                                                                                  |                                                                                                                                                                                                                                                            | Six Sigma projects                       |
|                                                                                                                                                             |                                                                                                                                                                                                                       |                                                                                                                                                                                  |                                                                                                                                                                                                                                                            | PRiSM                                    |
|                                                                                                                                                             |                                                                                                                                                                                                                       |                                                                                                                                                                                  |                                                                                                                                                                                                                                                            | Extreme Project Management (XPM)         |

Продовження табл. 1

|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               |                                    |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------|
| Agile<br>- гнучкий ітеративно-інкрементальний підхід до управління проектами, об'єднаних цінностями та принципами Agile-маніфесту. Проект розбивається на маленькі підпроекти, які потім об'єднуються в готовий продукт | Гнучкість та адаптивність; Швидкий запуск, покрокове коригування шляхом критики з боку замовника; Креативність та інновації; Постійний зв'язок між замовником і розробником; Менша кількість документації; Низькі ризики та витрати; Спрямованість на покращення якості; Задоволеність проектної команди та споживачів | Умовність планів; Вимагає висококваліфікованої, орієнтованої на замовників групи розробників; Потребує постійного залучення замовника до реалізації робіт за проектом; Відсутність довгострокових деталізованих планів; Зменшення документації (шаблонів) призводить до необхідності постійного пошуку нових рішень | Проекти-стартапи, коли розробляється щось нове і невідомо, що буде отримано в кінці; Проекти з розробки власними силами «з нуля» або під час яких передбачається глибока кастомізація рішення; Проекти із відкритим бюджетом. | Scrum                              |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | Lean                               |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | Kanban                             |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | Scrumban                           |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | Extreme Programming                |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | Agile Unified Process              |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | Dynamic Systems Development Method |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | Disciplined Agile Delivery         |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | Test-Driven Development            |
| Hybrid<br>- фази проекту плануються з використанням підходу водоспаду, а виконання фаз проекту (фактичних завдань) виконується методом Agile                                                                            | Поєднання гнучкості та структурованості; Високі очікування клієнтів – персоналізація; Легкість і швидкість у великих проектах; Неможливість ввести зміни на певному етапі; Розвиток потреб клієнта; Чітко визначена відповідальність                                                                                   | Для тих, хто приходить із середовища водоспаду, недолік – відмова від певного рівня впевненості в обмін на гнучкість Agile, і навпаки, хтось із чистого Agile може вважати, що недоліком є обмеження свободи, щоб отримати деякі переваги від планування бюджету водоспаду та розкладу                              | Великі проекти з невизначеністю кінцевих цілей і складністю реалізації; Може бути сумісним з будь-якою галуззю та командою будь-якого розміру                                                                                 | Rapid Application Development      |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | Scaled Agile Framework             |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | V-Model                            |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | Spiral Model                       |
|                                                                                                                                                                                                                         |                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                               | Iterative and Incremental          |

\*Джерело: Сформовано автором на основі [322, 323, 325, 327, 334]

Сьогодні не існує універсального підходу, який би підходив для всіх типів проектів і організацій, оскільки проекти різні, їх цілі та вимоги також унікальні, виробничі системи та методи є специфічними. Ось чому методологія, яка найкраще підходить для однієї проектної команди, може стати марною тратою часу для іншої. Наприклад, багато команд розробників програмного забезпечення вважають, що традиційні підходи до управління проектами приносять більше шкоди, ніж користі. Тому, команди розробників програмного забезпечення використовують принципи Agile, а не Waterfall для задоволення



своїх конкретних вимог. І навпаки, проєкти в космічній, обробній, будівельній промисловостях, в яких по завершенні зміни є надмірно дорогими, надають перевагу класичним методам, тобто каскадному підходу. Для кожного окремого проєкту рекомендується підбирати відповідний тип керування, враховуючи складність проєкту, замовника, цілі, головних стейкхолдерів, наявні ресурси та обмеження проєкту (в тому числі, зміни та ризик), можливість змін, часові межі, інструменти та персонал (табл. 2).

Таблиця 2.

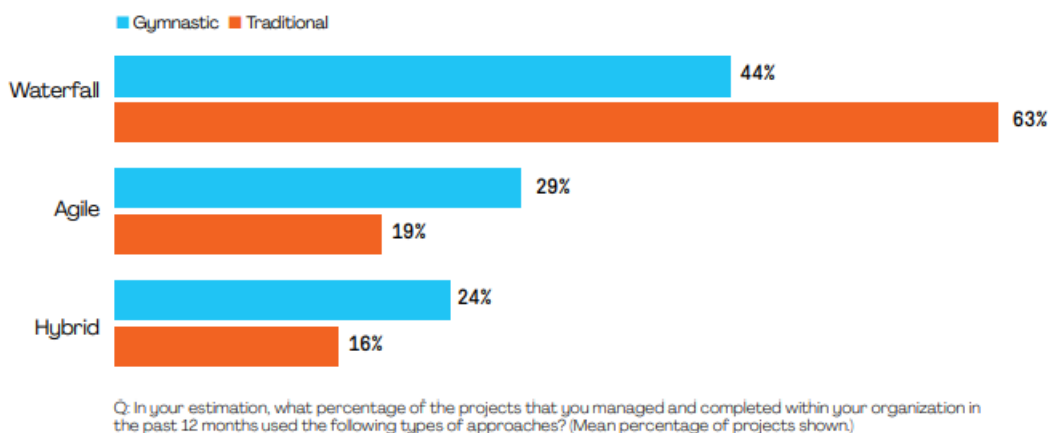
Порівняльна характеристика методологій управління проєктами за головними аспектами

| Аспекти              | Waterfall project management                      | Agile project management                                          | Hybrid project management                                          |
|----------------------|---------------------------------------------------|-------------------------------------------------------------------|--------------------------------------------------------------------|
| Репозиторій робіт    | Робоча структура проєкту (Project Work Breakdown) | Перелік задач проєкту з пріоритетами (Project Backlog)            | Перелік задач за фазами проєкту з пріоритетами (Phase Backlogs)    |
| Підхід до планування | Деталізоване планування                           | Планування ітерацій                                               | Високорівневе планування + планування ітерацій                     |
| Зобов'язання         | Раннє визначення мети проєкту                     | Узгодження мети з постійним навчанням та адаптацією через підцілі | Узгодження мети з постійним навчанням та адаптацією через підетапи |
| Фокус                | Фокус на етапах                                   | Фокус на швидких результатах                                      | Фокус на кінцевих результатах кожної стадії                        |
| Акцент               | Акцент на документації та деталях                 | Акцент на отриманій цінності                                      | Акцент на отриманій цінності                                       |
| Бюджет               | Фіксований бюджет                                 | Гнучкий бюджет                                                    | Гнучкий або фіксований бюджет                                      |
| Мета                 | Чітко встановлені цілі                            | Нечіткі або передбачувані цілі                                    | Передбачувані або чітко визначені цілі                             |
| Тривалість проєкту   | Від тривалого до середнього                       | Від короткого до середнього.                                      | Будь-яка тривалість                                                |
| Обсяг                | Фіксований обсяг                                  | Невідомий/гнучкий обсяг                                           | Фіксований або гнучкий обсяг                                       |
| Управління змінами   | Складно змінювати                                 | Найлегше змінювати (адаптований підхід)                           | Легко змінювати (адаптований підхід)                               |
| Оцінка               | Точне оцінювання                                  | Абстрактне оцінювання                                             | Абстрактне оцінювання                                              |

\*Джерело: переклад з[334]

Звертаючись до досвіду застосування методологій управління проєктами, то, відповідно до досліджень PMI, підприємства з традиційними підходами до управління (ті, які для вирішення завдань застосовували перевірені методи) більше використовують класичну методологію – Waterfall у 63%, Agile у 19%, Hybrid – 16%; гнучкі підприємства (ті, які в першу чергу використовували будь-

які нові і всі можливі методи для вирішення завдань) – гнучкий та гібридний підхід – Waterfall у 44%, Agile у 29%, Hybrid – 24% (рис. 1) [333].



**Рисунок 1.** Питома вага підходів, що використовувались при управлінні проєктами у 2020 році (Європа) [333]

Дослідження, проведені PMI (2021, Pulse of the Profession®) показали, що на сьогодні підприємства Європи, що використовують гнучкі методи управління з більшою ймовірністю, ніж традиційні підприємства, мали високу організаційну гнучкість (46% проти 21%) і повідомляли про підвищення продуктивності у 2020 році порівняно з попереднім роком (68% проти 48%); частіше, ніж підприємства з традиційним підходом до управління, прискорили зміни в бізнесі у 2020 році порівняно з попереднім роком, а саме: цифрова трансформація (76% проти 58% для традиційних підприємств), підхід до інновацій (66% проти 41%), бізнес-стратегія (63% проти 51%), організаційна адаптивність (62% проти 49%) [333]. Тобто підприємства, що використовують гнучкі методи управління мають високу організаційну гнучкість, прискорюють зміни в бізнесі і використовували більш різноманітні підходи до управління проєктами, ніж традиційні підприємства.

Таким чином, вибір правильної методики управління проєктами важливий, оскільки він визначає, як буде працювати команда над проєктом. Методологія управління проєктами створює спільну мову, загальні процеси та шаблони для організації. Тому, вирішуючи, яким методом управляти проєктом, необхідно

враховувати його складність, чіткість уявлення про кінцеві цілі, досвід проєктного менеджера, організаційну культуру, стейкхолдерів, замовника, бюджет, наявні ресурси та обмеження проєкту (в тому числі, зміни та ризик), часові межі, інструменти та розмір команди проєкту тощо. У випадку, коли планується великий проєкт, що комплексно впливає на всю архітектуру, зачіпає крос-функціональні бізнес-процеси та діючі регламенти, більш доречними до використання є Waterfall-методи. У випадку реалізації проєктів-стартапів, коли розробляється щось нове і немає чіткого уявлення того, що буде отримано як результат проєкту слід використовувати Agile-методи.