

УНІВЕРСИТЕТ «КРОК»

**Навчально-науковий інститут менеджменту та освіти дорослих
Кафедра управлінських технологій**

Кафедра управлінських технологій

Крискун Іван Миколайович

УДК 005.8

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

“Управління проектами в компанії із застосуванням
гнучких технологій”

Галузь знань: 07 Управління та адміністрування

Спеціальність: 073 Менеджмент

Освітня програма: «Менеджмент організацій»

Подається на здобуття освітнього рівня магістр

Кваліфікаційна робота містить результати власних доробок. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

І.М.Крискун

(підпис, ініціали та прізвище здобувача)

Науковий керівник (консультант)

Данченко Олена Борисівна

доктор технічних наук, професор

(прізвище, ім'я по батькові, науковий ступінь керівника)

Київ-2024

АНОТАЦІЯ

Кваліфікаційна робота присвячена розв'язанню проблеми впровадження гнучких методологій управління проектами у сфері корпоративного менеджменту. Метою роботи було всебічне дослідження теоретичних та практичних аспектів використання Agile-методологій в управлінні проектами для підвищення ефективності та результативності організацій. Для досягнення цієї мети було проаналізовано історію та принципи Agile, порівняно традиційні та гнучкі методи управління, досліджено практичне впровадження Agile в компанії ТОВ "ГлобалЛоджик Україна", а також надано рекомендації щодо покращення процесів.

Кваліфікаційна робота містить 103 сторінки, 18 рисунків, 12 таблиць, 45 джерел та 4 додатки.

Ключові слова: гнучке управління проектами, Agile, Scrum, Kanban, Lean, впровадження, ефективність, управління змінами, корпоративний менеджмент, ТОВ "ГлобалЛоджик Україна".

ABSTRACT

The qualification work is dedicated to solving the problem of implementing agile project management methodologies in the field of corporate management. The aim of the work was to comprehensively study the theoretical and practical aspects of using Agile methodologies in project management to improve the efficiency and effectiveness of organizations. To achieve this aim, the history and principles of Agile were analyzed, traditional and agile project management methods were compared, the practical implementation of Agile in the company GlobalLogic Ukraine LLC was studied, and recommendations for process improvement were provided.

The qualification work contains 103 pages, 18 figures, 12 tables, 45 sources, and 4 appendices.

Keywords: agile project management, Agile, Scrum, Kanban, Lean, implementation, efficiency, change management, corporate management, GlobalLogic Ukraine LLC.

ЗМІСТ

| | |
|--|-----|
| ВСТУП | 5 |
| РОЗДІЛ 1. РОЗУМІННЯ ГНУЧКОГО УПРАВЛІННЯ ПРОЕКТАМИ | 9 |
| 1.1 Історія та принципи методології Agile | 9 |
| 1.2 Гнучкі та традиційні підходи до управління проектами | 19 |
| 1.3 Гнучке управління проектами в ТОВ “ГлобалЛоджик Україна” | 35 |
| РОЗДІЛ 2. ВПРОВАДЖЕННЯ AGILE В ОРГАНІЗАЦІЯХ | 38 |
| 2.1 Підготовка організації до Agile | 38 |
| 2.2 Кроки для гнучкої трансформації та виклики | 49 |
| 2.3 Досвід впровадження Agile в ТОВ “ГлобалЛоджик Україна” | 52 |
| РОЗДІЛ 3. МАЙБУТНЄ AGILE ТА РЕКОМЕНДАЦІЇ ЩОДО ПОКРАЩЕННЯ ПРОЕКТУ В ТОВ “ГЛОБАЛЛОДЖИК УКРАЇНА” | 60 |
| 3.1 Нові тенденції в гнучкому управлінні проектами | 60 |
| 3.2 Аналіз Agile проекту компанії ТОВ "ГлобалЛоджик Україна" | 68 |
| 3.3 Пропозиції щодо покращення в управлінні проектом | 78 |
| ВИСНОВКИ | 94 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ | 97 |
| ДОДАТКИ | 100 |

ВСТУП

У сучасному бізнесі, що швидко розвивається, здатність адаптуватися до мінливого середовища та потреб клієнтів є не просто перевагою, але й необхідністю. Ця необхідність лежить в основі зростаючої актуальності методологій гнучкого управління проектами в корпоративному світі. Гнучкі методології, які спочатку зародилися в сфері розробки програмного забезпечення, вийшли за рамки свого походження і стали життєво важливою основою для управління проектами в різних секторах. Їх популярність у сучасній діловій практиці вимагає ретельного вивчення та розуміння, що є основною метою цієї кваліфікаційної роботи.

Об'єктом цього дослідження є методології гнучкого управління проектами, що застосовуються в сучасному бізнес-середовищі. Це включає детальне вивчення різних фреймворків Agile, таких як Scrum, Kanban та Lean, а також те, як ці методології адаптуються та застосовуються в різних галузях.

Предметом дослідження є практичне застосування Agile-методологій в компаніях. Основна увага приділяється стратегіям, викликам та перевагам, пов'язаним з впровадженням Agile-практик, а також впливу Agile-трансформації на організаційну культуру та продуктивність.

Метою цієї кваліфікаційної роботи є всебічний аналіз методологій гнучкого управління проектами, їх застосування в різних організаційних контекстах та отриманих результатів. Вона має на меті поєднати теоретичні знання з практичним досвідом, пропонуючи глибоке розуміння ролі Agile у підвищенні ефективності та результативності управління проектами в компаніях.

Методи дослідження, які будуть використані в кваліфікаційній роботі спрямовані на набуття достовірних наукових знань, умінь та практичних навичок у сфері управління проектами.

Літературний огляд та аналіз джерел - цей метод включає систематичний перегляд та аналіз наукових статей, книг, дисертацій та інших

джерел інформації, що стосуються управління проектами та гнучких технологій. Метою є визначення існуючих підходів, концепцій та моделей, а також виявлення прогалин в знаннях, які потребують подальшого дослідження. Літературний огляд допоможе встановити теоретичні основи дослідження та забезпечити контекст для подальших емпіричних досліджень.

Огляд існуючих практик - в рамках цього методу будуть розглянуті та проаналізовані існуючі практики управління проектами у різних компаніях, що використовують гнучкі технології. Це дозволить виявити найефективніші практики, а також визначити можливі проблеми та виклики, з якими стикаються компанії під час впровадження Agile методологій. Огляд практик може включати аналіз кейсів, інтерв'ю з менеджерами проектів та працівниками компаній.

Емпіричне дослідження - емпіричне дослідження включає проведення експериментів або спостережень за реальними проектами, що використовують гнучкі методи управління. Метою є оцінка ефективності впровадження Agile технологій, аналіз результатів проектів та порівняння з традиційними методами управління. Це дозволить отримати практичні висновки щодо доцільності використання гнучких підходів у конкретних умовах.

Завдання кваліфікаційної роботи є дослідження впровадження гнучких методів управління в організація, а також майбутнє та еволюція гнучких методів управління

Практичне значення цього дослідження полягає в його потенціалі для інформування та керівництва бізнесом, який планує або вже здійснює гнучкі трансформації. Висновки, отримані в результаті цього дослідження, можуть допомогти організаціям оптимізувати свої стратегії управління проектами, ефективніше адаптуватися до гнучких методологій та реалізувати весь потенціал гнучких практик у підвищенні операційної ефективності та реагування на динаміку ринку.

Ця кваліфікаційна робота намагається розкрити складність і динаміку методологій гнучкого управління проектами в умовах компанії, надаючи

цінну інформацію для науковців, практиків галузі та організацій, які прагнуть досягти досконалості в управлінні проектами за допомогою Agile-практик.

Дана кваліфікаційна робота складається з трьох основних розділів, кожен з яких має кілька підрозділів, що детально розглядають різні аспекти теми "Управління проектами в компанії із застосуванням гнучких технологій".

Розділ 1. Розуміння гнучкого управління проектами

У першому розділі наводиться загальний огляд гнучкого управління проектами. Спочатку розглядається історія та принципи методології Agile, що є основою для розуміння гнучких підходів. Далі проводиться порівняння гнучких та традиційних методів управління проектами, що дозволяє виявити їх переваги та недоліки. На завершення, аналізується використання гнучкого управління проектами в конкретній компанії - ТОВ "ГлобалЛоджик Україна".

1.1. Історія та принципи методології Agile

1.2. Гнучкі та традиційні підходи до управління проектами

1.3. Гнучке управління проектами в ТОВ "ГлобалЛоджик Україна"

Розділ 2. Впровадження Agile в організаціях

Другий розділ присвячено практичним аспектам впровадження Agile в організаціях. Він починається з опису підготовки організації до переходу на Agile, що включає оцінку готовності та необхідні підготовчі кроки. Далі розглядаються конкретні кроки для здійснення гнучкої трансформації та виклики, з якими може зіткнутися організація. Розділ завершується аналізом досвіду впровадження Agile в ТОВ "ГлобалЛоджик Україна".

2.1. Підготовка організації до Agile

2.2. Кроки для гнучкої трансформації та виклики

2.3. Досвід впровадження Agile в ТОВ "ГлобалЛоджик Україна"

Розділ 3. Майбутнє Agile та рекомендації щодо покращення проекту в ТОВ "ГлобалЛоджик Україна"

Третій розділ аналізує нові тенденції у сфері гнучкого управління проектами та перспективи їх розвитку. Також проводиться детальний аналіз поточного стану Agile проекту в компанії ТОВ "ГлобалЛоджик Україна". На

основі отриманих результатів та новітніх тенденцій, надаються рекомендації щодо покращення управління проектами в компанії.

- 3.1. Нові тенденції в гнучкому управлінні проектами
- 3.2. Аналіз Agile проекту компанії ТОВ "ГлобалЛоджік Україна"
- 3.3. Пропозиції щодо покращення в управлінні проектом

Кваліфікаційну роботу завершають Висновки

У висновках узагальнюються основні результати дослідження, робляться висновки щодо ефективності впровадження гнучких методів управління проектами та надаються рекомендації для подальших досліджень та практичного застосування результатів роботи.

РОЗДІЛ 1. РОЗУМІННЯ ГНУЧКОГО УПРАВЛІННЯ ПРОЕКТАМИ

1.1 Історія та принципи методології Agile

Методологія Agile виникла як відповідь на обмеження традиційних підходів до управління проектами, зокрема, моделі Waterfall, які часто виявлялися занадто жорсткими та повільними для швидкого та мінливого світу розробки програмного забезпечення. Витоки Agile можна простежити на початку 1990-х років, коли команди розробників програмного забезпечення почали експериментувати з більш гнучкими та ітеративними підходами.

Одним з перших впливів на Agile стала робота фреймворку Scrum, розробленого Джеффом Сазерлендом і Кеном Швабером в середині 1990-х років. Scrum запровадив концепцію ітеративної розробки, коли невеликі міжфункціональні команди працюють у коротких, обмежених у часі ітераціях, які називаються спринтами, щоб забезпечити поступове покращення продукту. Приблизно в той же час інші методології, такі як екстремальне програмування (XP), запроваджене Кентом Бекем, наголошували на таких практиках, як безперервна інтеграція, розробка, керована тестами, та часті релізи.

Історичні передумови

Зародження Agile бере свій початок на початку 1990-х років, коли індустрія розробки програмного забезпечення боролася з жорсткою та лінійною моделлю водоспаду. Критичний момент настав у 2001 році, коли 17 розробників програмного забезпечення зустрілися в Сноуберд, штат Юта, щоб обговорити альтернативні методи. Результатом цієї зустрічі стало створення Маніфесту Agile, основоположного документа, який сформулював нову філософію, зосереджену на гнучкості, швидкості та реагуванні на зміни.

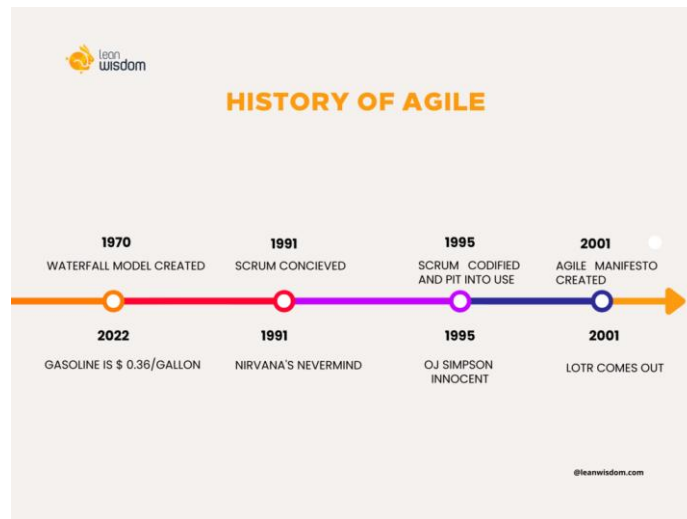


Рис. 1.1 Історія Agile методології

Джерело: [15]

Чотири основні цінності маніфесту:

1. Люди та взаємодія над процесами та інструментами
2. Робоче програмне забезпечення над вичерпною документацією
3. Співпраця з клієнтами під час переговорів щодо контрактів
4. Реагування на зміни за планом

На додаток до цих цінностей, Маніфест Agile включає дванадцять принципів, які більш детально описують, як повинні працювати Agile-команди для досягнення кращих результатів.

Принципи методології Agile

В основі методології Agile лежить набір принципів, якими команди керуються у своєму підході до розробки програмного забезпечення та управління проектами. Ці принципи наголошують на гнучкості, співпраці та задоволеності клієнтів. Ось дванадцять принципів Agile:

1. **Задоволеність клієнтів:** Найвищий пріоритет - задовольнити клієнта шляхом своєчасного і безперервного постачання цінного програмного забезпечення.
2. **Ласкаво просимо до змін:** Гнучкі процеси використовують зміни для конкурентної переваги клієнта, навіть на пізніх стадіях розробки.

3. Часті поставки: Часто надавати працююче програмне забезпечення, від кількох тижнів до кількох місяців, надаючи перевагу коротшим строкам.

4. Співпраця: Бізнесмени та розробники повинні працювати разом щодня протягом усього проекту.

5. Мотивовані люди: Створюйте проекти навколо мотивованих людей. Створіть для них необхідне середовище та підтримку і довіряйте їм у виконанні роботи.

6. Розмова віч-на-віч: Найбільш ефективним та дієвим методом передачі інформації команді розробників та всередині неї є особиста розмова.

7. Робоче програмне забезпечення: Працююче програмне забезпечення є основним показником прогресу.

8. Сталий розвиток: Гнучкі процеси сприяють сталому розвитку. Спонсори, розробники та користувачі повинні мати можливість підтримувати постійний темп протягом невизначеного часу.

9. Технічна досконалість: Постійна увага до технічної досконалості та хорошого дизайну підвищує гнучкість.

10. Простота: Простота - мистецтво максимізації обсягу невиконаної роботи - має важливе значення.

11. Самоорганізовані команди: Найкращі архітектури, вимоги та дизайни з'являються в командах, що самоорганізуються.

12. Рефлексія та коригування: Через певні проміжки часу команда розмірковує над тим, як стати більш ефективною, а потім налаштовує і коригує свою поведінку відповідно.

Ці принципи підкреслюють важливість швидкого та ефективного надання цінності клієнтам, а також сприяють створенню спільного та адаптивного середовища для команд розробників. Гнучкі методології заохочують постійний зворотній зв'язок, що дозволяє командам коригувати свої плани та пріоритети на основі реальних відгуків та мінливих вимог.

Впровадження гнучких методологій зробило революцію в індустрії розробки програмного забезпечення та за її межами, що призвело до покращення результатів проектів, підвищення рівня задоволеності клієнтів, а також до створення більш динамічних та стійких команд розробників.

Ключові фреймворки Agile

Під егідою Agile з'явилося кілька фреймворків, кожен з яких має свій підхід до реалізації принципів Agile:

Scrum: Фокусується на ітераціях фіксованої тривалості, які називаються спринтами, з регулярними переоцінками та коригуваннями. Scrum - один з найпопулярніших фреймворків Agile, відомий своїм структурованим, але гнучким підходом до управління проектами. В основі Scrum лежать ітерації фіксованої тривалості, відомі як спринти, що зазвичай тривають від двох до чотирьох тижнів. Ці спринти є основними періодами часу, протягом яких певна робота повинна бути завершена і підготовлена для перевірки.

Планування та виконання спринту

Кожен спринт починається з планової наради, на якій команда визначає роботу, яку потрібно виконати, виходячи з відставання в роботі над продуктом, пріоритетного списку завдань і вимог. Потім команда бере на себе зобов'язання щодо того, що вона може виконати протягом майбутнього спринту, створюючи беклог спринту. Під час спринту команда працює над завданнями, розробляючи, тестуючи та інтегруючи функції. Scrum-фреймворк заохочує щоденні стендап-зустрічі, які часто називають щоденними скрамами. Ці короткі, обмежені в часі зустрічі призначені для синхронізації роботи членів команди, обговорення прогресу та виявлення будь-яких перешкод або проблем.

Роль артефактів скраму

Скрам використовує кілька ключових артефактів, які допомагають керувати роботою та відстежувати прогрес. Беклог продукту, як згадувалося раніше, містить всю необхідну роботу. Беклог спринту складається з елементів, обраних для поточного спринту, разом з планом виконання

інкременту продукту і досягнення мети спринту. Інкремент - це сума всіх елементів беклогу продукту, завершених під час спринту і всіх попередніх спринтів, які повинні бути в придатному для використання стані.

Регулярні переоцінки та коригування

Однією з відмінних рис Scrum є акцент на регулярній переоцінці та коригуванні. Наприкінці кожного спринту відбуваються дві важливі зустрічі: огляд спринту та ретроспектива спринту. Підбиття підсумків спринту відбувається за участю зацікавлених сторін і дає можливість проконтролювати проріст і, за необхідності, адаптувати беклог продукту. Ретроспектива спринту проводиться з командою, щоб проаналізувати його, обговорити, що пройшло добре, що можна покращити, і на чому буде зосереджено увагу в наступному спринті. Цей цикл безперервного зворотного зв'язку та вдосконалення є фундаментальним для Скраму, гарантуючи, що команда завжди розвивається та адаптується до змін.

Ролі в скрамі

Скрам також визначає конкретні ролі: власника продукту, скрам-майстра та команди розробників. Власник продукту відповідає за максимізацію цінності продукту та управління відставанням продукту. Скрам-майстер виступає в ролі фасилітатора та коуча, допомагаючи всім зрозуміти теорію, практику, правила та цінності Скраму. Команда розробників - це група професіоналів, які працюють над створенням потенційно доступного Інкременту продукту в кінці кожного Спринту.

Канбан: Наголошує на безперервній доставці без перевантаження членів команди. Наголошує на безперервному виконанні робіт без перевантаження членів команди

Канбан, що походить від японських виробничих систем, є ще одним ключовим фреймворком Agile, який фокусується на візуалізації роботи, максимізації ефективності та досягненні безперервної доставки. Основний принцип Канбан полягає в управлінні роботою шляхом збалансування потреб

з наявними потужностями та покращення роботи з вузькими місцями на рівні системи.

Візуальне управління за допомогою дошок Канбан

Канбан найбільш відомий завдяки використанню дошок Канбан - візуального інструменту, який допомагає відстежувати роботу на різних етапах процесу. Типова дошка Канбан розділена на колонки, які представляють різні етапи робочого процесу, такі як "Потрібно зробити", "У процесі" і "Виконано". Робочі елементи, представлені картками або стікерами, розміщуються в цих колонках, забезпечуючи чіткий огляд поточного стану роботи, відповідальних за кожне завдання, а також будь-яких вузьких місць або затримок в процесі.

Обмеження незавершеного виробництва

Ключовим аспектом Канбану є обмеження кількості незавершених робіт (WIP). Встановлюючи ліміти незавершеного виробництва для кожного етапу робочого процесу, Канбан гарантує, що команда зосередиться на виконанні поточних завдань, перш ніж братися за нові. Такий підхід допомагає зменшити багатозадачність і перемикання контексту, що призводить до більш сфокусованої роботи і швидшого завершення. Він також допомагає виявити неефективність процесу та вузькі місця, оскільки накопичення робочих елементів на певному етапі вказує на проблему, яку потрібно вирішити.

Безперервний потік і вдосконалення

Канбан сприяє безперервному потоку роботи, коли завдання витягуються через систему в міру можливостей, а не проштовхуються на основі передбачуваних графіків. Ця система витягування узгоджує роботу з можливостями команди, зменшуючи перевантаження і стрес. Канбан також заохочує постійне, поступове вдосконалення. Команди регулярно переглядають свої робочі процеси, використовуючи такі показники, як час виконання (час, необхідний для завершення робочого елементу) і пропускну здатність (кількість елементів, виконаних за певний період часу), щоб визначити сфери для вдосконалення.

Гнучкість та оперативність

Однією з сильних сторін Kanban є його гнучкість. Його можна легко адаптувати до конкретних потреб і робочих процесів різних команд, а також інтегрувати з іншими гнучкими методологіями. Акцент Kanban на візуальному управлінні та безперервному виконанні робить його чудовим інструментом для команд, які повинні швидко реагувати на зміну пріоритетів та потреб клієнтів.

Екстремальне програмування (XP): Extreme Programming (XP) - це фреймворк Agile, орієнтований насамперед на підвищення якості розробки програмного забезпечення та реагування на мінливі потреби клієнтів. XP характеризується акцентом на відмінній технічній практиці, командній роботі та чіткій комунікації.

Основні практики та принципи

XP запроваджує набір практик та принципів, які сприяють підвищенню якості та швидкості розробки програмного забезпечення. До них відносяться

- Розробка, керована тестами (Test-Driven Development, TDD): Розробники спочатку пишуть автоматизовані тести для нових функцій, а потім створюють код, який проходить ці тести. Це гарантує, що весь новий код негайно тестується і заохочує простий дизайн і високу якість коду.

- Парне програмування: Два програмісти працюють разом на одній робочій станції. Один пише код, а інший перевіряє кожен рядок після написання. Така співпраця призводить до вищої якості коду, меншої кількості помилок та кращої передачі знань.

- Безперервна інтеграція (CI): Зміни коду часто інтегруються в основну гілку сховища вихідного коду. Ця практика дозволяє командам виявляти проблеми на ранніх стадіях і зменшує кількість конфліктів при інтеграції.

- Рефакторинг: Регулярний рефакторинг коду, або покращення його структури без зміни поведінки, гарантує, що кодова база залишається чистою

та зрозумілою, що є критично важливим для швидкої адаптації до мінливих вимог.

- Простий дизайн: XP заохочує прості та зрозумілі дизайнерські рішення, які відповідають поточним вимогам і легко змінюються в майбутньому.

- Залучення клієнтів: Регулярне, часто щоденне спілкування з клієнтами має важливе значення. Клієнти надають відгуки про кожну ітерацію та уточнюють вимоги, гарантуючи, що команда розробників завжди працює над найціннішими функціями.

Наголос на зворотному зв'язку та комунікації

XP робить сильний акцент на зворотному зв'язку, не тільки від автоматизованих тестів і кінцевих користувачів (клієнтів), але також через регулярні зустрічі команди і ретроспективи. Цей цикл зворотного зв'язку є ключем до адаптивності та оперативності XP. Регулярне спілкування всередині команди і з замовником забезпечує узгодженість і ясність цілей, зменшуючи непорозуміння і даремні зусилля.

Сталий темп роботи

XP сприяє підтримці сталого темпу роботи, уникаючи понаднормової роботи та вигорання. Цей підхід ґрунтується на розумінні того, що перевантажені команди роблять більше помилок і є менш продуктивними в довгостроковій перспективі.

Адаптація до змін

В основі XP лежить здатність адаптуватися до мінливих вимог, навіть на пізніх стадіях процесу розробки. Практики XP спрямовані на те, щоб зробити зміни менш дорогими і більш керованими, гарантуючи, що програмне забезпечення постійно розвивається, щоб відповідати поточним потребам замовника.

Lean: фокусується на мінімізації відходів та максимізації цінності.

Lean, що походить від принципів ощадливого виробництва, - це фреймворк Agile, який наголошує на ефективності та зменшенні відходів,

одночасно максимізуючи споживчу цінність. Lean застосовується до розробки програмного забезпечення та управління проектами, зосереджуючись на створенні цінності за допомогою оптимізованих процесів.

Ключові принципи Lean

- Усунення відходів: Методологія Lean визначає та усуває відходи (діяльність, що не створює доданої вартості) в процесі розробки. Сюди входить непотрібний код, завдання, документація та процеси, які не сприяють досягненню кінцевої мети. Зменшуючи відходи, команди можуть більше зосередитися на діяльності, яка безпосередньо додає цінності для замовника.

- Посилуйте навчання: Безперервне навчання та вдосконалення є центральним елементом Lean. Це передбачає регулярні цикли зворотного зв'язку, рефлексію та коригування процесів для забезпечення постійного підвищення якості та ефективності.

- Приймайте рішення якомога пізніше: Lean заохочує відкладати прийняття рішень до останнього відповідального моменту. Такий підхід дозволяє приймати більш гнучкі та обґрунтовані рішення, зменшуючи ризик дорогих змін та переробок.

- Виконувати якомога швидше: Зосередженість на швидкій доставці має на меті забезпечити швидку цінність для клієнтів і швидше отримати зворотній зв'язок. Це дозволяє командам вчасно вносити корективи та вдосконалення.

- Надайте команді більше повноважень: Lean наголошує на наданні автономії команді, дозволяючи приймати рішення тим, хто є найближчим до роботи. Це дозволяє командам бути більш інноваційними та швидко реагувати на зміни.

- Виховуйте доброчесність: Lean фокусується на створенні якості продукту з самого початку. Це включає в себе чіткий та надійний дизайн, ефективні процеси тестування та прихильність до якості в кожному аспекті процесу розробки.

- Оптимізуйте все в цілому: Lean заохочує дивитися на проект або продукт цілісно. Замість того, щоб оптимізувати окремі частини ізольовано, Lean фокусується на всьому потоці створення цінності, щоб гарантувати, що кожна частина процесу узгоджена і сприяє досягненню загальної мети.

Скорочення відходів в Lean

У Lean скорочення відходів поділяється на кілька типів, таких як

- Перевиробництво: Робити більше, ніж потрібно, або раніше, ніж це потрібно.

- Дефекти: Зусилля, витрачені на виправлення дефектів.

- Очікування: Час простою, коли ресурси не використовуються ефективно.

- Невикористаний талант: Неповне використання навичок і здібностей людей.

- Переміщення та рух: Непотрібні рухи або передачі в процесі роботи.

- Інвентаризація: Робота, яка простоює і не виконується.

Картування потоку створення цінності (Value Stream Mapping)

Ключовим інструментом ощадливого виробництва є картування потоків створення цінності, яке допомагає візуалізувати і зрозуміти потік матеріалів та інформації, коли продукт або послуга рухається по ланцюжку створення цінності. Ця візуалізація допомагає виявити вузькі місця, затримки і втрати, надаючи уявлення про те, як можна оптимізувати процеси.

Гнучка методологія на практиці

Agile успішно впроваджується в різних секторах, а не тільки в розробці програмного забезпечення. Компанії повідомляють про підвищення ефективності, скорочення термінів виконання замовлень і підвищення рівня задоволеності клієнтів.

Методологія Agile вирізняється своєю гнучкістю, зосередженістю на задоволенні потреб клієнтів та здатністю адаптуватися до змін. Оскільки бізнес-середовище продовжує стрімко розвиватися, принципи Agile є більш

актуальними, ніж будь-коли, пропонуючи надійну основу для управління проектами у швидкоплинному світі, що постійно змінюється.

1.2 Гнучкі та традиційні підходи до управління проектами

У сфері управління проектами переважають дві основні методології: Гнучка та традиційна (яку часто називають "водоспад"). Розуміння відмінностей між цими підходами має вирішальне значення для вибору правильного методу для проекту.

Традиційне управління проектами

Традиційне управління проектами - це методологія, яка наголошує на структурованому, систематичному підході до управління проектами. Ця методологія характеризується опорою на встановлені процеси, чітко визначені етапи та велику кількість документації. Її часто асоціюють з моделлю водоспаду, яка описує послідовний розвиток фаз проекту. Тут ми заглиблюємося в ключові аспекти традиційного управління проектами.

1. Послідовні фази

Однією з характерних рис традиційного управління проектами є його лінійний і послідовний характер. Підхід Waterfall є прикладом цього, де кожна фаза проекту повинна бути завершена, перш ніж переходити до наступної. Ці фази, як правило, включають

- Ініціація: Визначення проекту на високому рівні та отримання необхідних дозволів.
- Планування: Розробка детальних планів, графіків та бюджетів проекту.
- Виконання: Реалізація плану проекту та управління проектною командою.
- Моніторинг та контроль: Відстеження прогресу, управління змінами та забезпечення того, щоб проект не відставав від графіка.

- Закриття: Завершення всіх проектних заходів, отримання схвалення та закриття проекту.

Такий структурований підхід гарантує, що кожна фаза ґрунтується на попередній, забезпечуючи чітку дорожню карту від початку проекту до його завершення.

2. Детальне планування

Традиційне управління проектами значною мірою спирається на детальне планування. Це передбачає широку діяльність з попереднього планування, в тому числі:

- Визначення обсягу робіт: Чітке визначення обсягу та цілей проекту.
- Збір вимог: Збір та документування детальних вимог від зацікавлених сторін.
- Планування ресурсів: Визначення та розподіл ресурсів, необхідних для проекту.
- Управління ризиками: Аналіз потенційних ризиків та розробка стратегій їх зменшення.
- Планування: Створення детальних графіків проекту з етапами та дедлайнами.

Таке ретельне планування має на меті забезпечити, щоб усі аспекти проекту були ретельно розглянуті та задокументовані до початку його виконання.

3. Передбачуваність і контроль

Ключовою метою традиційного управління проектами є досягнення передбачуваності та контролю над проектом. Це досягається за допомогою

- Фіксовані вимоги: Встановлення фіксованих вимог до проекту на ранній стадії планування.
- Управління змінами: Впровадження суворих процесів контролю змін для управління та затвердження змін.

- Інструменти моніторингу: Використання різних інструментів і методів, таких як діаграми Ганта і показники ефективності, для моніторингу прогресу і результативності проекту.

Хоча цей підхід забезпечує високий рівень контролю, він також може зробити зміни дорогими і складними для впровадження після завершення проекту.

4. Акцент на документації

Документація відіграє важливу роль у традиційному управлінні проектами. Кожна фаза генерує значну кількість документації, в тому числі

- Плани проекту: Детальні плани, що окреслюють обсяг проекту, графік, бюджет і ресурси.

- Документи з вимогами: Вичерпні списки проектних вимог та специфікацій.

- Звіти про хід виконання проекту: Регулярні оновлення статусу, ходу та результатів проекту.

- Фінальні звіти: Документація щодо результатів проекту, отриманих уроків та заходів із закриття проекту.

Такий акцент на документації гарантує, що всі деталі проекту зафіксовані і доступні для довідки, але він також може створювати проблеми з адаптацією до змін.

5. Визначений кінцевий продукт

У традиційному управлінні проектами кінцевий продукт чітко визначений з самого початку. Цілі, результати та підсумки проекту встановлюються на етапі планування, залишаючи мало місця для відхилень. Такий підхід гарантує, що всі зацікавлені сторони мають чітке розуміння того, чого прагне досягти проект, і можуть відповідно управляти своїми очікуваннями.

Хоча традиційний підхід пропонує структуру і передбачуваність, він може не підходити для всіх типів проектів, особливо тих, які вимагають гнучкості та адаптивності. Жорсткість послідовних фаз і акцент на фіксованих

вимогах можуть ускладнити реагування на зміни і невизначеності, які можуть виникнути протягом життєвого циклу проекту.

Отже, традиційне управління проектами забезпечує дисциплінований і організований підхід до управління проектами, що робить його ідеальним для проектів з чітко визначеними цілями і стабільними вимогами. Однак його обмеження в управлінні змінами і невизначеностями підкреслюють потребу в альтернативних методологіях, таких як Agile, в певних проектних середовищах.

Гнучке управління проектами

1. Ітеративний та інкрементальний:

Методологія Agile використовує ітеративний та інкрементний підхід до управління проектами, що є ключовим аспектом, який відрізняє її від традиційних методологій, таких як Waterfall. Цей підхід передбачає розбиття проектів на менші, керовані частини, які виконуються під час робочих сесій, відомих як спринти. Кожен спринт зазвичай триває від одного до чотирьох тижнів, що дозволяє командам регулярно створювати функціональні компоненти продукту та отримувати безперервний зворотній зв'язок.

Ітеративна розробка

При ітеративній розробці проект поділяється на цикли, або ітерації, кожна з яких призводить до збільшення робочого продукту. Ці ітерації є короткими, фіксованими періодами, протягом яких команда працює над конкретними функціями або компонентами продукту. Наприкінці кожної ітерації команда аналізує прогрес, збирає відгуки та планує наступну ітерацію. Цей циклічний процес дозволяє командам поступово допрацьовувати та вдосконалювати продукт на основі зворотного зв'язку та мінливих вимог.

Ключові аспекти ітеративної розробки включають

- Безперервний зворотній зв'язок: Наприкінці кожної ітерації прогрес продукту демонструється зацікавленим сторонам, які надають зворотній зв'язок. Цей зворотній зв'язок має вирішальне значення для внесення коригувань і забезпечення відповідності продукту потребам користувачів.

- **Гнучкість:** Ітеративна розробка дозволяє вносити зміни та вдосконалення на будь-якому етапі проекту. Команди можуть адаптуватися до нових вимог або змін на ринку, гарантуючи, що кінцевий продукт залишається актуальним і цінним.

- **Зниження ризиків:** Регулярно впроваджуючи невеликі функціональні доповнення, команди можуть виявляти та вирішувати проблеми на ранніх стадіях процесу розробки. Це зменшує ризик виникнення серйозних проблем на пізніх етапах проекту.

Інкрементна розробка

Інкрементна розробка доповнює ітеративний процес, зосереджуючись на створенні продукту невеликими, керованими кроками. Кожен наступний крок ґрунтується на попередньому, поступово додаючи продукту функціональності та цінності. Такий підхід гарантує, що продукт постійно розвивається з плином часу, а кожна наступна частина є завершеною і придатною для використання частиною кінцевого продукту.

Ключові аспекти інкрементної розробки включають

- **Рання доставка:** Навіть на ранніх стадіях проекту команди створюють робочі інкременти продукту, з якими користувачі можуть взаємодіяти. Таке раннє надання цінності підвищує задоволеність клієнтів і забезпечує негайну вигоду.

- **Орієнтованість на користувача:** інкрементна розробка надає пріоритет функціям і можливостям, які приносять найбільшу користь користувачам. Зосереджуючись на потребах та вподобаннях користувачів, команди можуть забезпечити відповідність продукту очікуванням клієнтів.

- **Покращена якість:** З кожним оновленням, що проходить ретельне тестування та перевірку, загальна якість продукту поступово покращується. Команди можуть виявляти та виправляти проблеми поступово, що призводить до більш надійного кінцевого продукту.

Спринт-цикл

В Agile ітеративний та інкрементний підхід реалізується через спринт-цикл. Кожен спринт складається з наступних етапів:

- **Планування спринту:** На початку кожного спринту команда проводить планерку, щоб визначити цілі та завдання на майбутній спринт. Команда обирає користувацькі історії з беклогу продукту, розставляє пріоритети та визначає беклог спринту.

- **Виконання спринту:** Під час спринту команда працює над обраними завданнями, тісно співпрацюючи над розробкою, тестуванням та інтеграцією нових функцій. Щоденні наради допомагають команді залишатися злагодженою та вирішувати будь-які перешкоди.

- **Підбиття підсумків спринту:** Наприкінці спринту команда проводить оглядову зустріч, щоб продемонструвати завершену роботу зацікавленим сторонам. Ця демонстрація дає можливість стейкхолдерам надати зворотній зв'язок і запропонувати покращення.

- **Ретроспектива спринту:** Після огляду команда проводить ретроспективну зустріч, щоб обговорити процес спринту. Команда обговорює, що пройшло добре, що можна покращити, а також визначає практичні кроки для покращення майбутніх спринтів.

- Цей структурований, але гнучкий процес забезпечує безперервне створення цінності, дозволяючи командам швидко адаптуватися до змін і підтримувати стабільний темп розвитку. Ітеративний та інкрементний підхід Agile сприяє створенню динамічного та гнучкого проектного середовища, що в кінцевому підсумку призводить до вищої якості продуктів та більшої задоволеності клієнтів.

2. Адаптивність

Адаптивність є основним принципом методології Agile, що дозволяє командам ефективно реагувати на зміни навіть на пізніх стадіях розробки. Ця гнучкість відрізняє Agile від традиційних методів, які часто борються зі змінами на пізніх стадіях. Адаптивність Agile ґрунтується на кількох ключових практиках:

Прийняття змін

Agile вітає зміни як можливість покращити продукт. Маніфест Agile цінує "реагування на зміни замість того, щоб слідувати плану", просуваючи гнучкий підхід. Гнучкі проекти поділяються на короткі ітеративні цикли, які називаються спринтами, кожен з яких забезпечує функціональний приріст продукту. Регулярний зворотний зв'язок із зацікавленими сторонами наприкінці кожного спринту дозволяє постійно вносити корективи та вдосконалення.

Ключові елементи, що сприяють адаптивності

- Ітеративний розвиток: Проекти розбиваються на цикли, які призводять до робочих інкрементів. Це забезпечує регулярний зворотній зв'язок і коригування, гарантуючи, що продукт розвивається відповідно до реальних потреб.
- Співпраця з клієнтами: Часта взаємодія із зацікавленими сторонами забезпечує відповідність проекту їхнім очікуванням, що дозволяє швидко адаптуватися до нових вимог.
- Динамічний бэклог: Беклог продукту постійно оцінюється за пріоритетами, що дозволяє команді зосередитися на найбільш важливих завданнях і змінювати пріоритети в разі потреби.
- Мінімальне попереднє планування: Agile уникає великого попереднього планування, яке може застаріти. Натомість він наголошує на плануванні "точно вчасно" для кожної ітерації.

Адаптація на пізніх стадіях

Структура Agile підтримує зміни на пізніх стадіях:

- Безперервна інтеграція та розгортання: Регулярна інтеграція змін коду та автоматизоване розгортання дозволяють швидко впроваджувати та випускати оновлення.
- Рефакторинг: Безперервний рефакторинг коду гарантує, що система залишається гнучкою та підтримуваною, плавно пристосовуючись до нових функцій та змін.

- Тест-орієнтована розробка (TDD): Написання тестів перед створенням коду гарантує функціональність і забезпечує підстраховку для внесення змін без дефектів.

- Ретроспектива: Регулярні рефлексивні зустрічі визначають сфери для вдосконалення, сприяючи розвитку культури постійної адаптації та вдосконалення.

Переваги адаптивності

Висока адаптивність в Agile призводить до:

- Підвищення рівня задоволеності клієнтів: Швидка реакція на відгуки та зміни гарантує, що кінцевий продукт відповідає потребам клієнта.

- Зниження ризиків: раннє виявлення та вирішення проблем запобігає виникненню серйозних проблем на більш пізніх етапах проекту.

- Підвищення інноваційності: Гнучкість заохочує експерименти та творче вирішення проблем.

- Покращення результатів проекту: Адаптивність Agile призводить до створення високоякісних продуктів, які надають реальну цінність для зацікавлених сторін.

3. Співпраця з клієнтами

Акцент на регулярному зворотному зв'язку

Одним з основоположних принципів методології Agile є акцент на регулярному зворотному зв'язку з клієнтами та зацікавленими сторонами. Цей принцип гарантує, що процес розробки залишається узгодженим з потребами та очікуваннями кінцевих користувачів, що постійно змінюються. Регулярний зворотній зв'язок забезпечується за допомогою декількох ключових практик:

- Спринт-огляди: Наприкінці кожного спринту Agile-команди проводять оглядові зустрічі, на яких демонструють завершену роботу зацікавленим сторонам. Це дає можливість для негайного зворотного зв'язку, що дозволяє команді внести необхідні корективи перед тим, як рухатися далі.

- Часті демонстрації: Гнучкі команди часто проводять демонстрації нових функцій або можливостей, щоб зібрати інформацію від зацікавлених

сторін. Ці демонстрації не обмежуються закінченням спринтів, але можуть відбуватися щоразу, коли завершується значна частина роботи.

- Історії користувачів та догляд за бэклогом: Гнучкі команди використовують історії користувачів, щоб зафіксувати вимоги з точки зору клієнта. Регулярні сесії догляду за бэклогом включають перегляд і визначення пріоритетів цих історій на основі відгуків зацікавлених сторін, гарантуючи, що найцінніші функції будуть розроблятися в першу чергу.

Безперервне вдосконалення

Безперервне вдосконалення, або Кайдзен, є ключовим аспектом Agile. Воно включає в себе ітеративні вдосконалення як продукту, так і процесу на основі зворотного зв'язку та рефлексії. Це досягається за допомогою:

- Ретроспективи: Наприкінці кожного спринту команди проводять ретроспективні зустрічі, щоб обговорити, що пройшло добре, а що ні, і як можна покращити процеси. Це сприяє розвитку культури навчання та адаптації.

- Інкрементальний розвиток: Ітеративний підхід Agile дозволяє командам вносити постійні, поступові покращення в продукт. Кожен спринт ґрунтується на попередньому, враховуючи зворотній зв'язок та нові ідеї для покращення кінцевого продукту.

Переваги співпраці з клієнтами

- Покращене узгодження: Регулярна співпраця гарантує, що команда розробників завжди відповідає потребам та бізнес-цілям замовника, зменшуючи ризик отримання невідповідного результату.

- Підвищення рівня задоволеності клієнтів: Постійний зворотній зв'язок та ітеративні вдосконалення призводять до створення продукту, який краще відповідає очікуванням клієнтів, що призводить до більшої задоволеності та лояльності.

- Раннє виявлення проблем: Регулярно взаємодіючи з клієнтами, команди можуть виявляти та вирішувати проблеми на ранніх стадіях процесу розробки, зменшуючи вартість та вплив змін.

- **Покращення якості:** Регулярний внесок зацікавлених сторін допомагає уточнити вимоги і гарантувати, що кінцевий продукт буде якісним і відповідатиме своєму призначенню.

4. Мінімум документації

У методології Agile принцип "мінімальної документації" підкреслює, що основна увага має бути зосереджена на створенні функціонального робочого продукту, а не на створенні обширної документації. Цей підхід контрастує з традиційними методологіями, які часто надають пріоритет вичерпній документації на кожному етапі проекту. Гнучкий підхід до документування керується наступними принципами:

Зосередьтеся на документації, що додає цінність

Гнучкі команди надають пріоритет документації, яка додає безпосередню цінність процесу розробки та кінцевому продукту. Це означає створення лише необхідних документів, які підтримують розробку, покращують комунікацію та гарантують, що продукт відповідає потребам користувачів. Ключові документи зазвичай включають

- **Історії користувачів:** Короткі, прості описи функції з точки зору кінцевого користувача. Історії користувачів допомагають командам зрозуміти вимоги та розставити пріоритети в роботі.

- **Критерії прийнятності:** Конкретні умови, яким повинен відповідати продукт або функція, щоб бути прийнятим користувачем або зацікавленою стороною. Ці критерії гарантують, що всі члени команди мають чітке розуміння того, чого потрібно досягти.

- **Спрінт беклог:** Список завдань або елементів, які потрібно виконати під час спринту. Беклог допомагає команді залишатися організованою і зосередженою на найважливішій роботі.

Вчасна документація

Agile сприяє створенню документації "точно вчасно", тобто документи створюються та оновлюються за потребою, а не всі одразу на початку проекту.

Такий підхід гарантує, що документація завжди буде актуальною та сучасною, відображаючи поточний стан проекту та його вимоги.

- Оновлення в режимі реального часу: Документація постійно оновлюється в режимі реального часу на основі останніх подій та відгуків. Це дозволяє команді бути узгодженою та поінформованою без тягаря ведення об'ємних, статичних документів.

- Інструменти для співпраці: Гнучкі команди часто використовують цифрові інструменти співпраці, такі як вікі, спільні документи та програмне забезпечення для управління проектами, щоб ефективно підтримувати та оновлювати документацію. Ці інструменти забезпечують легкий доступ та співпрацю між членами команди в режимі реального часу.

Спілкування через документацію

Гнучкий підхід робить акцент на безпосередньому спілкуванні, а не на письмовій документації. Розмови віч-на-віч, щоденні наради та регулярна взаємодія із зацікавленими сторонами забезпечують швидкий і точний потік інформації. Такий підхід зменшує потребу в детальній письмовій документації, оскільки члени команди можуть прояснити питання і проблеми через пряме спілкування.

- Щоденні стенд-апи: Короткі щоденні зустрічі, на яких члени команди обговорюють прогрес, плани та перешкоди. Ці зустрічі допомагають усім залишатися на одній хвилі та швидко вирішувати проблеми.

- Спринт-огляди та ретроспективи: Регулярні зустрічі для огляду прогресу, збору відгуків та роздумів про процеси в команді. Ці зустрічі сприяють постійному вдосконаленню та узгодженню з очікуваннями зацікавлених сторін.

Переваги мінімальної документації

- Підвищена гнучкість: Мінімізуючи документацію, гнучкі команди можуть швидше адаптуватися до змін у вимогах або пріоритетах. Ця гнучкість має вирішальне значення для реагування на мінливі потреби клієнтів та ринкові умови.

- Зменшення накладних витрат: Менше часу витрачається на створення та підтримку об'ємної документації, що означає, що більше часу можна присвятити власне роботі над розробкою. Це призводить до підвищення продуктивності та швидшої реалізації цінних функцій.

- Посилений фокус на результатах: З акцентом на робочому програмному забезпеченні, а не на вичерпній документації, Agile-команди зосереджуються на створенні функціональних, високоякісних продуктів, які відповідають потребам користувачів.

Підхід Agile до мінімальної документації гарантує, що процес розробки залишається ефективним, гнучким і зосередженим на створенні цінності для замовника. Надаючи пріоритет важливим документам, що додають цінність, і сприяючи прямому спілкуванню, Agile-команди можуть швидко адаптуватися до змін і підтримувати високий рівень продуктивності.

5. Гнучкий кінцевий продукт

В методології Agile концепція гнучкого кінцевого продукту є центральною для її адаптивного та ітеративного підходу. На відміну від традиційного управління проєктами, де кінцевий продукт визначається на початку і зміни вносити важко, Agile дозволяє продукту еволюціонувати протягом усього життєвого циклу проєкту на основі постійного зворотного зв'язку та ітеративного тестування. Ця гнучкість забезпечує, що кінцевий продукт краще відповідає потребам користувачів та зацікавлених сторін.

Постійний зворотний зв'язок

Методології Agile підкреслюють важливість регулярного зворотного зв'язку від клієнтів та зацікавлених сторін. Цей зворотний зв'язок є невід'ємною частиною забезпечення відповідності продукту потребам і очікуванням користувачів. Ключові практики, що сприяють цьому постійному зворотному зв'язку, включають:

1. Огляди спринтів: Після кожного спринту команди демонструють виконану роботу зацікавленим сторонам. Це надає можливість для негайного зворотного зв'язку, який можна врахувати в наступному спринті.

2. Співпраця з клієнтами: Команди Agile постійно співпрацюють з клієнтами протягом всього проєкту. Регулярні перевірки та обговорення допомагають забезпечити відповідність процесу розробки вимогам клієнтів.

Ітеративне тестування та розробка

Ітеративний підхід Agile включає розробку продукту невеликими, поступовими етапами. Кожна ітерація включає фази розробки, тестування та огляду, що дозволяє командам постійно вдосконалювати та покращувати продукт. Цей процес забезпечує:

1. Раннє та часте тестування: Функції тестуються по мірі їх розробки, що дозволяє виявляти та вирішувати проблеми на ранніх етапах проєкту. Це знижує ризик виникнення серйозних дефектів та забезпечує вищу якість.

2. Поступові покращення: Кожна ітерація базується на попередній, включаючи зворотний зв'язок та нові знання для покращення продукту. Цей циклічний процес дозволяє продукту поступово еволюціонувати та покращуватися.

Адаптивність до змін

Здатність адаптуватися до змін є основним принципом Agile. Ця адаптивність означає, що кінцевий продукт не є жорстко визначеним на початку, а дозволяє еволюціонувати по мірі появи нових ідей та вимог. Практики Agile, що підтримують цю адаптивність, включають:

1. Динамічний беклог: Продуктовий беклог є живим документом, який постійно оновлюється та переоцінюється на основі зворотного зв'язку та змінних вимог. Це забезпечує, що найцінніші функції розробляються першими.

2. Гнучкий обсяг: Команди Agile приймають зміни в обсязі робіт, навіть на пізніх етапах процесу розробки. Ця гнучкість дозволяє командам швидко змінювати напрямок та фокусуватися на основі нової інформації або змін на ринку.

Переваги гнучкого кінцевого продукту

1. **Краща відповідність потребам користувачів:** Постійно включаючи зворотний зв'язок, кінцевий продукт більше відповідає потребам та очікуванням користувачів, що підвищує задоволеність та зручність.

2. **Збільшена інноваційність:** Гнучкість дозволяє командам експериментувати з новими ідеями та підходами, стимулюючи інновації та творчість у процесі розробки.

3. **Зниження витрат:** Фокусування на наданні поступової цінності забезпечує, що ресурси витрачаються на функції та можливості, які дійсно необхідні, зменшуючи втрати та підвищуючи ефективність.

4. **Підвищена конкурентоспроможність:** Здатність швидко адаптуватися до змін на ринку та зворотного зв'язку користувачів дає командам Agile конкурентну перевагу, дозволяючи їм створювати більш актуальні та своєчасні продукти.

Таким чином, гнучкість кінцевого продукту в методології Agile забезпечує, що розробка залишається орієнтованою на користувача та адаптивною до змін. Включаючи постійний зворотний зв'язок та ітеративне тестування, команди Agile можуть створювати високоякісні продукти, що еволюціонують для задоволення динамічних потреб користувачів та зацікавлених сторін.

Порівняння підходів

1. **Гнучкість та адаптація до змін**
 - Agile є більш гнучким, дозволяючи вносити зміни протягом усього життєвого циклу проекту.
 - Традиційні методи є жорсткими, а зміни є складними та дорогими.
2. **Планування та документація проекту**
 - Традиційні методи передбачають велику кількість документації та детальне попереднє планування.
 - Agile фокусується на полегшеній документації та адаптивному плануванні.
3. **Залучення клієнтів та зворотній зв'язок**

- Agile заохочує постійне залучення клієнтів і регулярну адаптацію на основі зворотного зв'язку.

- Традиційні методи передбачають обмежену взаємодію з клієнтом, зазвичай лише на ключових етапах.

4. Управління ризиками

- Ітеративний характер Agile дозволяє виявляти проблеми на ранніх стадіях і безперервно вдосконалювати їх, зменшуючи ризики.

- Традиційні методи можуть виявити ризики або помилки лише наприкінці проекту.

5. Реалізація проекту та часові рамки

- Agile спрямований на швидку реалізацію невеликих функціональних частин проекту.

- Традиційні методи часто призводять до більш тривалих термінів виконання через послідовний характер роботи.

6. Придатність для різних типів проектів

- Гнучкий ідеально підходить для проектів з невизначеними або мінливими вимогами.

- Традиційний підходить для проектів з чітко визначеними, стабільними вимогами та чіткими результатами.

-

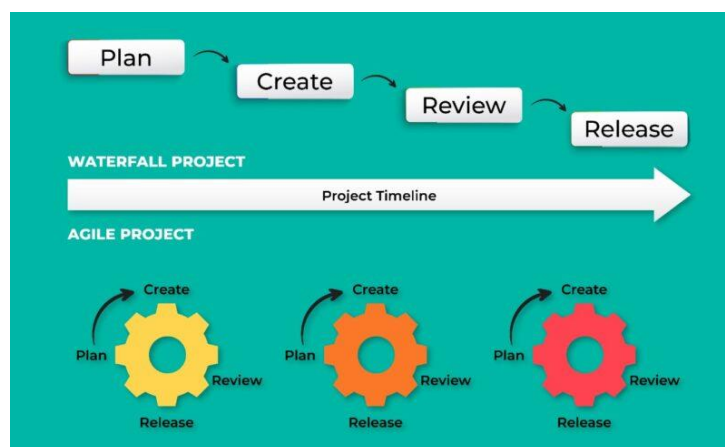


Рис 1.2. Візуалізація порівняння гнучкого та традиційного підходу

Джерело: [16]

Вибір між гнучкими та традиційними методами управління проектами значною мірою залежить від вимог проекту, очікувань зацікавлених сторін та рівня невизначеності. Agile пропонує гнучкість і адаптивність, що робить його добре придатним для проектів у швидкозмінному середовищі. На противагу цьому, традиційним методологіям надають перевагу в контекстах, де чіткий, незмінний шлях є вигідним.

1.3 Гнучке управління проектами в ТОВ “ГлобалЛоджик Україна”

Компанія ТОВ “ГлобалЛоджик Україна” є сервісною компанією з розробки цифрових продуктів. ГлобалЛоджик Україна є частиною міжнародної компанії GlobalLogic з головним офісом в Сполучених Штатах Америки. GlobalLogic в свою чергу є частиною Hitachi Group Company з головним офісом в Японії.

GlobalLogic допомагає клієнтам розробляти та створювати інноваційні продукти, платформи та цифрові сервіси для сучасного світу. Інтегруючи свій стратегічний дизайн, комплексний інжиніринг та вертикальну галузеву експертизу з можливостями операційних та інформаційних технологій Hitachi, GlobalLogic допомагаємо клієнтам уявити, що можливо, та прискорює їхній перехід до цифрового бізнесу завтрашнього дня. Штаб-квартира GlobalLogic розташована в Кремнієвій долині, а дизайн-студії та інжинірингові центри працюють по всьому світу, поширюючи глибоку експертизу на клієнтів в автомобільній, комунікаційній, фінансовій галузях, охороні здоров'я та медико-біологічних науках, медіа та розвагах, виробництві, напівпровідниковій та технологічній індустрії.

Деякі факти про GlobalLogic:

- Заснована у 2000 році
- Придбана компанією Hitachi Ltd. у 2021 році
- 30 050+ співробітників у 18 країнах

- 500+ активних клієнтів
- 70+ лабораторій під власними торговими марками
- 47 інженерних центрів
- 8 дизайн-студій
- 12 офісів продажів
- 350+ UX/UI дизайнерів та розробників через Method, наш досвідчений дизайнерський підрозділ

GlobalLogic надає послуги Agile розробки, які революціонізують поточні процеси розробки. Принципи гнучкої розробки стали фундаментальним елементом у просуванні послуг світового класу. Компанія надає комплексні послуги з Agile консалтингу та впровадження, допомагаючи організаціям швидше навчатися та ставати ефективнішими. GlobalLogic використовує підхід до проектування та системного мислення для впровадження гнучкості підприємства, що базується на принципах Lean. Це призводить до підвищення продуктивності та створення щасливих команд.

Компанія GlobalLogic надає наступні послуги:

- Розробка програмного забезпечення з використанням принципів та практик Agile
- Програма Agile-коучингу
- Agile-консалтинг
- Agile для підприємств
- Оцінювання та звітність за допомогою Agile-інструментів

Для ефективного впровадження гнучких методів управління (Agile) в компанії GlobalLogic створена Agile практика. Задача Agile практики полягає у визначенні та координації ідейного лідерства GlobalLogic через експертів у різних сферах гнучких методів роботи, які включають Scrum, Kanban, SAFe, Lean та інші гнучкі методи. Дивись інформацію про GlobalLogic можливості в Додаток А та Додаток Б. Команда GL Agile Practice допомагає проектним командам визначити індивідуальні гнучкі методи роботи, допомагає командам з навчанням, коучингом та наставництвом, а також допомагає розробляти та

впроваджувати Agile Frameworks of Maturity (системи гнучкої зрілості). GL Agile Practice також допомагає клієнтам з консультаціями, які допомагають клієнтам оцінити їх поточну зрілість і створити дорожні карти для гнучкої трансформації підприємства, відповідної організаційної структури та управління.

РОЗДІЛ 2. ВПРОВАДЖЕННЯ AGILE В ОРГАНІЗАЦІЯХ

2.1 Підготовка організації до Agile

Перехід на гнучкі методології - це значна організаційна зміна, яка потребує ретельного планування та виконання. Підготовка організації до Agile передбачає не лише впровадження нових процесів, але й зміну культури, мислення та операційних норм.

1. Оцінка організаційної готовності

- Розуміння Agile: ознайомлення зацікавлених сторін з принципами та практиками Agile. Перш ніж розпочати шлях Agile, дуже важливо, щоб вся організація, особливо зацікавлені сторони, розуміли, що таке Agile і чим він не є. Це передбачає вихід за межі базових визначень і заглиблення в основні принципи та практики, які лежать в основі Agile-методологій, таких як Scrum, Kanban та Lean. Для пояснення таких понять, як ітеративна розробка, безперервний зворотний зв'язок та важливість крос-функціональних команд, слід проводити воркшопи, семінари та інтерактивні сесії. Також корисно обговорити переваги Agile, такі як підвищена гнучкість, покращена якість продукції та підвищена задоволеність клієнтів, щоб заручитися підтримкою та ентузіазмом щодо трансформації.

- Оцінка готовності: Оцініть поточні процеси, культуру та мислення, щоб визначити, наскільки організація готова до Agile. Оцінка організаційної готовності передбачає ретельне вивчення поточних процесів, культурної динаміки та загального мислення працівників. На цьому етапі необхідно визначити, чим існуючі методології управління проектами відрізняються від гнучких практик, а також виявити сфери, які потребують значних змін. Оцінка готовності може включати опитування, інтерв'ю та аналіз результатів минулих проектів, щоб визначити, наскільки добре поточна структура може адаптуватися до принципів Agile. Ключовими сферами, на

яких слід зосередитися, є толерантність організації до ризику, її реакція на зміни, канали комунікації та процеси прийняття рішень. Така оцінка допомагає виявити потенційні перешкоди та сфери, які можуть потребувати більш цілеспрямованих зусиль в управлінні змінами.

- Виявлення чемпіонів: Визначте та наділіти повноваженнями внутрішніх Agile-чемпіонів, які можуть просувати та спрямовувати Agile-трансформацію. Виявлення та розширення можливостей Agile-чемпіонів є ключовим фактором у впровадженні Agile-трансформації. Ці лідери, як правило, є ентузіастами, які глибоко розуміють Agile та мають вплив, щоб мотивувати інших. Вони можуть бути з будь-якого рівня організації - від керівників до членів команди. Роль таких чемпіонів багатогранна: вони виступають у ролі наставників, допомагають вирішувати перешкоди, сприяють впровадженню Agile-практик та слугують мостом між командами та вищим керівництвом. Забезпечення цих лідерів додатковим навчанням та ресурсами може підвищити їхню ефективність. Вони відіграють вирішальну роль у підтримці імпульсу гнучкої трансформації, подоланні скептицизму та впровадженні цінностей і принципів Agile у повсякденну діяльність організації.

2. Розвиток гнучкого мислення

- Культурна зміна: Підкресліть важливість гнучкості, співпраці та оперативності. Перехід до гнучкого мислення починається зі значних культурних змін в організації. Ці зміни зосереджені на вкоріненні таких цінностей, як гнучкість, співпраця та оперативність на всіх рівнях організації. Гнучкість передбачає відкритість до змін та адаптацію до мінливих вимог проекту та ринкових умов. Співпраця підкреслює важливість командної роботи та колективного вирішення проблем, переходу від ізольованих відділів до інтегрованих, міжфункціональних команд. Оперативність - це здатність швидко реагувати на відгуки клієнтів та мінливі вимоги. Семінари, тимблдинги та регулярні комунікації можуть бути ефективними у формуванні цих цінностей. Також важливо створити середовище, в якому експерименти та

навчання на невдачах розглядаються як сходинки до інновацій та вдосконалення, а не як невдачі.

- Підтримка керівництва: Заручіться підтримкою вищого керівництва для формування культури, яка сприймає зміни та заохочує цінності Agile. Роль лідерства у розвитку гнучкого мислення неможливо переоцінити. Лідери та вище керівництво повинні не лише підтримувати гнучку трансформацію в принципі, але й брати активну участь у змінах та пропагувати їх. Це передбачає, що лідери повинні подавати приклад, приймаючи цінності Agile, наприклад, бути відкритими до зворотного зв'язку, сприяти прозорості та демонструвати прихильність до постійного вдосконалення. Вони також повинні розширювати можливості команд, делегуючи їм повноваження щодо прийняття рішень і надаючи ресурси, необхідні для впровадження Agile-практик. Лідери відіграють важливу роль в усуненні перешкод, з якими можуть зіткнутися команди, і в створенні безпечного середовища, в якому заохочується прийняття ризиків. Регулярне інформування керівництва про важливість гнучкої трансформації, її переваги та досягнутий прогрес посилює її значення і тримає її на передньому краї організаційних пріоритетів.

- Залучення працівників: Залучайте працівників на всіх рівнях до процесу переходу, щоб сформувати колективне гнучке мислення. Залучення працівників на всіх рівнях має вирішальне значення для розвитку колективного гнучкого мислення. Це залучення може набувати різних форм: від залучення працівників до процесів прийняття рішень до отримання їхньої думки про те, як найкраще впроваджувати Agile-практики. Тренінги та практичні семінари можуть допомогти працівникам зрозуміти свою роль у гнучкому середовищі і те, як вони можуть сприяти його успіху. Створення в організації груп підтримки Agile або спільнот практиків може сприяти підвищенню почуття причетності та заохоченню до обміну досвідом і найкращими практиками. Регулярні сесії зворотного зв'язку, де працівники можуть висловити свої занепокоєння та пропозиції, допомагають

відшліфувати підхід до впровадження Agile. Визнання та святкування успіхів, навіть невеликих, під час переходу може підвищити моральний дух і посилити цінність гнучкої трансформації.

3. Навчання та розвиток навичок

- Гнучкі навчальні програми: Проводьте тренінги для команд та окремих працівників щодо методологій та інструментів Agile. Впровадження гнучких навчальних програм є ключовим кроком на шляху переходу до Agile. Ці програми повинні відповідати різним рівням наявних знань та досвіду в організації, пропонуючи як базові, так і поглиблені курси з Agile-методологій. Важливо охопити різні фреймворки Agile, такі як Scrum, Kanban та Lean, що дозволить командам та окремим працівникам зрозуміти нюанси кожного з них і те, як вони можуть бути застосовані в їхніх конкретних робочих контекстах. Навчання також має включати практичні заняття з використання інструментів та програмного забезпечення Agile, які є важливими для ефективного управління проектами Agile. Навчальні програми повинні бути максимально інтерактивними та практичними, заохочувати активну участь та вирішення реальних проблем. Пропонуючи поєднання очних семінарів, онлайн-курсів та самостійних навчальних матеріалів, можна пристосуватись до різних стилів та графіків навчання.

- Коучинг та наставництво: Надавати постійний коучинг та наставництво, щоб допомогти командам адаптуватися до Agile-практик. Коучинг та наставництво відіграють вирішальну роль в успішному впровадженні Agile. Agile-тренери або досвідчені фахівці можуть допомогти командам розібратися в практичних аспектах впровадження Agile-методологій у повсякденній роботі. Ці експерти можуть допомогти виявити та вирішити конкретні проблеми, з якими можуть зіткнутися команди, наприклад, проблеми з комунікацією, управлінням бэклогом або плануванням спринтів. Менторство також може надаватися внутрішніми чемпіонами Agile, які можуть поділитися своїм досвідом та інсайтами. Така безперервна підтримка допомагає командам не лише навчитися практикам Agile, але й

засвоїти мислення Agile, зосереджуючись на постійному вдосконаленні, співпраці та клієнтоорієнтованості. Регулярні коучингові сесії в поєднанні з наставництвом на робочому місці гарантують правильне розуміння та ефективно впровадження Agile-практик.

- Крос-функціональні навички: Сприяти розвитку крос-функціональних навичок для підвищення універсальності та співпраці команд. Розвиток крос-функціональних навичок є фундаментальним для гнучких команд, щоб підвищити їхню універсальність та сприяти співпраці. Крос-функціональні тренінги допомагають членам команди отримати широке розуміння різних аспектів проекту, від розробки до тестування, і навіть у таких сферах, як UX дизайн та управління продуктом. Таке навчання сприяє більш цілісному баченню життєвого циклу проекту і дозволяє членам команди ефективніше підтримувати один одного, що призводить до підвищення ефективності та швидшого вирішення проблем. Семінари, ротація робочих місць та парне програмування є ефективними способами заохочення розвитку цих навичок. Заохочення членів команди виходити за рамки своїх традиційних ролей і опанувати нові навички не лише підвищує спроможність команди, але й підвищує моральний дух і задоволеність роботою, оскільки люди почувуються більш впевненими і залученими до роботи.

4. Адаптація організаційних структур

- Командні структури: Перехід до крос-функціональних команд, які самоорганізуються та уповноважені приймати рішення. Перехід до Agile вимагає фундаментальних змін у структурі команди. Відходячи від традиційних ієрархічних та департаментських поділів, Agile віддає перевагу крос-функціональним командам. Такі команди об'єднують різні набори навичок, що дозволяє більш цілісно підходити до розробки проектів та вирішення проблем. У середовищі Agile команди самоорганізуються, тобто вони самостійно планують, виконують і контролюють свою роботу. Така автономія дає командам можливість приймати рішення швидко та ефективно, без необхідності постійного схвалення з боку вищого керівництва, що сприяє

формуванню почуття відповідальності та підзвітності. Щоб полегшити цей перехід, організаціям може знадобитися переглянути поточний склад команди, забезпечити збалансоване поєднання навичок у кожній команді та провести навчання членів команди для адаптації до цього нового способу роботи. Регулярні вправи та семінари з командоутворення також можуть допомогти зміцнити командну динаміку та сприяти розвитку духу співпраці.

- **Канали комунікації:** Створення відкритих та ефективних каналів комунікації для сприяння співпраці та прозорості. Agile процвітає завдяки відкритій та ефективній комунікації. Встановлення чітких і прямих каналів комунікації має вирішальне значення для сприяння співпраці та прозорості в організації. Це передбачає руйнування ізоляції та заохочення відкритих діалогів між різними відділами та рівнями організації. Такі інструменти, як щоденні стендапи, спринт-огляди та ретроспективи, є невід'ємними частинами Agile-методологій, що сприяють регулярній комунікації. Крім того, використання таких технологій, як програмне забезпечення для гнучкого управління проектами, може забезпечити платформи для безперервної комунікації та обміну інформацією. Створення організаційної культури, де зворотний зв'язок вітається і на нього реагують, також є життєво важливим. Така культура відкритої комунікації допомагає швидко виявляти та вирішувати проблеми, сприяє навчанню та орієнтує всіх на досягнення спільних цілей.

- **Показники ефективності:** Переосмислити показники ефективності, щоб вони відповідали цінностям та цілям Agile. В Agile-організації традиційні показники ефективності можуть втратити свою актуальність. Показники ефективності потрібно переорієнтувати, щоб вони відповідали цінностям і цілям Agile. Цей зсув означає зосередження більшої уваги на командній роботі та результатах, а не на індивідуальних завданнях та внесках. Такі показники, як швидкість роботи команди, досягнення цілей спринту та задоволеність клієнтів, можуть краще відображати гнучкі процеси. Важливо переконатися, що ці показники використовуються для управління та

вдосконалення процесів, а не як засіб індивідуального оцінювання. Впровадження збалансованої системи показників може допомогти в оцінці як кількісних, так і якісних аспектів роботи. Цей підхід дозволяє більш комплексно оцінити, наскільки добре команди адаптуються до гнучких методологій і сприяють досягненню загальних бізнес-цілей. Регулярний перегляд і коригування цих показників ефективності гарантує, що вони залишатимуться актуальними і відповідатимуть практикам Agile, що розвиваються.

5. Впровадження гнучких практик

- Пілотні проекти: Почніть з пілотних проектів, щоб застосувати Agile-практики в невеликих масштабах перед ширшим впровадженням. Впровадження Agile-практик часто виграє від поступового підходу, починаючи з пілотних проектів. Це невеликі, керовані проекти, де методології Agile можуть бути протестовані та вдосконалені в контрольованому середовищі. Вибір правильного пілотного проекту має вирішальне значення - він повинен бути достатньо складним, щоб забезпечити цінну інформацію, але не настільки критичним, щоб перші помилки мали значні негативні наслідки. Висновки з цих пілотних проектів є безцінними, оскільки вони дають реальний досвід і розуміння того, що працює, а що ні в конкретному контексті організації. Успіхи та виклики цих пілотних проектів мають бути ретельно задокументовані та проаналізовані, щоб удосконалити Agile-практики та підготуватися до ширшого впровадження. Такий поетапний підхід допомагає побудувати довіру та підтримку з боку решти організації, коли вони стають свідками відчутних переваг Agile-практик.

- Ітеративний підхід: Впроваджуйте ітеративні та інкрементальні процеси розробки. В основі Agile лежить ітеративний та інкрементний підхід. На відміну від традиційних методів, де ретельне планування передуює виконанню, Agile ділить проекти на менші, керовані частини, які називаються ітераціями або спринтом. Кожна ітерація включає цикл планування, виконання та оцінки, що дозволяє постійно вдосконалювати продукт і процес.

Такий підхід дозволяє командам швидко адаптуватися до змін, будь то зміни у вимогах клієнтів, ринкових умовах чи технологіях. Він також сприяє ранньому і частому наданню цінності клієнту, підвищуючи його задоволеність і забезпечуючи більш своєчасний зворотний зв'язок. Наголос на цьому підході під час впровадження Agile-практик допомагає командам дотримуватися принципів Agile і гарантує, що гнучкість лежить в основі розробки продукту.

- Механізми зворотного зв'язку: Впровадити механізми регулярного зворотного зв'язку з клієнтами та зацікавленими сторонами. Впровадження механізмів регулярного зворотного зв'язку є життєво важливим для процесу Agile. Agile підкреслює важливість постійного зворотного зв'язку як від клієнтів, так і від внутрішніх зацікавлених сторін, щоб гарантувати, що продукт або послуга, які розробляються, дійсно відповідають їхнім потребам та очікуванням. Такі практики, як огляди спринтів, демонстрації для клієнтів і сесии користувацького тестування, можуть забезпечити часті можливості для зворотного зв'язку. У середині команди ретроспектива наприкінці кожного спринту чи ітерації дає можливість команді поміркувати над своїми процесами та визначити сфери для вдосконалення. Цей зворотний зв'язок слід активно використовувати для інформування майбутніх ітерацій, гарантуючи, що продукт розвивається в правильному напрямку, а процеси команди постійно оптимізуються. Створення культури, в якій зворотний зв'язок не тільки заохочується, але й враховується, має вирішальне значення для підтримки гнучкості та клієнтоорієнтованості фреймворку Agile.

6. Вирішення проблем та подолання опору

- Управління опором: Визнавати та долати опір з боку працівників та зацікавлених сторін. Опір змінам є поширеною проблемою при впровадженні нових методологій, таких як Agile. Вкрай важливо розпізнавати та долати цей опір на випередження. Це починається з розуміння першопричин опору, які можуть варіюватися від страху перед невідомим до занепокоєння щодо безпеки роботи або дискомфорту від нових способів роботи. Відкрита, чесна комунікація є ключовим фактором у вирішенні цих проблем. Лідери та

чемпіони Agile повинні вести діалог зі співробітниками та зацікавленими сторонами, які чинять опір, вислуховувати їхні занепокоєння та надавати чітку, фактологічну інформацію про переваги та вплив Agile. Залучення працівників до процесу переходу, надання їм можливості ділитися своїми ідеями та відгуками також може допомогти зменшити опір. Важливо підкреслити особисті та професійні переваги Agile, такі як можливості для розвитку навичок, більша автономія та шанс працювати в більш динамічному та приємному робочому середовищі.

- Стратегії управління змінами: Застосовувати стратегії управління змінами, щоб полегшити перехідний період і вирішити проблеми. Ефективне управління змінами має вирішальне значення для плавного переходу до Agile. Це передбачає ретельне планування процесу змін, встановлення реалістичних очікувань і надання необхідної підтримки та ресурсів. Стратегії управління змінами можуть включати проведення комплексного навчання, створення системи підтримки за допомогою Agile-тренерів або наставників та впровадження поетапного підходу до переходу. Також корисно визначити та наділити повноваженнями внутрішніх агентів змін, які можуть пропагувати Agile та допомагати своїм колегам адаптуватися до нових процесів. Визнання та святкування швидких перемог на початку шляху до Agile може створити імпульс і продемонструвати відчутні переваги змін. Регулярний перегляд процесу змін і відкритість до коригування підходу на основі зворотного зв'язку гарантує, що перехід залишається узгодженим з потребами організації та працівників.

- Безперервне навчання та адаптація: Сприяти розвитку культури безперервного навчання та адаптації для вдосконалення практик Agile. Agile-трансформація - це не одноразова подія, а безперервний процес навчання та адаптації. Створення культури, яка цінує безперервне навчання, має важливе значення для стійкого успіху Agile-практик. Це передбачає надання можливостей для безперервної освіти та розвитку навичок, заохочення експериментів та інновацій, а також створення безпечного середовища, де

помилки розглядаються як можливості для навчання. Заохочення команд регулярно переглядати свої процеси та результати, а також повторювати свої практики допомагає впровадити безперервне вдосконалення в організаційну культуру. Лідери повинні моделювати таку поведінку, шукаючи зворотний зв'язок і демонструючи власну відданість навчанню та зростанню. Розвиваючи цю культуру, організація може постійно розвивати свої практики Agile, залишаючись чутливою до змін у бізнес-середовищі та гарантуючи, що Agile залишається рушійною силою для підвищення ефективності та досконалості.

7. Підтримка гнучкої трансформації

- Регулярне оцінювання: Проводити регулярні оцінки для визначення ефективності Agile-практик та внесення необхідних коректив.
- Спільнота практиків: Створіть в організації спільноту практиків Agile для обміну знаннями та досвідом.
- Довгострокові зобов'язання: Підкресліть, що Agile - це довгострокове зобов'язання, а не одноразова ініціатива.

Підготовка організації до Agile - це комплексний процес, який передбачає зміну культури, мислення та практик. Він вимагає відданості, навчання та готовності адаптуватися і розвиватися. Ретельно плануючи і виконуючи ці кроки, організації можуть успішно перейти на Agile-методологію і скористатися перевагами підвищеної гнучкості, поліпшеної співпраці та покращених результатів проєктів.

2.2 Кроки для гнучкої трансформації та виклики

Гнучка трансформація - це складний шлях, який вимагає ретельного планування та виконання. Цей шлях складається з кількох етапів, кожен з яких має свій унікальний набір викликів.

1. Створення бачення трансформації

- Визначення цілей: Чітко визначте цілі та завдання гнучкої трансформації.
 - Комунікація бачення: Донесіть бачення та переваги Agile до всіх рівнів організації.
 - Виклик: Подолання скептицизму та опору змінам з боку працівників та керівництва.
2. Розробка стратегічного плану
- Розробка дорожньої карти: Розробити покрокову дорожню карту трансформації.
 - Розподіл ресурсів: Розподіліть ресурси, включаючи час, бюджет і персонал, для процесу трансформації.
 - Виклик: збалансувати поточні бізнес-потреби з потребами в ресурсах, необхідних для трансформації.
3. Розвиток гнучкого лідерства
- Навчання лідерів: Навчити лідерів та менеджерів принципам Agile та лідерським навичкам.
 - Рольове моделювання: Заохочувати лідерів до моделювання гнучкої поведінки та практик.
 - Виклик: Перехід від традиційного командно-адміністративного лідерства до більш фасилітаційного стилю, який надає більше можливостей.
4. Створення гнучких команд
- Формування команди: Сформувати крос-функціональні команди з членами з різних відділів.
 - Навчання Agile: Забезпечити комплексне навчання Agile для всіх членів команди.
 - Виклик: Розвиток командної культури співпраці та подолання ізоляції всередині організації.
5. Впровадження гнучких практик
- Впровадження фреймворків: Прийняти відповідні фреймворки Agile, такі як Scrum, Kanban або XP.

- Ітеративний розвиток: Почніть з ітеративних циклів розробки та практики безперервного вдосконалення.
 - Виклик: Адаптація існуючих процесів і систем до гнучких методологій.
6. Забезпечення безперервного навчання та вдосконалення
- Цикли зворотного зв'язку: Налагодити регулярний зворотній зв'язок із зацікавленими сторонами та клієнтами.
 - Ретроспективи: Проводьте регулярні ретроспективи для безперервного навчання та вдосконалення.
 - Виклик: Створення культури, яка заохочує зворотний зв'язок, навчання на помилках і постійне вдосконалення.
7. Масштабування Agile по всій організації
- Розширення практик Agile: Поступово поширюйте практики Agile на більшу кількість команд та відділів.
 - Послідовність та інтеграція: Забезпечити узгодженість практик Agile в організації.
 - Виклик: Підтримка принципів і практик Agile в міру зростання масштабів трансформації.
8. Підтримка та розвиток гнучких практик
- Постійна підтримка: Надавати постійну підтримку та ресурси для Agile-команд.
 - Гнучкі метрики: Використовуйте Agile-метрики для вимірювання та відстеження прогресу.
 - Виклик: Зберегти динаміку Agile і не повернутися до старих звичок.
9. Вирішення проблем трансформації
- Опір змінам: Керуйте опором змінам за допомогою ефективної комунікації та залучення.
 - Культурні зміни: Враховуйте культурні зміни, необхідні для Agile, такі як посилення співпраці та відкритість до змін.

- Управління ресурсами: Балансування між потребою в досвідчених практиках Agile та навчанням нових членів команди.

Гнучка трансформація - це шлях, який вимагає чіткого бачення, стратегічного планування і прагнення до постійного вдосконалення. Хоча трансформація пов'язана з низкою викликів, зокрема культурними зрушеннями, опором змінам і потребою в постійній підтримці, їх можна подолати за допомогою ефективного лідерства, чіткої комунікації та відданості принципам Agile. В результаті організація стає більш гнучкою, оперативною та здатною до співпраці.

2.3 Досвід впровадження Agile в ТОВ “ГлобалЛоджик Україна”

Компанія GlobalLogic є сервісною компанією, тобто надає свої послуги іншим компаніям. Основні сервіси які надає компанії є сервіси з розробки програмного забезпечення. Під час надання таких сервісів компанія впроваджує гнучкі методи (Agile) розробки. Один із багатьох прикладів є надання послуг для компанії Avid, під час надання яких було використано Agile методологію для розробки та доставки програмного забезпечення.

Avid - світовий лідер у розробці рішень для індустрії створення та розповсюдження цифрових медіа. Протягом багатьох років багато з її відзначених нагородами продуктів, включаючи ProTools, Media Composer та рішення Avid Everywhere, визначили свої ринки.

Команда інженерів GlobalLogic, що базується в Києві, Україна, працювала в середовищі, яке відтворює американські потужності Avid, включаючи найсучасніші апаратні та програмні лабораторії, узгоджену структуру управління та спеціалізований персонал служби підтримки. Команди GlobalLogic органічно інтегровані в штат Avid і тісно пов'язані з усіма етапами життєвого циклу продукту Avid [19].

Результатом партнерства з Avid стали революційні рішення, продукти та послуги, які люди споживають щодня: фільми, відео, музичні записи, телевізійні шоу, живі концерти та випуски новин. Стратегічне бачення Avid, в розробку якого GlobalLogic зробила безпосередній внесок, охоплює весь ланцюжок створення вартості медіа - від способу створення та управління медіа до методів монетизації, розповсюдження та відстеження контенту. Усе це дає змогу професіоналам у сфері теле-, відео- та аудіоіндустрії налагоджувати зв'язок зі своєю аудиторією у більш потужний, ефективний, спільний та прибутковий спосіб.

Результати

- Економічна ефективність: До 50% скорочення витрат на робочу силу для Avid, економія до \$20 млн щорічно для інвестицій Avid у нові продукти та послуги.
- Час виходу на ринок: Прискорені дорожні карти для Avid, щоб продовжувати руйнувати галузь новими концепціями, технологіями, продуктами та послугами.
- Швидке розширення: Український інжиніринговий центр Avid збільшився в шість разів менш ніж за чотири роки. GlobalLogic підтримує згуртовану та високопродуктивну команду у співпраці з Avid
- Гнучкість: Швидке реагування на зміну пріоритетів завдяки гнучкій та потужній команді рекрутингу в Україні, яка гарантує, що інвестиції Avid в людей перетворюються на першокласні інженерні команди.
- Трансформація: Роль GlobalLogic у впровадженні Agile в індустрії програмного забезпечення допомогла перетворити Avid на високопродуктивну, розподілену Agile-організацію.

Нижче наведені рекомендації Девіда Гайгбаргера (Agile тренер) які використовує компанія GlobalLogic при впровадженні Agile.

1. Підтримка з боку вищого керівництва

Підтримка допомагає усунути перешкоди та встановити спільну мету, яка була чітко прокомунікована.

Доводиться працювати з клієнтами, де трансформація Agile відбувалася під керівництвом керівників середньої ланки. Це створювало свої перешкоди, оскільки змушує, Agile-тренерів, шукати підтримки серед різних команд вищого керівництва. Це стало більше схоже на гнучку трансформацію підмножин компанії замість того, щоб бути трансформацією самого підприємства. Добре відомо, що оптимізація підсистем не дорівнює оптимізації системи. Тобто, коли ви оптимізуєте лише меншу бізнес-одиницю, ця бізнес-одиниця (разом з іншими бізнес-одиницями) повинна буде знову змінитися, коли ви зосередитеся на впровадженні Agile в масштабах всієї компанії. Це включає в себе перегляд процесів, інструментів тощо, щоб краще підтримувати цей рівень трансформації.

2. Залишайтеся поза політикою всередині клієнта

Одна з найкращих речей у тому, чому залучається зовнішній консультант, полягає в тому, що ми можемо чесно сказати, що не належимо до жодної політичної фракції. Більшість Agile-коучів знають, що потрібно триматися подалі від політики і зберігати нейтралітет. Від цього залежить успіх Agile-коучів (і успіх клієнта).

Однак це не означає, що на Agile-коучів не впливає політика. Не один клієнт просить не розмовляти з певною групою - наприклад, з РМО - або навіть з іншою командою Agile-коучів, які працюють у тому ж офісі клієнта. Хоча Agile-коучі прагнуть успіху там, де можуть, такі вказівки можуть дуже розчарувати, і це гальмує зусилля з трансформації.

3. Agile-коучі повинні звітувати перед вищим керівництвом

Зміни повинні обговорюватися зі сторонами, яких вони стосуються, і в межах обмежень, з якими ми погоджуємося. Однак іноді Agile-коучам можуть перешкоджати, а то й прямо блокувати політично впливові люди. Якщо Agile-коуч не підпорядковується особі, яка займає досить високу посаду в організаційній структурі, то нам доводиться шукати альтернативи, що не є найкращим використанням часу Agile-коуча і, крім того, призводить до неоптимального прогресу в процесі трансформації.

4. Визнайте, що ми змінюємо компанію - це не діяльність за контрольним списком

З досвіду Agile-коучів, а також з численних дискусій з потенційними клієнтами, виявляється, що чимало компаній ставляться до процесу Agile-трансформації більше як до "контрольного списку". Наприклад, якщо у вас є проект, ви залучаєте проектних менеджерів. Після того, як ви їх залучили, ви можете викреслити цю діяльність зі свого списку справ і перейти до чогось іншого.

Agile-трансформація - це трансформація. Це трансформація того, як ви робите речі, як ви обговорюєте та вирішуєте проблеми, а також значні зміни в корпоративній культурі. Багато компаній залучають Agile-коуча, ставлять галочку, що вони завершили цю діяльність, а потім забувають про неї - за винятком періодичного контролю прогресу. Такий підхід ставить під сумнів ефективність роботи Agile-коучів і - якщо його не виправити - гальмує трансформацію.

5. Знати, чому клієнт проводить гнучку трансформацію

Це улюблене питання для будь-якого потенційного клієнта - чому ви хочете пройти через Agile-трансформацію? Існує широке розмаїттям відповідей.

Якщо коротко, то причини не мають великого значення, якщо їх можна усунути за допомогою Agile-трансформації. Існують відповіді, в яких, по суті, містилося прохання до Agile-трансформації вирішити інші проблеми, наприклад, змінити спосіб вимірювання ефективності роботи співробітників. Хоча Agile (і, наприклад, SAFe) дійсно обговорює тему того, як вимірювати прогрес працівника в гнучкому середовищі, це не повинно бути основною рушійною силою.

6. Використовуйте метрики, але робіть їх продуктивними

Дотримуючись старої приказки: "Люди реагують на те, як їх вимірюють", ви повинні використовувати метрики, які відображають цілі гнучкої трансформації. Існує багато метрик, які зосереджені на розробці,

таких як час циклу, дефекти, швидкість тощо. Незважаючи на те, що багато компаній хочуть пройти Agile-трансформацію, щоб реагувати на зміни на ринку, я ніколи не бачив, щоб це обговорювалося або вимірювалося, що я вважаю досить цікавим.

Так само важливо, як і те, які метрики ви використовуєте, також важливо не перестаратися з метриками. Я бачив, як люди використовують метрики як негнучкі огорожі. Наприклад, якщо команда має один дефект P1 (найсерйозніший) на виробництві, автоматично вважається, що це погана команда. Такий підхід створює культуру страху, яка є протилежною тому, чого ми хочемо досягти, і це досить серйозно вплине на продуктивність.

7. Пропагуйте Agile, але залишайтеся практичними

Agile - це не завжди рішення. В певних ситуаціях потрібно використовувати дуже індивідуальний підхід до Agile. Один із прикладів, що описує автор, це те що він зіштовхнувся зі структурою PMP (сертифікація управління проектами), яка була розроблена на основі процесів, що використовувалися для будівництва атомних підводних човнів. Створення чогось подібного вимагало б, щоб значна частина вимог була визначена і вимита заздалегідь.

Інший приклад це компанії, що працюють у високорегульованих галузях. Одним із прикладів було програмне забезпечення для медичної галузі, де і федеральний уряд, і кожен штат мають право голосу щодо того, що можна, треба, потрібно і не можна робити. У цій ситуації ви не можете просто зайти в кімнату з концепцією високого рівня і вийти з готовими до розробки історіями, як це пропонує робити один з фреймворків Agile.

8. Гнучку трансформацію слід впроваджувати поступово

Бажано поступово впроваджувати Agile-трансформацію. Хоча метою Agile-трансформації, як правило, є здатність реагувати на зміни на ринку (тобто вона потребує активного залучення бізнесу), нерідко (і це, як правило, хороша ідея) компанія може почати Agile-трансформацію з технологічних

команд першою. Запустіть цю частину, а потім залучайте інші підрозділи компанії.

Коли розглядається можливість гнучкої трансформації для вашого клієнта, важливо розуміти деякі елементи, щоб підвищити шанси на успіх, а також проблеми, на які слід звернути увагу.

Якщо дивитись на досліджуваний проект по розробці додатку для 5G мереж то він використовує Scrum фреймворк для свого процесу розробки. Scrum був обраний через його тісну відповідність динамічним та ітеративним потребам розробки додатку для управління мережею 5G. Цей фреймворк підтримує швидкі ітерації, гнучкість у роботі з мінливими вимогами до проекту та сильний акцент на безперервній комунікації та зворотному зв'язку, які є важливими для вирішення складних проблем, пов'язаних з технологіями 5G.

Обґрунтування застосування Scrum:

1. Гнучкість та адаптивність: Скрам дозволяє адаптувати продуктивний беклог у міру появи нових вимог, ідей та ринкових умов - критично важливий фактор при роботі з такими передовими технологіями, як 5G.

2. Покращена співпраця: Завдяки частим зустрічам Scrum, фреймворк сприяє кращій командній співпраці та швидшому вирішенню проблем, гарантуючи, що всі члени команди узгоджені з цілями проекту.

3. Зосереджується на створенні цінності: Scrum наголошує на швидкому та частому створенні функціонального програмного забезпечення, що має вирішальне значення для своєчасного тестування та доопрацювання програми в реальних мережевих сценаріях.

Планування ітерацій/спринту

1. Тривалість спринту: Кожен спринт у цьому проекті триває три тижні, що дає командам достатньо часу для розробки значущих функцій, водночас підтримуючи імпульс частих релізів та циклів зворотного зв'язку.

Процес планування спринту:

1. **Постановка цілей спринту:** Кожен спринт починається з планової зустрічі, на якій власник продукту представляє пріоритетні пункти з беклогу продукту, які необхідно вирішити. Команда співпрацює, щоб визначити мету спринту, яка дає чітке уявлення про те, чого має досягти спринт.

2. **Розбивка та оцінка завдань:** Потім завдання розбиваються на менші, керовані одиниці, і команда оцінює зусилля, необхідні для виконання кожного завдання, використовуючи сюжетні точки або години. Це допомагає ефективно розподіляти завдання та керувати робочим навантаженням.

3. **Планування потужностей:** Враховуючи тритижневу тривалість, команда оцінює свою спроможність виконувати завдання, переконуючись, що зобов'язання реалістичні та досяжні у встановлені терміни без шкоди для якості.

Спринт-виконання:

Щоденні наради проводяться для обговорення прогресу, усунення будь-яких перешкод і відповідного коригування робочого плану на день.

Постійна інтеграція та тестування допомагають виявити та вирішити проблеми на ранніх стадіях циклу, тим самим підвищуючи якість результатів.

Ролі та обов'язки команди

Склад гнучких команд:

Кожна з трьох команд розробників та команда з якості має власного Scrum Master та Product Owner, що забезпечує зосередження уваги як на управлінні процесом, так і на відстоюванні функцій продукту.

Scrum Master: Сприяє процесу Scrum, допомагає вирішувати перешкоди, гарантує, що всі практики дотримуються, і працює над тим, щоб команда була високопродуктивною.

Власник продукту: Відповідає за підтримання беклогу продукту, визначення пріоритетів на основі бізнес-цінності та цінності для клієнтів, а також за те, щоб команда завжди працювала над найефективнішими завданнями. Власник продукту також виступає в ролі ключової контактної особи із зацікавленими сторонами.

Розробники: Включає розробників програмного забезпечення, та тестувальників. Вони відповідають за виконання завдань, визначених у спринті, забезпечення високої якості розробки та досягнення цілей спринту.

Члени команди тестувальників: Зосереджуються на безперервному тестуванні під час спринтів. Вони тісно співпрацюють з розробниками, щоб гарантувати, що кожен реліз відповідає визначеним стандартам якості за допомогою автоматизованого та ручного тестування.

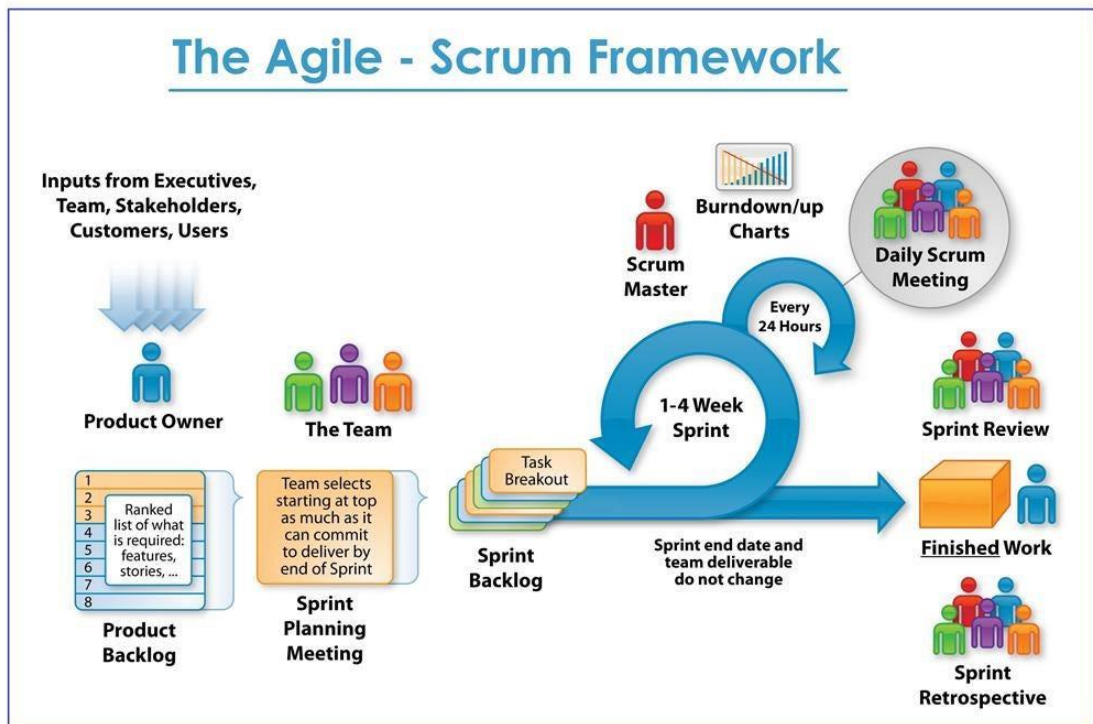


Рис 2.3.1 Гнучкий фреймворк розробки

Джерело [16]

РОЗДІЛ 3. МАЙБУТНЄ AGILE ТА РЕКОМЕНДАЦІЇ ЩОДО ПОКРАЩЕННЯ ПРОЕКТУ В ТОВ “ГЛОБАЛЛОДЖИК УКРАЇНА”

3.1 Нові тенденції в гнучкому управлінні проектами

Гнучке управління проектами постійно розвивається, адаптуючись до мінливого бізнес-середовища, технологічного прогресу та потреб організації. Ця еволюція призвела до появи кількох нових тенденцій, які формують майбутнє Agile.

1. Підвищений акцент на гнучкому масштабуванні

- Масштабування фреймворків: Такі фреймворки, як SAFe (Scaled Agile Framework), LeSS (Large-Scale Scrum) та DaD (Disciplined Agile Delivery) стають все більш популярними для масштабування Agile у великих організаціях.

- Цілісне впровадження Agile: Організації все частіше впроваджують Agile не лише в IT, але й у різних відділах.

- Виклик: Збереження мислення та принципів Agile під час масштабування.

2. Інтеграція з DevOps

- Конвергенція Agile-DevOps: Зростаючою тенденцією є інтеграція Agile з практиками DevOps для покращення співпраці між командами розробників та операційних команд.

- Безперервна доставка та інтеграція: Акцент на автоматизованому тестуванні, безперервній інтеграції та безперервній доставці для прискорення циклів випуску.

- Виклик: подолання розриву між швидкими циклами розробки та стабільною операційною діяльністю.

3. Зосередження на гнучкості бізнесу

- За межами IT: Принципи гнучкості застосовуються не лише в розробці програмного забезпечення, але й у таких сферах, як маркетинг, управління персоналом та фінанси.

- Організаційна гнучкість: Наголос на тому, щоб зробити всю організацію гнучкою, підвищити адаптивність до ринкових змін.

- Виклик: Адаптація Agile-методологій до контекстів, що не стосуються IT.

4. Посилене використання Agile-метрик

- Agile на основі даних: все більше використання метрик та аналітики для вимірювання та підвищення ефективності Agile.

- Прогностична аналітика: Використання аналітики даних для прийняття прогностичних рішень та управління ризиками.

- Виклик: Вибір правильних метрик, які точно відображають ефективність Agile, не підриваючи при цьому принципи Agile.

5. Віддалені та розподілені Agile-команди

- Глобальні команди: Оскільки віддалена робота стає все більш поширеною, методології Agile адаптуються для розподілених команд.

- Інструменти віртуальної співпраці: Використання передових інструментів і технологій для ефективної комунікації та співпраці між віддаленими командами.

- Виклик: Підтримка ефективної комунікації та командної динаміки у віртуальному середовищі.

6. Agile в проектах зі штучного інтелекту та машинного навчання

- AI та Agile: застосування методологій Agile в проектах штучного інтелекту та машинного навчання для роботи зі складністю та невизначеністю.

- Ітеративна розробка: Використання Agile для швидкого створення прототипів та ітеративної розробки в проектах AI/ML.

- Виклик: Балансування швидкості з етичними міркуваннями та складнощами, притаманними проектам AI/ML.

7. Збільшення залучення клієнтів і користувачів

- Клієнтоорієнтований підхід: Більше уваги приділяти залученню клієнтів і кінцевих користувачів до Agile-процесу для забезпечення постійного зворотного зв'язку.

- Дизайн користувацького досвіду (UX): Інтеграція принципів UX-дизайну в Agile для підвищення зручності користування продуктом і задоволеності клієнтів.

- Виклик: балансування між відгуками клієнтів і термінами та обсягом проекту.

У Додаток Г представлена візуалізація розвитку Agile до 2030 року, для генерації графіку використаний Штучний Інтелект.

Майбутнє гнучкого управління проектами відзначається виходом за традиційні межі, інтеграцією з іншими методологіями, такими як DevOps, та адаптацією до викликів сучасного бізнес-середовища, що швидко змінюється. Ці нові тенденції вказують на перехід до більш комплексного, загальноорганізаційного застосування принципів Agile, з акцентом на постійному вдосконаленні, адаптивності та клієнтоорієнтованості.

Гнучкі методології стали важливим фасилітатором цифрової трансформації та Індустрії 4.0, впливаючи на те, як організації адаптуються до швидкозмінного цифрового ландшафту. У цьому підрозділі досліджується роль Agile в цьому контексті.

1. Agile як каталізатор цифрової трансформації

- Швидка адаптація: Ітеративний та гнучкий характер Agile дозволяє організаціям швидко адаптуватися до нових технологій та ринкових змін.

- Інновації, орієнтовані на клієнта: Agile надає пріоритет зворотному зв'язку з клієнтами та користувацькому досвіду, що є ключовими компонентами успішних цифрових продуктів та послуг.

- Виклик: Інтеграція Agile у традиційно не гнучкі сектори, що перебувають у процесі цифрової трансформації.

2. Agile в контексті Індустрії 4.0

- Визначення Індустрії 4.0: Індустрія 4.0 - це сучасна тенденція автоматизації та обміну даними у виробничих технологіях, включаючи кібер-фізичні системи, Інтернет речей (IoT), хмарні обчислення та когнітивні обчислення.

- Гнучке виробництво: Гнучкі методології застосовуються у виробництві для підвищення гнучкості, скорочення часу виходу на ринок та покращення кастомізації продукції.

- Виклик: Застосування Agile у високоструктурованих і традиційно повільно змінюваних промислових середовищах.

3. Посилення співпраці в цифрову епоху

- Крос-функціональні команди: Agile сприяє міжфункціональній співпраці, що має важливе значення для вирішення складних проектів цифрової трансформації.

- Подолання ізоляції: Agile допомагає подолати організаційну ізоляцію, сприяючи кращій комунікації та співпраці в цифрових ініціативах.

- Виклик: Сприяння розвитку культури співпраці у великих, географічно розподілених організаціях.

4. Agile та Інтернет речей (IoT)

- Гнучка розробка IoT: Agile добре підходить для розробки додатків IoT через необхідність швидкого створення прототипів та ітеративного тестування в таких проектах.

- Адаптивна адаптація: Гнучкі методології дозволяють швидко адаптуватися до змін і нових відкриттів у розробці IoT-проектів.

- Виклик: Управління складністю та проблемами безпеки, притаманними розробці IoT.

5. Гнучкість у хмарних та когнітивних обчисленнях

- Хмарна гнучкість: Гнучкі методології підтримують гнучкий і масштабований характер хмарних обчислень.

- Проекти когнітивних обчислень: Гнучкі підходи сприяють експериментальному та ітеративному характеру проектів когнітивних обчислень.

- Виклик: збалансувати швидкі темпи розвитку хмарних і когнітивних обчислень з ретельним тестуванням і заходами безпеки.

6. Роль Agile у безперервному навчанні та інноваціях

- Культура навчання: Agile сприяє розвитку культури безперервного навчання та експериментів, що має вирішальне значення для цифрової трансформації та Індустрії 4.0.

- Інноваційні цикли: Короткі спринти та ітеративні цикли в Agile сприяють швидкому впровадженню інновацій та тестуванню нових ідей.

- Виклик: Створення середовища безперервних інновацій при збереженні операційної стабільності.

Гнучкі методології відіграють ключову роль у цифровій трансформації та розвитку Індустрії 4.0. Сприяючи швидкій адаптації, інноваціям, орієнтованим на клієнта, та міжфункціональній співпраці, принципи Agile допомагають організаціям орієнтуватися в складнощах цифрової епохи. Серед викликів - інтеграція Agile у традиційні галузі, управління технологічними складнощами та підтримка культури безперервних інновацій.



Рис 3.2. Візуалізація складових індустрії 4.0

Джерело: [17]

Оскільки гнучкі методології продовжують розвиватися, вони формують майбутнє управління проектами та організаційними операціями. У цьому підрозділі розглядаються прогнози та майбутні напрямки розвитку гнучких методологій, висвітлюються потенційні тенденції та розробки.

1. Розширення Agile за межі ІТ

- Ширше застосування: Agile, ймовірно, пошириться за межі ІТ на різні бізнес-функції, такі як маркетинг, управління персоналом та стратегічне планування.

- Організаційна гнучкість: Все більше організацій впроваджуватимуть принципи Agile на стратегічному рівні, що призведе до більш гнучких та оперативних бізнес-практик.

- Прогнозування: Принципи Agile стануть невід'ємною частиною всіх бізнес-операцій, а не лише управління проектами.

2. Гнучкість та нові технології

- Інтеграція зі штучним інтелектом і машинним навчанням: гнучкі методології все більше перетинатимуться зі штучним інтелектом (ШІ) і машинним навчанням (МН), покращуючи прогнозний аналіз і процеси прийняття рішень в управлінні проектами.

- Блокчейн та Agile: технологія блокчейн може інтегруватися з методологіями Agile для підвищення прозорості та децентралізованого прийняття рішень.

- Прогноз: Agile буде розвиватися, щоб включати та адаптуватися до нових технологій, підвищуючи ефективність та інновації.

3. Посилений фокус на клієнтському досвіді

- Клієнтоорієнтована гнучкість: Фокус буде зосереджений на наданні цінності для клієнта, з більшою кількістю практик, спрямованих на розуміння і задоволення потреб та уподобань клієнта.

- Дизайн взаємодії з користувачем (UX): Гнучкі методології все більше включатимуть принципи UX-дизайну, щоб продукти були не лише функціональними, але й зручними та цікавими для користувачів.

- Прогнозування: Гнучкі методології будуть тісніше переплітатися зі стратегіями клієнтського досвіду, акцентуючи увагу на зворотному зв'язку та залученні користувачів.

4. Еволюція гнучкого лідерства та культури

- Гнучке лідерство: Зростатиме акцент на лідерах, які можуть сприяти розвитку Agile-культури, зосереджуючись на розширенні можливостей, співпраці та постійному вдосконаленні.

- Культурні зміни: Організації надаватимуть пріоритет створенню культури, яка повністю охоплює цінності Agile, включаючи відкритість до змін та зосередження на командній співпраці.

- Прогнозування: Роль лідерства та організаційної культури в успіху Agile стане більш помітною.

5. Досягнення в галузі Agile-інструментів та метрик

- Інновації інструментів: Розробка більш досконалих інструментів для відстеження Agile-проектів, що включають аналітику в реальному часі та можливості прогнозування.

- Еволюція метрик: Гнучкі метрики розвиватимуться, щоб точніше вимірювати і підтримувати безперервне вдосконалення, продуктивність команди і задоволеність клієнтів.

- Прогнозування: Agile-інструменти та метрики стануть більш досконалими та невід'ємною частиною Agile-практик, що сприятиме підвищенню ефективності та результативності.

6. Гнучкість у віддаленому та гібридному робочому середовищі

- Віддалені Agile-команди: Оскільки віддалені та гібридні робочі середовища стають все більш поширеними, гнучкі методології будуть адаптуватися для підтримки ефективної співпраці та управління проектами в цих умовах.

- Віртуальна співпраця: Використання інструментів і методів віртуальної співпраці стане більш досконалим і невід'ємною частиною Agile-практик.

- Прогнозування: Гнучкі методології будуть розвиватися, щоб відповідати викликам і використовувати можливості віддалених і гібридних робочих середовищ.

7. Сталий розвиток та Agile

- Гнучкість для сталого розвитку: Гнучкі методології будуть все частіше використовуватися для вирішення проблем сталого розвитку, включаючи принципи соціальної відповідальності та впливу на навколишнє середовище в планування та виконання проектів.

- Етичні гнучкі практики: Більший акцент на етичних міркуваннях в управлінні проектами Agile, узгодження практик Agile з більш широкими соціальними та екологічними цілями.

- Прогнозування: Гнучкі методології відіграватимуть значну роль у розвитку сталих та етичних бізнес-практик.

8. Персоналізація гнучких практик

- Індивідуальні фреймворки Agile: Організації все більше адаптуватимуть фреймворки Agile до своїх унікальних контекстів і потреб, відходячи від універсальних рішень.

- Гнучкі методології: Більша гнучкість у застосуванні принципів Agile, що дає можливість для більшої креативності та інновацій у їх впровадженні.

- Прогнозування: Гнучкі методології стануть більш персоналізованими та адаптованими до конкретних організаційних контекстів та викликів.

Майбутнє Agile-методологій є динамічним і багатообіцяючим, воно характеризується більш широким застосуванням у різних галузях, інтеграцією з новими технологіями, а також більш глибоким фокусом на клієнтському досвіді, стійкості та організаційній культурі. Оскільки Agile продовжує розвиватися, він буде пропонувати більш досконалі інструменти та підходи для задоволення мінливих потреб бізнесу та проектів, забезпечуючи свою актуальність та ефективність у світі, що швидко змінюється.

3.2 Аналіз Agile проекту компанії ТОВ "ГлобалЛоджик Україна"

В якості компанії для аналізу була вибрана компанія ТОВ "ГлобалЛоджик Україна" та проект ТОВ "ГлобалЛоджик Україна" спрямований на розробку передового додатку для управління мережами 5G. Оскільки телекомунікаційна галузь рухається в напрямку більш досконалих і надійних послуг, потреба в надійних рішеннях для управління мережею стала першочерговою. Цей проект спрямований на використання сучасних ІТ-технологій для створення додатку, який може впоратися зі складнощами та вимогами технології 5G. Мета полягає в тому, щоб надати інструмент, який підвищує ефективність мережі, покращує надійність послуг і дозволяє телекомунікаційним операторам пропонувати кінцевим користувачам послуги вищої якості.

Значення цього проекту в галузі телекомунікаційних OSS (Operational Support System) не можна недооцінювати. Системи операційної підтримки мають вирішальне значення для управління мережевими компонентами, забезпечення якості послуг та підтримки клієнтів - все це є невід'ємною частиною успіху телекомунікаційних операцій в епоху 5G. Розробляючи додаток, спеціально для мереж 5G, ГлобалЛоджик позиціонує себе в авангарді телекомунікаційних інновацій, задовольняючи зростаючі потреби операторів зв'язку та сприяючи досягненню більш широких цілей цифрової трансформації в галузі.

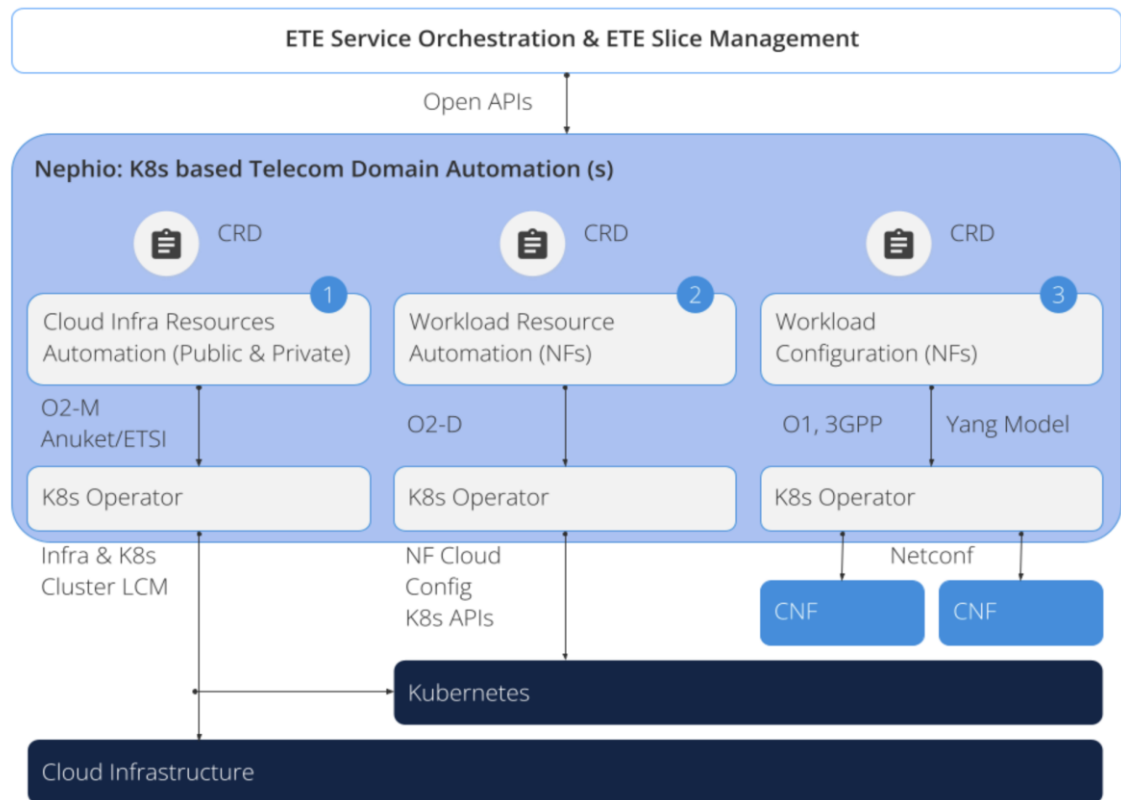


Рис 3.2.1 Орієнтовна архітектура рішення.

Джерело: [33]

У проєкті використовується надійний технологічний стек, розроблений з урахуванням високих вимог до масштабованості, продуктивності та надійності, характерних для сучасних телекомунікаційних додатків.

1. Мікросервісна архітектура: Додаток побудований на основі мікросервісної архітектури, яка розділяє весь додаток на невеликі, незалежно розгорнуті сервіси. Кожна послуга розроблена навколо конкретної бізнес-можливості і може бути розроблена, розгорнута і масштабована незалежно. Ця архітектура підвищує гнучкість і стійкість програми, дозволяючи оновлювати окремі сервіси без впливу на всю систему, що має вирішальне значення для постійного вдосконалення та обслуговування в динамічному середовищі, такому як управління мережею 5G.

2. Мова програмування Go: Go, також відома як Golang, була обрана в якості основної мови програмування для цього проєкту завдяки своїй простоті, ефективності та відмінній підтримці паралельних операцій та мікросервісів. Потужна стандартна бібліотека мови та ефективність під час

виконання є особливо вигідними для створення високопродуктивних мережеских додатків, які потребують обробки великих обсягів даних з мінімальною затримкою.

3. Kubernetes: Для ефективного управління мікросервісами використовується Kubernetes як платформа для оркестрування контейнерів. Вона автоматизує розгортання, масштабування та роботу контейнерів додатків на кластерах хостів. Kubernetes розширює можливості проекту з управління мікросервісами в масштабі, гарантуючи, що додаток може надійно обслуговувати мережеских операторів навіть при змінному навантаженні, а також підтримує розгортання та відкат без простоїв, що є важливим для роботи мережі в режимі 24/7.

Інструменти для управління проектами та співпраці:

1. JIRA: JIRA є центральним інструментом, що використовується для управління проектами, відстеження помилок, відстеження проблем та оперативних проектних завдань. Він надає командам можливість планувати спринти, відстежувати хід виконання завдань та керувати змінами з повною видимістю процесу розробки. Кастомні робочі процеси в JIRA адаптовані до методології Scrum, прийнятої командами, підтримуючи гнучкі практики, такі як відставання та планування спринтів.

2. Конфлюенс: Confluence слугує базою знань проекту і використовується для зберігання всієї документації, специфікацій проекту та нотаток про зустрічі. Це забезпечує безперешкодну співпрацю між членами команди, дозволяючи їм обмінюватися ідеями, прогресом проекту та результатами досліджень. Confluence тісно інтегрується з JIRA, пов'язуючи вимоги та обговорення безпосередньо із завданнями розробки, тим самим підтримуючи цілісне розуміння цілей та деталей проекту.

3. Jenkins: Використовується для безперервної інтеграції та доставки, Jenkins автоматизує процес тестування та розгортання коду. Створюючи конвеєри в Jenkins, проект гарантує, що нові зміни коду, подані розробниками, послідовно і автоматично збираються, тестуються і готуються

до запуску в експлуатацію або виробниче середовище без ручного втручання. Ця практика підтримує високі стандарти розробки та швидкі ітерації, які мають вирішальне значення для гнучкої розробки.

4. **Docker:** Docker використовується для контейнеризації окремих мікросервісів, гарантуючи, що кожен сервіс може працювати незалежно в будь-якому середовищі. Це допомагає Kubernetes ефективно розгортати та керувати мікросервісами, забезпечуючи узгоджене середовище від розробки до виробництва.

5. **Prometheus та Grafana:** Для моніторингу мікросервісів Prometheus використовується для збору метрик, а Grafana - для візуалізації цих метрик. Разом вони надають потужні можливості моніторингу, які дозволяють в режимі реального часу спостерігати за робочим станом додатку, допомагаючи виявляти та вирішувати потенційні проблеми до того, як вони вплинуть на роботу сервісу.

The image shows the logos for Jira and Confluence. Jira is represented by a blue icon of three arrows pointing up and right, followed by the word "Jira" in a bold, dark blue font. Confluence is represented by a blue icon of two crossed arrows, followed by the word "Confluence" in a bold, dark blue font.

Jenkins

Рис 3.2.1 Інструменти, що використовуються

Джерело: самостійно розроблений рисунок

У проекті використовується матрична структура для управління розробкою додатку. Ця організаційна структура об'єднує функціональні та проектні обов'язки в рамках всієї організації, сприяючи спільному використанню ресурсів та міждисциплінарній співпраці.

Впровадження матричної структури:

- Функціональні та проектні осі: Матрична структура в цьому контексті поєднує дві осі управління - функціональне (вертикальне) управління та управління проектами (горизонтальне). Функціональні менеджери зосереджуються на технічній експертизі та управлінні ресурсами в межах конкретних відділів, таких як розробка, забезпечення якості та операційна діяльність. Керівники проектів, з іншого боку, відповідають за нагляд за термінами, результатами та міжфункціональною інтеграцією проектів.

- Взаємодія між ролями: Кожен член команди звітує перед двома керівниками: функціональним менеджером і менеджером проекту. Наприклад, розробник може звітувати керівнику відділу розробки з питань кар'єрного розвитку та технічного консультування, а також звітувати керівнику проекту щодо завдань і цілей, пов'язаних з проектом.

Переваги матричної структури

- Гнучкість та ефективність використання ресурсів: Матрична структура дозволяє ефективно розподіляти ресурси, коли працівники можуть бути призначені на різні проекти за потреби, не обмежуючись одним відділом. Така гнучкість допомагає оптимізувати використання ресурсів, зменшуючи час простою та дублювання зусиль.

- Покращена комунікація та співпраця: Заохочуючи членів команди працювати за різними функціями, матрична структура сприяє високому рівню комунікації та розширює сферу співпраці. Це особливо корисно в таких складних проектах, як розробка додатку для 5G, де інтеграція різних технологій і підходів має вирішальне значення.

- Підвищена адаптивність: Подвійне підпорядкування в матриці допомагає організації більш гнучко реагувати на зміни та виклики. Керівники проектів можуть швидко перерозподіляти ресурси для вирішення нових пріоритетів або вузьких місць проекту, підвищуючи здатність організації адаптуватися до нових вимог або несподіваних викликів.

- Ширший розвиток співробітників: Робота за такою структурою дає працівникам можливість розвивати ширший спектр навичок і знань, оскільки вони взаємодіють з багатьма менеджерами та проектами. Така диверсифікація може підвищити задоволеність роботою та професійне зростання, що призводить до формування більш всебічно розвинених працівників.

Виклики матричної структури

- Складність в управлінні: Однією з головних проблем матричної структури є складність подвійного підпорядкування, що іноді може призвести до плутанини та конфліктів щодо пріоритетів чи ресурсів. Працівникам може бути складно збалансувати вимоги та очікування від кількох керівників.

- Потенціал для конфлікту: Перетин повноважень може призвести до боротьби за владу або конфлікту між функціональними та проектними менеджерами, особливо якщо їхні цілі та методи відрізняються. Це вимагає ретельного управління та чіткого розподілу ролей і обов'язків.

- Комунікаційне перевантаження: Хоча матрична структура сприяє комунікації, збільшення кількості взаємодій іноді може призвести до інформаційного перевантаження членів команди, що ускладнює виділення критично важливої інформації з рутинної комунікації.

- Затримки у прийнятті рішень: Подвійне підпорядкування може ускладнити процес прийняття рішень, що потенційно може призвести до затримок. Забезпечення ефективного прийняття рішень вимагає чітких процесів та чітко визначених ролей, щоб уникнути вузьких місць.

У проекті матрична структура відіграє вирішальну роль у використанні міжфункціональних можливостей для вирішення завдань розробки високотехнологічних рішень з управління мережею 5G. Хоча структура пропонує значні переваги з точки зору гнучкості ресурсів та ширшого розвитку навичок, вона також вимагає ретельного управління, щоб пом'якшити складнощі та потенційні конфлікти. Встановлення чітких інструкцій і каналів комунікації, а також постійне навчання з вирішення

конфліктів і управління командою можуть допомогти максимізувати переваги матричної структури, мінімізуючи при цьому її виклики.

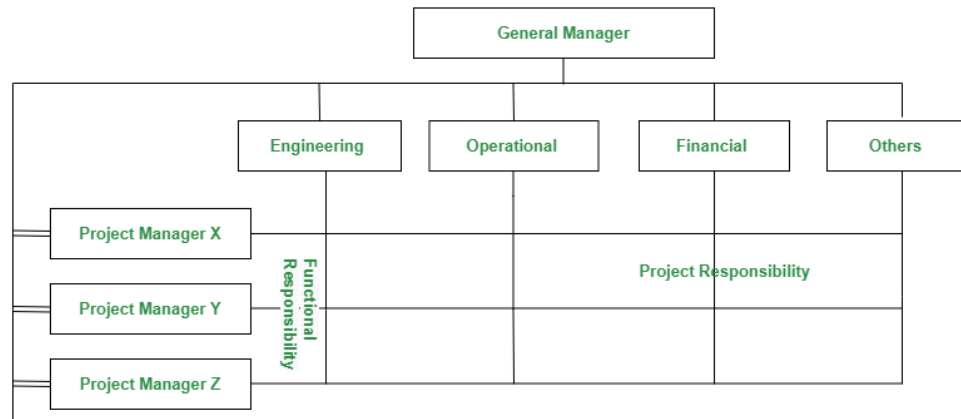


Рис 3.2 Візуалізація матричного управління проектом

Джерело: [21]

Процес ідентифікації ризиків:

На проекті управління ризиками є критично важливим компонентом, інтегрованим у життєвий цикл проекту, щоб забезпечити успішну реалізацію програми 5G OSS. Процес виявлення ризиків включає кілька ключових стратегій:

- Консультації із зацікавленими сторонами: Регулярні зустрічі із зацікавленими сторонами, включаючи операторів мереж, спонсорів проекту і кінцевих користувачів, для збору інформації про потенційні ризики з різних точок зору.
- Експертні оцінки: Використання досвіду технічних керівників і галузевих експертів для аналізу планів і архітектури проекту з метою виявлення потенційних технічних і операційних ризиків.
- Вивчені уроки: Врахування відгуків та уроків, винесених з попередніх проектів компанії, особливо тих, що стосуються подібних технологій або методологій.

- Семінари з питань ризиків: Проведення спеціальних семінарів з ідентифікації ризиків з проектними командами для мозкового штурму потенційних ризиків на основі обсягу проекту, технології та ринкового середовища.

Стратегії зменшення ризиків:

Після виявлення ризиків застосовується структурований підхід до їхнього зменшення, який включає

- Превентивні заходи: Впровадження заходів, спрямованих на запобігання виникненню ризиків, таких як впровадження перевірених технологій, проведення ретельного аналізу ринку та посилення заходів безпеки.

- Планування на випадок непередбачених обставин: Розробка планів на випадок ризиків, які неможливо повністю запобігти. Сюди входять альтернативні стратегії, додатковий розподіл ресурсів і створення буферів у розкладі.

- Відповідальність за ризики: Розподіл відповідальності за управління конкретними ризиками між відповідними членами команди, забезпечення того, щоб зменшення ризиків було безперервним і контрольованим процесом.

- Регулярний аналіз ризиків: Планування регулярних оглядів ландшафту ризиків протягом життєвого циклу проекту для коригування стратегій зниження ризиків у відповідь на нову інформацію або зміни в проектному середовищі.

Інструменти та технології для управління ризиками:

- Програмне забезпечення для управління ризиками: Інструменти на кшталт Реєстру ризиків, які допомагають систематично відстежувати та оцінювати ризики.

- Інструменти управління проектами: Інтеграція функцій управління ризиками з інструментами управління проектами, такими як JIRA, для моніторингу ризиків паралельно з прогресом проекту.

- Програмне забезпечення для моделювання: Використання інструментів моделювання для моделювання потенційних наслідків виявлених ризиків та ефективності стратегій пом'якшення ризиків.

Технічні ризики та їх вплив:

- Інтеграційні збої: З огляду на складну інтеграцію, необхідну для систем 5G OSS з існуючими телекомунікаційними інфраструктурами, збої в інтеграції становлять значний ризик. Такі збої можуть затримати терміни реалізації проекту і вплинути на загальну якість і продуктивність результату.

- Застарілість технологій: Швидкий технологічний прогрес може зробити обрані технології або підходи застарілими, що вплине на актуальність і конкурентоспроможність проекту.

Операційні ризики та їх вплив:

- Доступність ресурсів: Наявність кваліфікованих ресурсів, особливо у сфері новітніх технологій, таких як 5G, є критичним ризиком. Дефіцит або висока плинність кадрів може затримати цикли розробки та вплинути на графіки випуску продукції.

- Відповідність та регуляторні зміни: Зміни в телекомунікаційному законодавстві або невідповідність стандартам можуть призвести до затримок проекту, додаткових витрат або навіть до відмови від певних функцій або можливостей.

Ринкові ризики та їх вплив:

- Прийняття ринком: Ризик того, що ринок не прийме новий додаток так, як очікувалося, може суттєво вплинути на успіх проекту та майбутні етапи розробки.

- Конкурентний тиск: Поява нового конкурента або раптова зміна конкурентних стратегій може вплинути на ринкове позиціонування проекту і призвести до незапланованих змін в обсязі або функціях проекту.

Стратегічне реагування на вплив ризиків:

Для ефективного реагування на ці ризики проект зосереджується на стратегічному плануванні, надійних практиках управління проектами та

гнучких механізмах реагування. Команда управління проектами тісно співпрацює з усіма зацікавленими сторонами, щоб забезпечити своєчасне прийняття рішень та коригування відповідно до мінливих обставин, таким чином мінімізуючи негативний вплив на результати та строки виконання проекту.

Завдяки комплексному управлінню ризиками проект прагне захистити проект від потенційних негативних впливів, гарантуючи, що додаток буде доставлений вчасно, в рамках бюджету та відповідатиме або перевищуватиме очікування щодо продуктивності. Такий проактивний підхід до виявлення та пом'якшення ризиків є наріжним каменем стратегії проекту для досягнення його цілей та сприяння зростанню компанії в конкурентному телекомунікаційному секторі.

3.3 Пропозиції щодо покращення в управлінні проектом

Оптимізація планування спринту:

- Сесія попереднього планування: Проведіть сесію попереднього планування з власником продукту та кількома ключовими членами команди, щоб уточнити бэклог перед плануванням спринту всієї команди. Це забезпечить більш чітке бачення та більш сфокусоване обговорення під час фактичної зустрічі з планування [1, 2].

- Тайм-боксинг: Суворо дотримуйтеся часових обмежень для кожної частини планування спринту, щоб обговорення були стислими і точними. Це може включати встановлення конкретних часових рамок для обговорення кожної користувачької історії або завдання [1, 2].

- Чіткі цілі та результати: Починайте кожну сесію планування спринту з чіткого формулювання цілей та очікуваних результатів. Це вирівнює зусилля команди з самого початку і посилює фокус на критично важливих цілях [1, 2].

Покращення щоденних зустрічей:

- Сфокусовані оновлення: Обмежте дискусії на стенд-апах обговоренням прогресу, негайних наступних кроків і блокуючих факторів. Не рекомендуйте вирішувати проблеми або проводити тривалі дискусії під час стенд-апу, щоб він не перевищував 15 хвилин [3].
- Ротація фасилітатора: Ротуйте роль фасилітатора між членами команди, щоб сприяти залученню та забезпечити свіжі перспективи у проведенні стенд-апів [4].
- Використання технологій: Використовуйте інструменти управління проектами, щоб під час стендапів виводити на екран дошку спринту команди, надаючи візуальні підказки, які допомагають команді тримати себе в тонусі і зосереджуватися на цілях спринту [3].

Покращення відгуків про спринт:

- Структурований фідбек: Запровадьте структурований формат для збору відгуків під час огляду спринтів. Підготуйте конкретні запитання або опитування для стейкхолдерів, щоб зворотній зв'язок був всебічним та дієвим [1].
- Підготовка демонстраційних версій: Заохочуйте команди готувати демонстрації або презентації своєї роботи перед оцінюванням. Це сприяє підвищенню відповідальності та підзвітності, а також покращує зрозумілість представленої роботи [2].
- Залучення зацікавлених сторін: Докладіть спільних зусиль, щоб залучити різні зацікавлені сторони до перегляду, включаючи команди маркетингу, продажів і підтримки клієнтів, щоб зібрати різні точки зору і отримати більш повний зворотній зв'язок [4].

Вдосконалення ретроспективи спринту:

- Практичні висновки: Зосередьтеся на визначенні 2-3 дієвих пунктів з кожної ретроспективи для реалізації в наступному спринті. Це забезпечить постійне вдосконалення, не перевантажуючи команду надто великою кількістю змін одночасно [5].

- Тематичні ретроспективи: Час від часу визначайте теми для ретроспективи, щоб зосередитися на конкретних сферах, таких як співпраця, технології або процеси [4].

- Святкування успіхів: Візьміть за звичку визнавати і святкувати успіхи під час ретроспективи, щоб підтримувати моральний дух і мотивувати команду [5].

Таблиця 3.3.1

Потенційні заощадження від впровадження покращень в Agile проект

| Область покращення | Метричний показник/характеристика | Поточний стан | Покращений стан | Оціночні заощадження (USD) |
|----------------------------------|--|---------------|-----------------|----------------------------|
| Оптимізація планування спринтів | Час, витрачений на планування спринтів (год/тиждень) | 6 | 4 | \$2,400 |
| | Кількість планувальних сесій за спринт | 1 | 2 | \$500 |
| | Вирівнювання та фокус команди (%) | 75 | 90 | \$1,000 |
| Покращення щоденних стендапів | Час, витрачений на стендапи (хв/день) | 20 | 15 | \$1,200 |
| | Залучення команди до стендапів (%) | 70 | 85 | \$800 |
| Покращення спринт-оглядів | Час, витрачений на огляди (год/спринт) | 3 | 2 | \$1,000 |
| | Якість зворотного зв'язку від стейкхолдерів (%) | 60 | 80 | \$1,500 |
| Покращення ретроспектив спринтів | Дієві інсайти за ретроспективу | 1 | 2 | \$800 |
| | Мораль та мотивація команди (%) | 65 | 85 | \$1,200 |
| Загальні заощадження | | | | \$10,400 |

Джерело: самостійно розроблені обчислення

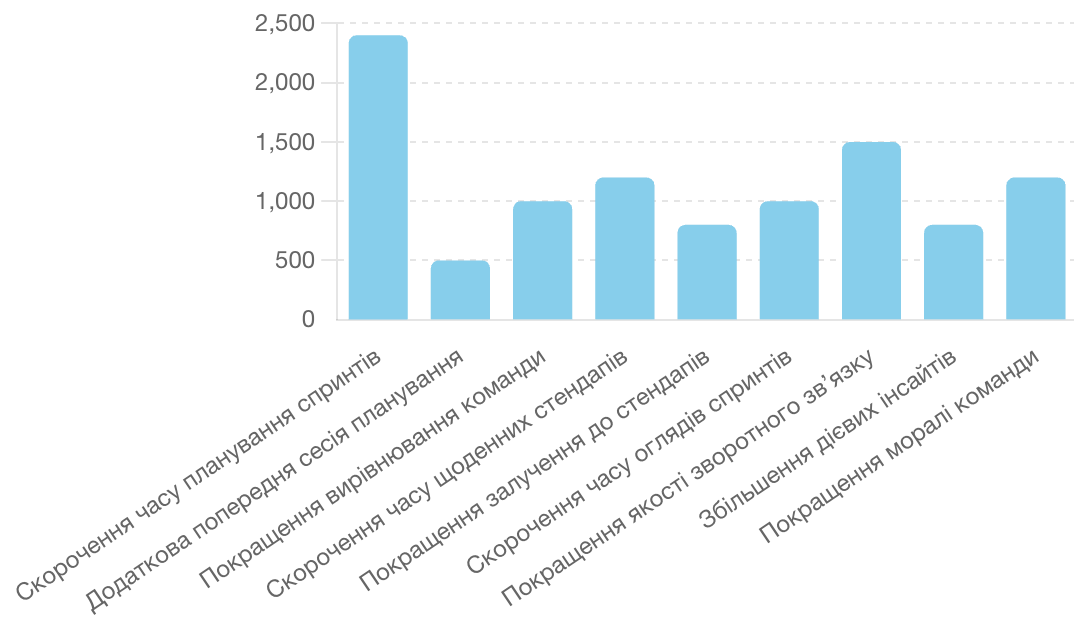


Рис. 3.3.1. Потенційні заощадження від впровадження покращень в Agile процес.

Джерело: самостійно розроблений графік

Припущення та розрахунки:

1. Оптимізація планування спринтів:

- Скорочення часу планування спринтів з 6 до 4 годин заощаджує 2 години на тиждень. Припускаючи, що команда складається з 24 членів (3 команди по 8 членів) і середня погодинна ставка \$50, заощадження складають: $2 \text{ години/тиждень} * 24 \text{ члени} * \$50/\text{година} = \$2,400$.

- Додаткова попередня планувальна сесія коштує додатково 1 годину, але покращує вирівнювання та фокус команди, оціночні заощадження через покращення ефективності: \$500.

- Збільшення вирівнювання та фокусу команди з 75% до 90% веде до покращення продуктивності, оціночні додаткові заощадження: \$1,000.

2. Покращення щоденних стендапів:

- Скорочення часу стендапів з 20 до 15 хвилин заощаджує 5 хвилин на день. Припускаючи 5 днів стендапів на тиждень, 24 члени і \$50/година: $5 \text{ хвилин/день} * 5 \text{ днів} * 24 \text{ члени} * \$50/\text{година} = \$1,200$.

- Покращення залучення веде до кращого вирішення проблем і продуктивності команди, оціночні додаткові заощадження: \$800.

3. Покращення спринт-оглядів:

- Скорочення часу оглядів з 3 до 2 годин за спринт заощаджує 1 годину. Припускаючи в середньому 2 спринти на місяць, 24 члени і \$50/година: $1 \text{ година/спринт} * 2 \text{ спринти/місяць} * 24 \text{ члени} * \$50/\text{година} = \$1,000$.

- Покращення якості зворотного зв'язку веде до кращих результатів продукту, оціночні додаткові заощадження: \$1,500.

4. Покращення ретроспектив спринтів:

- Збільшення кількості дієвих інсайтів з 1 до 2 за ретроспективу веде до більш ефективних змін. Припускаючи, що ретроспективи проводяться кожен спринт і кожен дієвий інсайт заощаджує 1 годину роботи команди за спринт: $1 \text{ година} * 3 \text{ команди} * \$50/\text{година} = \$800$.

- Підвищена мораль та мотивація команди, веде до покращення продуктивності, оціночні додаткові заощадження: \$1,200.

Загальні оціночні заощадження при впровадженні цих покращень складають \$10,400.

Коригування ролей

Коригування ролей у скрам-команді:

- Розширення повноважень Скрам-майстра: Посилити роль Скрам-майстра як коуча та фасилітатора, а не лише як адміністратора процесу. Надайте Скрам-майстрам тренінги з вирішення конфліктів, коучингу та лідерства, щоб підвищити їхню ефективність [1].

- Посилення співпраці з власниками продукту: Заохочуйте власників продукту проводити більше часу з командами розробників, не обмежуючись лише церемоніями. Це може включати участь у відповідних технічних обговореннях та неформальних зустрічах команди, щоб краще зрозуміти виклики та нюанси процесу розробки[1].

- Перехресне навчання розробників: Сприяйте перехресному навчанню розробників, щоб сформувати більш універсальну команду. Це може допомогти зменшити кількість вузьких місць, розподілити знання більш рівномірно та підвищити стійкість команди у вирішенні різноманітних завдань.

Створення ролі для Agile-коучів:

- Впровадження Agile-коучів: Для великих проектів або команд, які тільки починають працювати за гнучкими підходами, варто розглянути можливість введення ролі Agile-коуча. Agile-коучі можуть допомогти командам ефективніше адаптувати свої робочі практики та глибоко впровадити гнучкі принципи в робочий процес команди [4].

- Безперервне навчання та вдосконалення: Покладіть на Agile-коучів обов'язок бути в курсі останніх гнучких практик та інструментів, які вони можуть впроваджувати в командах за потреби [5].

Ці рекомендовані вдосконалення гнучких практик Scrum Agile та коригування ролей покликані підвищити ефективність та результативність процесу управління проекту. Вдосконалюючи ці аспекти, проектні команди можуть досягти більшої продуктивності, покращити результати проекту та створити більш залучену та вмотивовану команду.

Підвищення чіткості ролей

Для покращення чіткості та визначення ролей у матричній структурі управління проектом рекомендується виконати наступні кроки:

- Детальні описи ролей: Розробити всебічні описи ролей, які включають конкретні обов'язки, рівні повноважень та очікувані результати для кожної посади. Це має бути зроблено у співпраці з відділом кадрів та керівниками відділів, щоб забезпечити узгодженість у всій організації.

- Семінари з розподілу ролей: Організуйте семінари за участю членів команди та менеджерів для обговорення та уточнення рольових очікувань. Ці сесії можуть допомогти усунути будь-які збіги або прогалини в обов'язках і сприяти спільному розумінню.

- Матриця RACI: Впровадьте матрицю RACI (Responsible, Accountable, Consulted, Informed - Відповідальний, Підзвітний, Поінформований) для основних проектів та процесів. Цей інструмент допомагає визначити ролі з точки зору їхньої участі у виконанні різних завдань і прийнятті рішень, тим самим уточнюючи, хто за що відповідає, і зменшуючи потенціал для конфліктів.

- Регулярні наради з перегляду ролей: Запровадьте періодичні огляди ефективності ролей, залучаючи зворотній зв'язок від колег та керівників. Ці огляди можуть допомогти адаптувати ролі до мінливих потреб проекту та організаційних цілей.

Інтеграція відділів

Для посилення інтеграції різних відділів та функцій в межах ТОВ "ГлобалЛоджик Україна", що забезпечить кращу узгодженість та узгодженість проектів, розгляньте наступні кроки:

- Міжфункціональні команди: Створіть постійні міжфункціональні команди, до складу яких увійдуть представники різних відділів. Ці команди повинні працювати разом над проектами від початку до завершення, що може покращити взаєморозуміння та співпрацю між функціональними підрозділами.

- Інтеграційні лідери: Призначити інтеграційних лідерів у проектах, основним обов'язком яких є забезпечення злагодженої роботи різних відділів. Ці керівники повинні добре знати роботу кожного відділу і виступати в ролі будівельників мостів у проекті.

- Спільні сесії планування: Проводьте спільні сесії планування за участю керівників і ключових членів команди з усіх відповідних відділів на початку кожної основної фази проекту. Ця практика гарантує, що всі відділи узгоджені з цілями проекту і розуміють, як вони сприяють їх досягненню.

- Спільні цілі та KPI: Розробіть спільні цілі та ключові показники ефективності (KPI), для досягнення яких необхідна співпраця між відділами.

Такий підхід гарантує, що відділи будуть зацікавлені в ефективній спільній роботі.

- **Міжвідомчі тренінги та ротації:** Впровадьте міжвідомчі навчальні програми, в рамках яких працівники проводять час в інших відділах. Це може покращити взаєморозуміння та сприяти відчуттю спільної мети. Крім того, розгляньте програми ротації для управлінських рівнів, щоб розширити їхнє розуміння організації.

- **Комунікаційні платформи:** Використовуйте централізовані цифрові комунікаційні платформи, де можна ділитися оновленнями проєктів, новинами відділів та історіями успіху. Це дозволяє всім бути в курсі того, що відбувається в компанії, і залучатись до подій.

- **Механізми зворотного зв'язку:** Створіть чіткі та доступні механізми зворотного зв'язку, які дозволять членам команди висловлювати занепокоєння та пропозиції щодо інтеграції відділів. Цей зворотний зв'язок має регулярно переглядатися та розглядатися керівниками проєктів та організацій.

Впроваджуючи ці рекомендації, проєкт може зміцнити свою матричну структуру, покращивши як чіткість розподілу ролей, так і інтеграцію відділів. Ці покращення призведуть до більш ефективного управління проєктами, зменшення кількості конфліктів та кращого узгодження на шляху до спільних цілей, що в кінцевому підсумку сприятиме успіху проєктів та організаційному зростанню.

Впровадження нових технологій

Для подальшого підвищення ефективності розробки та можливостей управління проєктами в я рекомендую впровадити наступні технології та інструменти:

- **Безсерверні обчислювальні платформи:** Впровадьте безсерверні архітектури, такі як AWS Lambda або Azure Functions, щоб керувати внутрішніми сервісами програми без необхідності виділення

серверів або управління ними. Це може зменшити операційні витрати та складність, одночасно покращуючи масштабованість.

- Інструменти аналітики на основі штучного інтелекту: Використовуйте аналітичні інструменти на основі штучного інтелекту, такі як Google BigQuery або AWS QuickSight, щоб отримувати корисну інформацію з великих наборів даних, створених програмою управління мережею 5G. Ці інструменти можуть покращити процеси прийняття рішень та оптимізувати продуктивність мережі.

- Блокчейн для посилення безпеки: Впровадьте технологію блокчейн для захисту конфіденційних транзакцій і обміну даними в системі управління мережею. Блокчейн може забезпечити вищий рівень безпеки та прозорості в таких операціях, як зміна конфігурації та контроль доступу.

- Вдосконалені інструменти CI/CD: Розширте поточне використання Jenkins, інтегрувавши його з передовими платформами CI/CD, такими як CircleCI або GitLab CI. Ці інструменти пропонують більш надійні можливості автоматизації, краще управління середовищем і кращу інтеграцію з іншими інструментами, що використовуються в проєкті.

- Моніторинг та аналітика в режимі реального часу: Доповніть існуючі інструменти моніторингу, такі як Prometheus та Grafana, інструментами обробки даних у реальному часі, такими як Apache Kafka та Apache Storm. Ці технології можуть допомогти в обробці та аналізі потоків даних у реальному часі, що має вирішальне значення для динамічних і високооб'ємних мережевих середовищ 5G.

Таблиця 3.3.2

Потенційні заощадження від впровадження нових технологій

| Технологія/Інструмент | Метричний показник/характеристика | Поточний стан | Покращений стан | Оціночні заощадження (USD) |
|----------------------------------|-----------------------------------|---------------|-----------------|----------------------------|
| Платформи безсерверних обчислень | Витрати на і надання серверів | \$20,000/рік | \$10,000/рік | \$10,000 |

| | | | | |
|---|--|----------|---------|----------|
| | Масштабованість та операційна ефективність | Середній | Високий | \$5,000 |
| Інструменти аналітики на основі ШІ | Ефективність аналізу даних | 60% | 90% | \$15,000 |
| | Швидкість прийняття рішень | Середній | Високий | \$5,000 |
| Блокчейн для підвищеної безпеки | Кількість порушень безпеки та втрат даних | 2/рік | 0/рік | \$20,000 |
| | Прозорість і аудит | Низький | Високий | \$5,000 |
| Покращені інструменти CI/CD | Час розгортання (годин/тиждень) | 20 | 10 | \$7,000 |
| | Надійність автоматизації | Середній | Високий | \$3,000 |
| Моніторинг і аналітика в реальному часі | Швидкість обробки даних | Середній | Високий | \$10,000 |
| | Час виявлення і вирішення проблем | Середній | Низький | \$8,000 |
| Загальні заощадження | | | | \$88,000 |

Джерело: самостійно розроблена таблиця



Рис. 3.3.2. Потенційні заощадження від впровадження нових технологій.

Джерело: самостійно розроблений графік

Припущення та розрахунки:

1. Платформи безсерверних обчислень:
 - Скорочення витрат на обслуговування і надання серверів з \$20,000/рік до \$10,000/рік заощаджує \$10,000.
 - Покращена масштабованість та операційна ефективність, оціночні додаткові заощадження: \$5,000.
 2. Інструменти аналітики на основі ІІІ:
 - Підвищення ефективності аналізу даних з 60% до 90% призводить до кращих інсайтів і операційних покращень, оціночні заощадження: \$15,000.
 - Швидше прийняття рішень завдяки кращим аналітичним інструментам, оціночні додаткові заощадження: \$5,000.
 3. Блокчейн для підвищеної безпеки:
 - Скорочення кількості порушень безпеки та втрат даних з 2 на рік до 0 на рік, оціночні заощадження: \$20,000.
 - Підвищена прозорість і аудит операцій, оціночні додаткові заощадження: \$5,000.
 4. Покращені інструменти CI/CD:
 - Скорочення часу розгортання з 20 годин/тиждень до 10 годин/тиждень заощаджує 10 годин. Припускаючи команду з 10 членів і середню погодинну ставку \$70: $10 \text{ годин/тиждень} * 52 \text{ тижні/рік} * 10 \text{ членів} * \$70/\text{година} = \$7,000$.
 - Покращена надійність автоматизації, оціночні додаткові заощадження: \$3,000.
 5. Моніторинг і аналітика в реальному часі:
 - Покращення швидкості обробки даних з середнього до високого рівня, оціночні заощадження: \$10,000.
 - Скорочення часу виявлення і вирішення проблем з середнього до низького рівня, оціночні додаткові заощадження: \$8,000.
- Загальні оціночні заощадження при впровадженні цих нових технологій складають \$88,000.

Удосконалення засобів комунікації

Щоб сприяти кращій співпраці в матричній структурі, особливо в середовищі, де використовуються такі інструменти, як Jira, Confluence та інші, я рекомендую наступні вдосконалення та зміни:

- Платформа уніфікованих комунікацій: Впровадити уніфіковану комунікаційну платформу, яка інтегрується з Jira, Confluence, Jenkins, Docker, Prometheus та Grafana. Такі інструменти, як Slack або Microsoft Teams, можуть централізувати комунікації, забезпечуючи канали, прямий обмін повідомленнями та інтеграцію з існуючими інструментами для впорядкування комунікацій робочого процесу.

- Покращена спільна робота над документами: Оновіть Confluence або інтегруйте додаткові інструменти, такі як Google Workspace або Microsoft 365, для спільної роботи з документами в режимі реального часу. Ці платформи пропонують чудові функції для редагування, коментування та спільного використання документів, які можуть покращити управління документами та спільну роботу над ними в різних відділах.

- Інструменти для візуальної співпраці: Використовуйте інструменти візуальної співпраці, такі як Miro або Lucidchart, які дозволяють командам проводити мозкові штурми, планувати та керувати проектами візуально. Ці інструменти особливо корисні для мапування складних архітектур або робочих процесів і можуть покращити розуміння та залучення членів команди.

- Інтеграції для управління проектами: Удосконалюйте Jira за допомогою додаткових плагінів для управління проектами, таких як BigPicture або Advanced Roadmaps, щоб забезпечити кращі можливості планування, управління ресурсами та видимість графіків проектів для декількох команд.

- Інструменти для зворотного зв'язку та опитувань: Інтегруйте такі інструменти, як SurveyMonkey або Typeform, у комунікаційні платформи, щоб систематично збирати відгуки від членів команди. Це може допомогти

зрозуміти ефективність комунікаційних стратегій та визначити сфери для вдосконалення.

- Автоматизовані інструменти звітування: Використовуйте автоматизовані інструменти звітності, які можуть генерувати звіти з Jira, Jenkins та інших систем і поширювати їх через обрані комунікаційні платформи. Такі інструменти, як Tableau або більш інтегровані рішення в рамках існуючих платформ, можуть періодично надавати інформацію про стан проекту, використання ресурсів та інші важливі показники.

Впроваджуючи ці технології та вдосконалюючи комунікаційні інструменти, проект може значно підвищити ефективність розробки та сприяти створенню більш спільного та продуктивного проектного середовища. Ці вдосконалення не лише спростять процеси, але й відповідатимуть цілям компанії щодо інновацій та досконалості в телекомунікаційному секторі.

Для подальшого вдосконалення управління проектом необхідний надійний підхід до навчання та розвитку. У цьому розділі викладено рекомендації щодо навчальних програм, спрямованих на посилення Agile-компетенцій та лідерських якостей, зокрема в рамках матричної структури управління.

Базові та поглиблені концепції Agile:

- Курс з основ Agile: Пропоную комплексний навчальний курс, який охоплює основи гнучких методологій, зосереджуючись на практиках Scrum, Kanban та Lean. Цей курс має бути обов'язковим для всіх нових членів команди, а також доступний для підвищення кваліфікації існуючих співробітників.

- Просунуті Agile-методи: Для команд, які вже знайомі з базовими практиками Agile, слід запропонувати поглиблені курси, які заглиблюються в більш складні аспекти методологій Agile, такі як масштабування Agile, інтеграція зворотного зв'язку з клієнтами в режимі реального часу та вдосконалене управління бэклогом.

Навчання Agile для конкретних ролей:

- Семінари для власників продуктів: Спеціалізовані воркшопи для Product Owners, які зосереджені на догляді за бэклогом, картах історій користувачів та комунікації із зацікавленими сторонами. Ці воркшопи покращать їхню здатність ефективно управляти бэклогами та максимізувати цінність продукту.
- Сертифікати Scrum Master: Пропонуйте сертифікаційні програми для Scrum-майстрів, які зосереджені на фасилітації, лідерстві та коучингу в Agile-командах. Ці сертифікати повинні бути визнані у всьому світі, наприклад, Certified ScrumMaster (CSM) або Professional Scrum Master (PSM).
- Agile-коучинг: розробити програму підготовки Agile-коучів у компанії, які потім зможуть підтримувати та наставляти інші команди, допомагаючи глибоко та стабільно впроваджувати Agile-практики в організації.

Практичне застосування Agile:

- Практичні Agile-лабораторії: Створіть лабораторні заняття, на яких команди зможуть практикувати Agile-методи в контрольованому середовищі з реальними проєктними сценаріями. Це допоможе їм застосувати теоретичні знання в практичних умовах, покращуючи їх розуміння та навички.
- Міжкомандні Agile-ритрити: Організуйте періодичні виїзні заходи, на яких різні команди обмінюються досвідом, проблемами та рішеннями, пов'язаними з їхніми практиками Agile. Це не тільки покращує навчання, але й сприяє розвитку гнучкої культури в компанії.

Тренінги з лідерства та управління

- Семінари з матричного управління: Провести семінари, спеціально розроблені для вирішення проблем управління в матричному середовищі. Вони повинні охоплювати такі теми, як ефективна комунікація між відділами, управління командною динамікою та збалансування подвійного підпорядкування.

- Індивідуальні лідерські програми: Розробити індивідуальні навчальні програми з лідерства, пристосовані до конкретних потреб лідерів проекту. Ці програми повинні бути зосереджені на прийнятті стратегічних рішень, управлінні змінами та сприянні інноваціям в рамках матричної структури.

Тренінги з вирішення конфліктів:

- Курси з управління конфліктами: Забезпечити курси, які навчають менеджерів та лідерів команд навичкам виявлення, управління та ефективного вирішення конфліктів. Ці курси повинні включати методи медіації, ведення переговорів та сприяння позитивному діалогу між членами команди.

- Заходи з командоутворення: Регулярні заходи з тимбілдингу можуть допомогти пом'якшити конфлікти, покращуючи стосунки та взаєморозуміння між членами команди. Ці заходи мають бути інклюзивними, цікавими та відповідати робочому середовищу.

Особистий розвиток і наставництво:

- Програми наставництва: Створіть програму наставництва, в рамках якої досвідчені лідери наставлятимуть менш досвідчених менеджерів, зосереджуючись на подоланні складнощів матричного управління та лідерських викликів.

- Можливості безперервного навчання: Заохочуйте безперервне навчання, надаючи доступ до постійних освітніх ресурсів, конференцій, вебінарів та семінарів, які лідери можуть відвідувати, щоб бути в курсі останніх тенденцій в управлінні та лідерстві.

Впровадження цих програм навчання та розвитку не лише покращить гнучкі компетенції та лідерські навички проекту, та зробить значний внесок у загальну ефективність та успішність управління проектами в компанії. Ці ініціативи покликані сприяти розвитку культури постійного вдосконалення та адаптації, що має вирішальне значення для підтримки конкурентоспроможності та інновацій у динамічній галузі телекомунікаційних технологій.

ВИСНОВКИ

Комплексний аналіз методологій гнучкого управління проектами, з особливим акцентом на їх впровадженні в ТОВ "ГлобалЛоджік Україна", дозволив отримати значні знання про еволюцію, виклики та майбутні перспективи Agile-практик.

Інтеграція принципів Agile та традиційних методологій: Зіставлення історичної та принципової основи Agile з традиційними методологіями управління проектами підкреслює гнучкість та адаптивність Agile. Це порівняння не лише підкреслює здатність Agile реагувати на зміни, але й його здатність співіснувати та доповнювати більш структуровані, планові підходи. Впровадження Agile в GlobalLogic Ukraine LLC продемонструвало підвищення ефективності проектів та задоволеності стейкхолдерів, підтвердивши ефективність Agile в реальному корпоративному середовищі.

Виклики та стратегії гнучкої трансформації: Підготовка організації до Agile передбачає подолання культурних та операційних бар'єрів. Кроки з переходу на Agile-технології, які спостерігалися в ТОВ "ГлобалЛогістик Україна", включали в себе значне планування, навчання та реструктуризацію. Виклики, що виникали, включаючи опір змінам та необхідність постійного вдосконалення, були подолані за допомогою спеціальних стратегій, що наголошували на комунікації, гнучкості та розширенні можливостей команди. Цей перехід не лише покращив результати проекту, але й сприяв створенню більш динамічного та спільного робочого середовища.

Роль Agile у цифровій трансформації та майбутні тенденції: Гнучкі методології виявилися ключовим фактором цифрової трансформації та розвитку галузі. Вони дозволяють організаціям швидко адаптуватися до технологічного прогресу та ринкових змін. Майбутнє Agile позначене такими тенденціями, як посилення інтеграції з цифровими інструментами, акцент на клієнтоорієнтованих підходах та постійний розвиток Agile-практик відповідно до різноманітних проектних середовищ. Ці тенденції передбачають більш

тонкий, гібридний підхід до управління проектами, поєднання Agile з іншими методологіями відповідно до конкретних потреб проекту.

В кваліфікаційній роботі було проаналізовано дійсний Agile проект та надано рекомендації щодо його покращення:

- Оптимізація планування спринта з акцентом на сесію попереднього планування, тайм-боксингу та чіткості цілей та аналіз результатів.

- Покращення щоденних зустрічей з фокусом на структурований фідбек, демо сесії та залучення стейкхолдерів.

- Покращення ретроспективи

Очікується, що впровадження запропонованих змін дозволить зекономити близько 10400 \$ на спринт.

Крім цього було запропоновано ряд інших ініціатив серед яких впровадження таких нових технологій як:

- Безсерверні обчислювальні платформи
- Інструменти аналітики на основі штучного інтелекту
- Блокчейн для посилення безпеки
- Вдосконалені інструменти CI/CD:
- Моніторинг та аналітика в режимі реального часу

Очікувана економія від впровадження таких технологій складе близько 88000\$.

Насамкінець, дослідження показує, що гнучке управління проектами - це не просто методологія, а спосіб мислення, який наголошує на адаптивності, постійному вдосконаленні та співпраці. Успішне впровадження Agile в ТОВ "ГлобалЛоджик Україна" є прикладом його практичності та ефективності. Майбутній розвиток методологій Agile, ймовірно, буде характеризуватися більшою інтеграцією з технологіями, фокусом на клієнтському досвіді та подальшим розвитком у напрямку гібридних практик управління проектами. Проведене дослідження підтверджує, що гнучкі методології, при правильному

впровадженні та адаптації, можуть призвести до значного покращення ефективності управління проектами, адаптивності та загального успіху.

Ці висновки не лише відповідають цілям, окресленим у вступі, але й пропонують практичні рекомендації для організацій, які прагнуть впровадити або вдосконалити Agile-практики. Впровадження Agile слід розглядати як стратегічне рішення, що вимагає комплексного планування, сприятливої організаційної культури та готовності до постійного навчання і вдосконалення.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Роберт С. М. Чистий AGILE. Назад до основ., 2021 р., 224 с., Видавництво Фабула
2. Майк Кон, Оцінювання і планування в Agile, 2019 р., 336 с., Видавництво Фабула
3. Agile Alliance Resources: Комплексний веб-сайт, що пропонує широкий спектр ресурсів, включаючи статті, дослідницькі роботи та вебінари на тему Agile-методологій. URL: <https://www.agilealliance.org/resources/>
4. Scrum.org Resources: На цьому веб-сайті ви знайдете багато інформації про Скрам, включаючи посібники, кейси та публікації в блогах, написані експертами з Скраму. URL: <https://www.scrum.org/resources>
5. Mountain Goat Software Blog: Цей блог, який веде Майк Кон, провідний експерт з Agile та Scrum, пропонує змістовні статті та поради щодо практик Agile. URL: <https://www.mountaingoatsoftware.com/blog>
6. Agile Connection: Онлайн-спільнота зі статтями, інтерв'ю та дискусіями на різні теми Agile. URL: <https://www.agileconnection.com/>
7. InfoQ - Agile: новинний сайт спільноти, керований практиками, який надає новини, статті та презентації про розробку програмного забезпечення та методології Agile. URL: <https://www.infoq.com/agile/>
8. The Agile Coach від Atlassian: Обширний ресурсний центр, що містить посібники, підручники та статті про Agile-практики, адаптовані для всіх рівнів експертизи. URL: <https://www.atlassian.com/agile>
9. ProjectManagement.com - Agile: Інтернет-ресурс для менеджерів проектів, що пропонує вебінари, статті та шаблони, присвячені гнучкому управлінню проектами. URL: <https://www.projectmanagement.com/topics/agile/>
10. Блог LeadingAgile: Містить статті та ідеї про Agile-трансформацію та про те, як узгодити Agile-методології з бізнес-цілями. URL: <https://www.leadingagile.com/blog/>
11. Ресурси Scrum Alliance: Пропонує різноманітні ресурси,

включаючи статті, кейси та дослідницькі роботи про практики Scrum та Agile.
URL:

<https://www.scrumalliance.org/>

12. Блог про навчання Lean Agile: Пропонує ідеї, поради та стратегії впровадження методологій Lean та Agile в різних організаціях. URL: <https://leanagiletraining.com/blog/>

13. Agile Дайджест: Інтерактивна платформа з навчальними посібниками, інструментами та шаблонами, пов'язаними з Agile-методологіями. URL: <https://agiledigest.com/>

14. Бібліотека ресурсів SAFe (Scaled Agile Framework): Надає вичерпний посібник та ресурси для впровадження Agile в організаціях. URL: <https://scaledagileframework.com/>

15. Історія Agile методології URL: <https://www.leanwisdom.com/blog/the-history-of-agile-methodology>

16. Причина вибрати гнучке керування проектами замість традиційного керування проектами URL: <https://agilemania.com/agile-project-management-vs-traditional-project-management>

17. Сайт компанії GlobalLogic <https://www.globallogic.com/>

18. Охоплення Індустрії 4.0: розкриття потужності цифрової трансформації у виробництві URL: <https://faun.pub/embracing-industry-4-0-unlocking-the-power-of-digital-transformation-in-manufacturing-7c9685a36515>

19. Опис успішної Agile трансформації від GlobalLogic. URL: <https://www.globallogic.com/ua/work/avid-agile-development-services-2/>

20. Джозефф Сазерленд Scrum. Навчись робити вдвічі більше за менший час. 2022 р., 280 с., Видавництво Книжковий клуб “Клуб Сімніного Дозвілля”

21. Дичківська О. О. Інноваційний менеджмент : конспект лекцій. Київ : ДІА, 2018. 82 с.

22. Палеха В. І., Карпова П. В. Менеджмент організацій : навч. посіб. Запоріжжя : ЗНУ, 2015. 120 с.

23. Гуревич Д. Т., Чекан О. С., Грибан О. М., Макарова В. В. Інновації : навч. посіб. Запоріжжя : ЗНУ, 2016. 389 с.
24. Бікулов Д.Т, Чкан А.С., Олійник О.М., Маркова С.В. Менеджмент : навч. посіб. Запоріжжя : ЗНУ, 2017. 360 с.
25. Операційний менеджмент : підручник / С. М. Поплавська та ін. Київ : ЦУЛ, 2011. 267 с.
26. Майк Коун. "Agile Estimating and Planning", 2005 р., 368 с., Видавництво Prentice Hall.
27. Ерік Ріс. "The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses", 2011 р., 336 с., Видавництво Crown Business.
28. Джонатан Расмуссон. "The Agile Samurai: How Agile Masters Deliver Great Software", 2010 р., 280 с., Видавництво The Pragmatic Bookshelf.
29. Джин Кім, Кевін Бер і Джордж Спэффорд. "The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win", 2013 р., 432 с., Видавництво IT Revolution Press.
30. Лісса Адкінс. "Coaching Agile Teams: A Companion for Scrum Masters, Agile Coaches, and Project Managers in Transition", 2010 р., 352 с., Видавництво Addison-Wesley Professional.
31. Кеннет Рубін. "Essential Scrum: A Practical Guide to the Most Popular Agile Process", 2012 р., 504 с., Видавництво Addison-Wesley.
32. Мішель Слігер і Стейша Бродерік. "The Software Project Manager's Bridge to Agility", 2008 р., 416 с., Видавництво Addison-Wesley Professional.
33. Проект з відкритим кодом по впровадженню рішення по оркестрації 5G. <https://nephio.org/>

ДОДАТКИ

Додаток А

Огляд Agile експертизи в GlobalLogic



Agile Expertise Overview

| | | | | |
|---|----------------------------------|--------------------------------------|------------------------------|---|
| 2000+ Certified Agile Practitioners | 800+ Executed Projects | 120+ Agile Transformations | 400+ Happy Clients | Partnership Scaled Agile SILVER |
|---|----------------------------------|--------------------------------------|------------------------------|---|

- **16+ Years** of Agile Principles and Practices Experience. Agile is at the core of GlobalLogic's DNA
- **Programs Ranging 100 to 500 people** working together delivering time critical products using complex engineering in a distributed environment across the globe making geographical boundaries meaningless
- **Scaled Silver Partnership** enabling GlobalLogic and its clients embrace best of Lean Agile Thinking and help provide SAFe trainings and certifications at a competitive rate

| GL ADVANTAGES | VALUE DELIVERED | CASE STUDIES |
|--|--|---|
| Leaders in establishing distributed Scaled Agile Processes | 500% Improved Cycle Time | 350+ team running on SAFe, distributed in 5+ geographies, working together to develop an education platform for a global education provider 150+ team running on SAFe, distributed between US, Europe and Asia developing pharmacy software for a leading global healthcare provider |
| Proven Enterprise Agile Transformation Expertise | 200% Productivity Gains | |
| Inhouse Planning & Automated Reporting Solutions | ZERO Critical & Blocker Defects | |
| Specialized in Agile Training, Coaching and Mentoring | 2 Weeks Build Feedback Loop | |

**Consultative Approach
Operational Excellence
Smooth Execution
Deeper Insights
Actionable Intelligence**

| Service Offerings | |
|---|------------------------|
| Agile Coaching Program | |
| Agile Training Program | |
| Agile Advisory Program | |
| Agile Tools Assessments, Reporting | |
| Agile Product Management Workshop | |
| Technology Experience (Short representative list) | |
| Agile Project Management | Jira, Azure DevOps |
| Knowledge Management | Confluence, SharePoint |
| Reporting | Klerna, Power BI |
| Agile Expertise (Short representative list) | |
| 500+ Certified Project Managers, incl. PMP, SAFe, Scrum | |
| 1000+ Certified Scrum Masters (CSM) | |
| 200+ Certified Scrum Product Owners (CSPO) | |
| 50+ Certified Scrum Professionals (CSP) | |
| 500+ SAFe Certified Engineers and SPCs (6) | |

Джерело: внутрішній сайт компанії GlobalLogic.

Дозволяється використовувати дану інформацію лише в рамках поточної кваліфікаційної роботи

Огляд GlobalLogic Agile практики

GlobalLogic
A Hitachi Group Company

Agile Practice

8

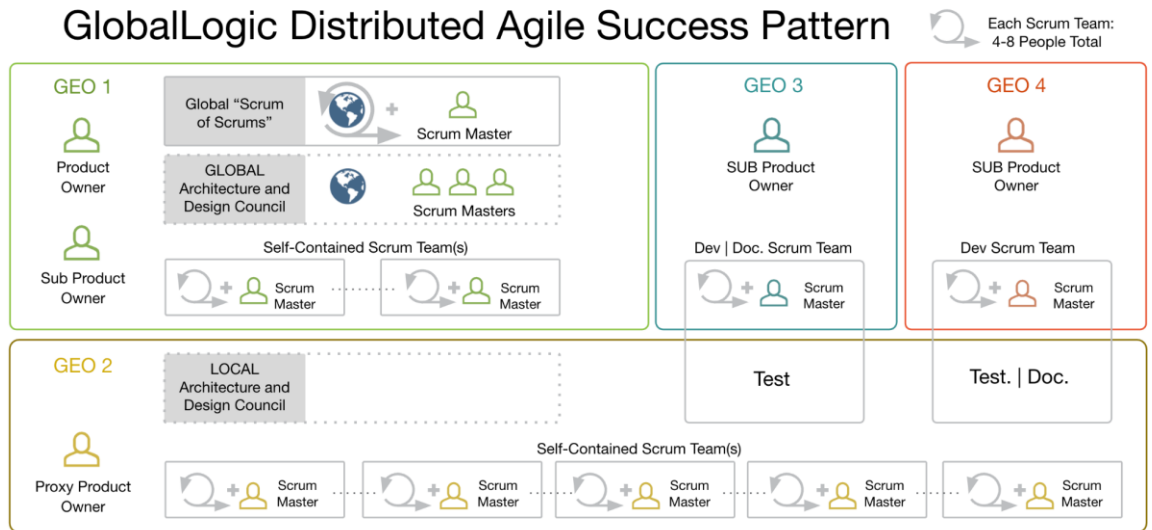
At GlobalLogic, we understand the need for focus and investment in Agile space. We have a dedicated practice group, of technically strong individuals, motivated and passionate about Distributed Agile



Джерело: внутрішній сайт компанії GlobalLogic.

Дозволяється використовувати данну інформацію лише в рамках поточної кваліфікаційної роботи

Типовий GlogalLogic шаблон для розподілений Agile команд



Джерело: внутрішній сайт компанії GlobalLogic.

Дозволяється використовувати данну інформацію лише в рамках поточної кваліфікаційної роботи

