**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**
**ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**
**ІНЖЕНЕРНИЙ НАВЧАЛЬНО-НАУКОВИЙ**
**ІНСТИТУТ ім. Ю. М. ПОТЕБНІ**
Кафедра інформаційної економіки, підприємництва та фінансів
(повна назва кафедри)

# Кваліфікаційна робота(проєкт)

магістра
(рівень вищої освіти)

на тему Моделювання управління проєктами ІТ-підприємства

Виконав: студент 2 курсу, групи 8.0513-ie
спеціальності 051 Економіка
(код і назва спеціальності)
спеціалізації _____
(код і назва спеціалізації)
освітньої програми Інформаційна економіка
(назва освітньої програми)
(ініціали та прізвище) Є. О. Солдатова

Керівник проф., д.е.н., проф. Клопов І.О.
(посада, вчене звання, науковий ступінь, прізвище та ініціали)
Рецензент_____
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Запоріжжя
2024

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**
**ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

Інженерний навчально-науковий інститут ім. Ю. М. Потебні
Кафедра інформаційної економіки, підприємництва та фінансів
Рівень вищої освіти магістр
Спеціальність 051 Економіка
(код та назва)
Спеціалізація _____
(код та назва)
Освітня програма Інформаційна економіка

**ЗАТВЕРДЖУЮ**
Завідувач кафедри_____
«_____»_____20____року

**З А В Д А Н Н Я**
НА КВАЛІФІКАЦІЙНУ РОБОТУ (ПРОЄКТ)СТУДЕНТОВІ (СТУДЕНТЦІ)
Солдатовій Єлизаветі Олександрівні
(прізвище, ім'я, по батькові)

1. Тема роботи (проекту) Моделювання управління проєктами ІТ-підприємства керівник роботи Клопов І. О., д.е.н., доц.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом ЗНУ від «28» лютого 2024 року
Строк подання студентом роботи 21 листопада 2024 р.
2. Вихідні дані до роботи організаційна структура та інформаційні потоки управління проєктами ТОВ «Антс»
3. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз системи управління проєктами. 2. Методи та алгоритми управління проєктами іт підприємства. 3 Проєктування автоматизованої системи управління проєктами ТОВ «Антс».
4. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) Етапи життєвого циклу проєкту. Блок-схема робочого процесу розробки програмних продуктів. Блок-схема алгоритму *R&D.* Модель бізнес-процесів етапу *R&D* на ІТ підприємстві. Схема взаємодії баз даних. Алгоритм ухвалення рішення про успішність моделі. Дерево рішень моделі *Crisp DM.* Модель управління процесами *As Is.* Діаграма бізнес-процесів моделі *To Be* управління проєктами на етапі R&D

5. Консультанти розділів роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|---|---|---|---|
| | | завдання видав | завдання прийняв |
| 1 | професор, д.е.н., в.о. зав. кафедри інформаційної економіки, підприємництва та фінансів Клопов І.О. | 17.09 | 19.10 |
| 2 | професор, д.е.н., в.о. зав. кафедри кафедри інформаційної економіки, підприємництва та фінансів Клопов І.О. | 19.10 | 29.10 |
| 3 | професор, д.е.н., в.о. зав. кафедри кафедри інформаційної економіки, підприємництва та фінансів Клопов І.О. | 29.10 | 16.11 |

6. Дата видачі завдання_____

## КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів кваліфікаційної роботи | Строк виконання етапів роботи | Примітка |
|---|---|---|---|
| 1 | Призначення наукових керівників. Затвердження тем дипломних робіт | *28.02.2024* | |
| 2 | Напрацювання теоретичного матеріалу: дослідження сутності об’єкту та предмету дослідження, критичний аналіз існуючих методологічних засад, вибір та обґрунтування напрямку проведення дослідження | *29.10.2024* | |
| 3 | Апробація результатів на Міжнародних та Всеукраїнських конференціях | *5.11.2024* | |
| 4 | Розробка економіко-математичного забезпечення основних елементів концептуального підходу | *19.11.2024* | |
| 5 | Збір та систематизація статистичного та нормативного матеріалу дослідження. | *30.11.2024* | |
| 6 | Узагальнення отриманих результатів. Оформлення роботи | *01.12.2024* | |
| 7 | Надання роботи до рецензії. Нормоконтроль | *21.11.2024* | |
| 8 | Прилюдний захист дипломної роботи на засіданні ЕК | *12.12.2024* | |

Студент _____ Є. О. Солдатова
(підпис) (ініціали та прізвище)

Керівник роботи (проєкту) _____ І. О. Клопов
(підпис) (ініціали та прізвище)

**Нормоконтроль пройдено**

Нормоконтролер _____ _____
(підпис) (ініціали та прізвище

# РЕФЕРАТ

Солдатова Є. О. Кваліфікаційна робота магістра «Моделювання управління проєктами ІТ-підприємства» : 83 с., 27 рис., 10 табл., 40 джерел.

Ключові слова: УПРАВЛІННЯ ПРОЄКТАМИ, АВТОМАТИЗОВАНА СИСТЕМА, МОДЕЛЮВАННЯ, БІЗНЕС-ПРОЦЕСИ, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ, ОПТИМІЗАЦІЯ РЕСУРСІВ, ПРОЄКТНА ДІЯЛЬНІСТЬ

Магістерська робота присвячена розробці та теоретичному обґрунтуванню моделі автоматизованої системи управління проєктами.

В роботі вивчено сучасні методи та інструменти управління проєктами; проаналізовано існуючі інформаційні системи з управління проєктами; з'ясовано особливості управління проєктами ІТ комапній; виконано проєктування та розроблена модель АСУ проєктами.

Вперше розроблено модель автоматизованої системи управління проєктами для ІТ-компанії, яка спеціалізується на створенні продуктів із використанням штучного інтелекту.

Уточнено підхід до подання життєвого циклу проєкту у вигляді дерева рішень із чітким описом фаз проєктної діяльності; алгоритм процесу проєктної діяльності на етапі досліджень і розробок (*R&D*), що забезпечує оптимізацію ресурсів і підвищення ефективності управління.

# ABSTRACT

Soldatova E. O. Master's degree thesis "Modeling project management of IT enterprises": 83 p., 27 fig., 10 tables, 40 sources.

Keywords: PROJECT MANAGEMENT, AUTOMATED SYSTEM, MODELING, BUSINESS PROCESSES, INFORMATION TECHNOLOGY, RESOURCE OPTIMIZATION, PROJECT ACTIVITY

The master's thesis is devoted to the development and theoretical substantiation of a model of an automated project management system.

The work studies modern methods and tools of project management; analyzes existing information systems for project management; clarifies the features of project management of IT companies; designs and develops a model of an automated project management system.

For the first time, a model of an automated project management system for an IT company specializing in the creation of products using artificial intelligence has been developed.

The approach to presenting the project life cycle in the form of a decision tree with a clear description of the phases of project activity has been refined; the algorithm of the project activity process at the research and development (R&D) stage, which ensures resource optimization and increased management efficiency.

Keywords: PROJECT MANAGEMENT, AUTOMATED SYSTEM, MODELING, BUSINESS PROCESSES, INFORMATION TECHNOLOGY, RESOURCE OPTIMIZATION, PROJECT ACTIVITY

ЗМІСТ

ВСТУП…… 7
РОЗДІЛ 1 АНАЛІЗ СИСТЕМИ УПРАВЛІННЯ ПРОЄКТАМИ …. 9
1.1. Основні поняття в управлінні проєктами ……. 9
1.2. Моделі життєвих циклів та методи управління проєктами…. 16
1.3. Аналіз інструментів управління проєктами……... 33
1.4. Висновки до розділу 1…….. 45
РОЗДІЛ 2 МЕТОДИ ТА АЛГОРИТМИ УПРАВЛІННЯ ПРОЄКТАМИ ІТ ПІДПРИЄМСТВА……. 47
2.1 Алгоритми процесу розробки програмних продуктів та методи управління проєктами……. 47
2.2 Організація процесу дослідницької діяльності ІТ підприємства …… 53
2.3 Модель системи управління проєктами «*As Is*»…….. 62
2.4. Висновки до розділу 2……..
РОЗДІЛ 3 ПРОЄКТУВАННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ УПРАВЛІННЯ ПРОЄКТАМИ ТОВ «АНТС» 67
3.1. Модель автоматизованої системи управління проєктами підприємства ТОВ «Антс»…… 67
3.2. Якісна оцінка моделі автоматизованої системи управління проєктами ТОВ «Антс»……. 76
3.3. Висновки до розділу 3…….. 81
ВИСНОВКИ…… 82
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ…….. 84

# ВСТУП

*Актуальність теми* дослідження обумовлена сучасними викликами у сфері управління проєктами, які постають перед ІТ-компаніями в умовах швидкозмінного ринку технологій. Зростання конкуренції, прискорення темпів впровадження інновацій та необхідність адаптації до нових умов роботи вимагають від компаній використання ефективних підходів до управління проєктами. Традиційні методи часто не відповідають сучасним вимогам, що призводить до збоїв у роботі проєктних команд, затримок у реалізації проєктів та збільшення витрат. Це створює потребу у вдосконаленні існуючих підходів шляхом впровадження автоматизованих систем управління.

Особливе значення в управлінні проєктами набуває використання інформаційних систем, які дозволяють координувати дії команди, ефективно розподіляти ресурси, аналізувати ризики та забезпечувати контроль за ходом виконання завдань. Розробка моделей таких систем відкриває нові можливості для підвищення продуктивності компаній, оптимізації бізнес-процесів та адаптації до сучасних викликів. Це особливо важливо для підприємств, які працюють над створенням складних технічних рішень і прагнуть зберігати свою конкурентоспроможність.

*Об'єкт* дослідження: система управління проєктами в ІТ-компанії.

*Предмет дослідження*: методи та засоби моделювання автоматизованої системи управління проєктами.

*Метою дослідження* є розробка та теоретичне обґрунтування моделі автоматизованої системи управління проєктами.

Для досягнення мети були поставлені та вирішені такі *завдання*:

– вивчено сучасні методи та інструменти управління проєктами;

– проаналізовано існуючі інформаційні системи з управління проєктами;

– з'ясовано особливості управління проєктами ІТ компній;

– виконано проєктування та розроблена модель АСУ проєктами на

підприємстві ТОВ «Антс».

*Методи дослідження.* У процесі виконання дослідження були використані такі підходи та методи: загальнонаукові прийоми та способи логічного пізнання: аналіз та синтез, абстрагування; системно-структурний, функціональний та формально-логічний підходи; аналіз літератури з питань теорії управління проектами, моделювання АСУ проєктами, існуючих стандартів управління проектами; емпіричні методи (індивідуальні та групові інтерв'ю зі співробітниками компанії, спостереження за їх діяльністю протягом процесу створення програмного продукту, мозковий штурм, бенчмаркінг); методи структурного та об'єктно-орієнтованого аналізу моделювання інформаційної системи; експертна оцінка; SWOT-аналіз; оцінка рівня автоматизації бізнес-процесів.

Наукова новизна одержаних результатів полягає у наступному:

*вперше:*

– розроблено модель автоматизованої системи управління проєктами для ІТ-компанії, яка спеціалізується на створенні продуктів із використанням штучного інтелекту;

*уточнено:*

– підхід до подання життєвого циклу проєкту у вигляді дерева рішень із чітким описом фаз проєктної діяльності;

– алгоритм процесу проєктної діяльності на етапі досліджень і розробок (*R&D*), що забезпечує оптимізацію ресурсів і підвищення ефективності управління.

Результати теоретичного аналізу проблеми розробки та теоретичного обґрунтування моделі автоматизованої системи управління проєктами пройшли апробацію на Міжнародній науково-практичній конференції «Формування концепції цифрової трансформації освіти та соціально-відповідального цифрового громадянина у контексті європейського досвіду»

# РОЗДІЛ 1
# АНАЛІЗ СИСТЕМИ УПРАВЛІННЯ ПРОЄКТАМИ

## 1.1. Основні поняття, що використовуються в управлінні проєктами

Термін «проєкт» широко використовується у різних сферах: науці, виробництві та навіть у повсякденному житті. Існує безліч підходів до його визначення, що відображають специфіку цієї категорії в науковій літературі. Зокрема, поняття «проєкт» можна трактувати як:

- організовану діяльність, що має на меті створення унікальних продуктів, послуг або результатів.
- окреме підприємство, яке переслідує конкретні цілі, враховуючи часові, фінансові та якісні обмеження щодо досягнення запланованих результатів.
- намір або ініціативу, що характеризується унікальністю сукупності умов, таких як цілі, обмеження ресурсів (часових, фінансових, людських) та специфічна організація виконання завдань [30].

Адлер О. визначає проєкт як послідовність взаємопов'язаних подій, що реалізуються протягом встановленого обмеженого періоду часу і спрямовані на досягнення унікального, але водночас чітко визначеного результату [4].

Дячок І. пропонує таке трактування: проєкт – це обмежена в часі цілеспрямована зміна окремої системи, яка має чітко визначені цілі. Досягнення цих цілей завершує проєкт, а сам процес супроводжується встановленням вимог до строків, результатів, ризиків, бюджету, ресурсів та організаційної структури [9].

Кіндрат О. наголошує, що проєкт – це цілеспрямована, обмежена в часі діяльність, що виконується для задоволення конкретних потреб з урахуванням зовнішніх і внутрішніх обмежень, використовуючи обмежені ресурси [29].

Незважаючи на різноманітність визначень, можна виділити кілька спільних рис, що визначають сутність проєкту:

– чітко сформульовані цілі із відповідними технічними, економічними та іншими показниками.

– чистемний характер: проєкт розглядається як сукупність взаємозалежних елементів (цілей, завдань, операцій, ресурсів, результатів). Це дозволяє алгоритмізувати проєкт, представляючи його у вигляді комплексу взаємопов’язаних процесів.

– встановлені часові рамки: чітко визначені дати початку та завершення проєкту.

– обмеженість ресурсів: фінансових, людських, матеріальних тощо.

– унікальність цілей і умов виконання.

Таким чином, проєкт – це динамічна система взаємопов’язаних дій, які спрямовані на досягнення заданих результатів у багатокритеріальному середовищі. Він виконується протягом визначеного часу, з урахуванням обмежених ресурсів та із залученням фахівців, які володіють необхідними знаннями і навичками.

Контекстна діаграма (рис. 1.1) демонструє проєкт як комплексну систему, в якій взаємодіють різні елементи. На вхід подаються технічні вимоги та умови фінансування, що визначають рамки і напрямок роботи. Основною метою проєкту є досягнення запланованого результату у відповідності до поставлених цілей.




Рисунок 1.1 – Проєкт як цілісна система

Ключову роль у виконанні проєкту відіграють ресурси: матеріальні, фінансові та людські. Саме вони забезпечують реалізацію необхідних робіт. Однак, ефективність виконання безпосередньо залежить від якісного управління процесом.

До функцій управління входять:

– розподіл ресурсів з урахуванням потреб кожного етапу проєкту;

– координація послідовності операцій, що забезпечує узгодженість дій між виконавцями;

– компенсація впливу зовнішніх і внутрішніх факторів, що можуть вплинути на виконання запланованих робіт [21, 22].

Детальний опис складових елементів проєкту подано у таблиці 1.1, яка розкриває їх зміст і функціональну роль у досягненні кінцевого результату.

Таблиця 1.1 – Характеристика складових елементів системи проєкту

| Елемент | Зміст |
|---|---|
| Ціль (Результат) | Описуються нові продукти або послуги, які отримає замовник в результаті реалізації проєкту |
| Вартість проекту | Фінансові витрати, необхідні для виконання робіт проекту |
| Технічні вимоги | Обсяги робіт (кількісні показники обсягу робіт проєкту); терміни виконання; якість (відповідність характеристик проекту та його продукції встановленим заздалегідь параметрам якості) |
| Ресурси | Устаткування, матеріали, персонал, програмне забезпечення, інформаційні системи; виробничі площі; фахівці та організації, залучені до виконанню робіт проекту, їх кваліфікація |
| Внутрішні фактори впливу | Стиль керівництва проектом; організація проекту з крапки зору комунікації між основними учасниками проекту, розподілу має рацію, відповідальності і обов'язків; методи і кошти взаємодії між співробітниками всіх рівнів на проекті; умови праці та техніки безпеки, страхування та соціальне забезпечення тат.п. |
| Зовнішні фактори впливу | Взаємодія з замовником і конкурентами; ситуація на ринку і пов'язані з цим ризики; непередбачені обставини |

Існує безліч підходів до визначення терміна «управління проєктами». Зупинимося на деяких із них.

Згідно з PMBoK, управління проєктами – це процес використання знань, навичок, методів, інструментів і технологій для виконання проєктних завдань з метою досягнення цілей, поставлених учасниками проєкту [17, 31].

Ентоні Вокер визначає управління проєктами як процес, що охоплює планування, координацію та контроль над реалізацією проєкту від імені замовника. Його метою є досягнення встановлених параметрів корисності, функціонального призначення, якості, строків виконання та витрат. Він також акцентує увагу на важливості встановлення зв'язків між ресурсами, контролю за внеском кожного учасника та вибору найкращих альтернатив для максимальної відповідності інтересам замовника [25].

Гарольд Оберлендер розглядає управління проєктами як мистецтво та здатність координувати матеріальні й нематеріальні ресурси, організовувати послідовність виконання робіт у встановлені строки та в межах узгодженого бюджету [23].

На думку Фонар Л.С. управління проєктами є синтетичною дисципліною, яка об'єднує спеціалізовані та міждисциплінарні знання [28].

Таким чином, різні підходи до визначення «управління проєктами» підкреслюють його багатогранність. Незважаючи на це, спільною рисою є важливість організації взаємодії між учасниками проєкту. Ця взаємодія забезпечується через діяльність проєктного менеджера, який відіграє ключову роль у визначенні проблем і пошуку їхніх рішень.

Очевидно, що людський ресурс є найціннішим активом у процесі управління проєктами, оскільки саме люди генерують ідеї та втілюють їх у життя.

У науковій літературі існують різноманітні класифікації проєктів за різними критеріями. Узагальнюючи дані з джерел [1, 9, 20], можна побудувати таблицю 1.2, у якій лівий стовпець містить критерії класифікації, а правий – відповідні види проєктів.

Таблиця 1.2 – Види проєктів

| Критерій | Вид проєкту |
|---|---|
| по тривалості періоду здійснення проєкту | короткострокові (до 2-х років), середньострокові (до 5-ти років), довгострокові (понад 5-ти років) |
| по складу і структурі проєкту | монопроєкт , мультипроєкт , мегапроєкт |
| за характером предметний області проєкту | інвестиційний, інноваційний, науково-дослідницький, навчально-освітній,змішаний |
| по складності | прості, складні, дуже складні |
| по галузям економіки і соціальної сфери | промисловість, будівництво, транспорт, охорона здоров'я, туризм |
| по результуючому продукту | продукт (частина іншого вироби, удосконалення виробу або кінцеве виріб); послуга (бізнес-функція, оптимізує виробничі процеси); покращення (існуючою лінійки продуктів або послуг); нематеріальний продукт (наприклад, дослідницький проект приносить нові знання) |
| за основними сферами діяльності, в яких здійснюється проєкт | технічний, організаційний, економічний, соціальний, змішаний |
| за масштабом (за розмірами бюджету, кількості учасників і ступеня впливу на навколишній світ) | дрібні, середні, великі |
| ступеня охоплення етапів інноваційного процесу | повні інноваційні проєкти, що включає НДР, ДКР, освоєння нововведення та його комерціалізацію, неповні інноваційні проекти, що включають окремі етапи інноваційного процесу |

Класифікація проєктів дає змогу чітко структурувати як перспективні, так і вже реалізовані проєкти. Це дозволяє встановлювати досяжні цілі, визначати реалістичні терміни їх досягнення та оптимально залучати необхідні ресурси для успішної реалізації [2, 19]. Такий підхід сприяє ефективному управлінню проєктами, знижуючи ризики невиконання або перевищення бюджету.

Одним із ключових аспектів управління проєктами є поняття життєвого циклу проєкту. Воно відображає певну послідовність етапів реалізації ідеї у виробничому або управлінському процесі. Значення цього поняття полягає у його здатності забезпечувати:

Фіксацію тривалості проєкту, що дозволяє чітко визначити його часові межі.

Деталізацію процесу реалізації, розбиваючи його на конкретні фази, що полегшує управління кожним етапом.

Оптимізацію ресурсів і персоналу, шляхом визначення кількості людей та засобів, необхідних для виконання завдань на кожному етапі.

Спрощення процедури контролю, забезпечуючи можливість регулярної оцінки прогресу і виявлення проблемних зон.

Життєвий цикл проєкту охоплює всі фази від початкової ідеї до її реалізації. Цей підхід має не лише теоретичне, а й практичне значення, оскільки забезпечує ефективний контроль над процесом розробки продукту чи виконання завдань.

Для ІТ-проєктів, зокрема, виділяють такі основні етапи життєвого циклу:

Ініціація – формулювання ідеї, визначення мети проєкту та попередня оцінка його доцільності.

Планування – розробка плану дій, встановлення строків, бюджету та визначення ресурсів.

Виконання – реалізація проєкту відповідно до затвердженого плану.

Завершення – підведення підсумків, оцінка результатів та передача продукту замовнику (рис. 1.2).

Таким чином, життєвий цикл проєкту є ключовим інструментом управління, що забезпечує ефективну організацію роботи, мінімізує ризики та підвищує ймовірність успішної реалізації задуму.

Аналіз життєвих циклів проєктів виявив низку спільних закономірностей, які притаманні всім проєктам незалежно від їхньої специфіки:

– пікові витрати і залученість персоналу припадають на середину життєвого циклу проєкту. Початкові та завершальні етапи, навпаки, характеризуються низьким рівнем витрат і меншою кількістю залучених фахівців.

– рівень ризику найвищий на початкових етапах, оскільки в цей період відбувається багато невизначеностей і закладаються основи подальшого виконання.

– гнучкість у внесенні змін і вдосконаленні процесів реалізації проєкту також є найбільшою на початкових етапах, тоді як із часом можливості для змін значно зменшуються [8, 17].




Рисунок 1.2 – Основні фази життєвого циклу проєкту

Султанов І. А. зазначає, що з набуттям досвіду проєктні менеджери дедалі краще усвідомлюють важливість правильного управління життєвим циклом для зниження ризиків і досягнення більш передбачуваних результатів. Одним із ключових інструментів є система оцінки ризиків.

Важливою складовою є планування проєкту з урахуванням фаз життєвого циклу. Після завершення кожного етапу встановлюються контрольні точки (так звані «віхи»), у яких керівники:

– аналізують досягнуті результати;

– проводять прогнозну оцінку подальших дій;

– приймають рішення щодо наступних кроків у реалізації унікального завдання.

Досвід, знання та інтуїція проєктного менеджера відіграють вирішальну роль у прийнятті таких важливих рішень [32].

Модель життєвого циклу має високу методологічну цінність, оскільки:

– вона забезпечує детальний опис послідовності дій, необхідних для виконання проєкту;

– сприяє своєчасному прийняттю управлінських рішень;

– забезпечує повний контроль над виконанням завдань на кожній стадії життєвого циклу.

Таким чином, модель життєвого циклу дозволяє оптимізувати процеси управління проєктом, мінімізуючи ризики і підвищуючи ефективність реалізації задуму.

## 1.2. Моделі життєвих циклів проєктів та методи управління проєктами

Управління проєктами як професійна діяльність і наукова дисципліна почало активно розвиватися у 1980-х роках. Це був період, коли світова економіка виходила з кризи, ринки насичувалися товарами і послугами, а перед компаніями постали нові масштабні завдання. У цей час з'явилася перша значуща праця *PMBoK* (*A Guide to the Project Management Body of Knowledge*), розроблена *Project Management Institute* (*PMI*) – організацією, яка нині є провідною професійною асоціацією у сфері стандартизації управління проєктами.

Окрім *PMI*, вагомий внесок у стандартизацію управління проєктами зробили такі організації:

– *IPMA (International Project Management Association)*;

– *OGC* (*The Office of Government Commerce*, розробник стандарту *PRINCE2*);

– *ISO* (*International Organization for Standardization*);

– *APM* (*Association for Project Management*) та інші.

Варто зазначити, що, крім міжнародних стандартів, існують національні, галузеві та корпоративні системи стандартизації, які враховують специфіку окремих регіонів і сфер діяльності.

Швидкий розвиток IT-сфери та активна робота над створенням програмного забезпечення сприяли накопиченню великого обсягу практичного досвіду, який постійно збагачується. Цей досвід, відомий як «найкращі практики» (*best practices*), застосовується на різних етапах життєвого циклу проєкту. Стандарт *SWEBOK* називає сукупність таких практик методологією розробки програмного забезпечення.

Методології створення програмних продуктів є однією з найдинамічніших сфер інформаційних технологій, оскільки вони базуються на реальних практичних знаннях і постійно вдосконалюються.

Методи управління проєктами нерозривно пов’язані з фазами життєвого циклу проєкту. Вони не лише впливають один на одного, а й формують загальну структуру процесу управління. Фази життєвого циклу, представлені на рис. 1.2, є універсальними й застосовуються у різних галузях, таких як:

– будівництво, де етапи проєкту чітко визначені, а послідовність робіт суворо регламентована.

– авіаційна та космічна промисловість, де можливе паралельне виконання операцій або повторення окремих видів робіт, якщо цього вимагає специфіка проєкту [5, 7, 15, 19].

Стандарти і методології управління проєктами дозволяють організувати роботу максимально ефективно, знижуючи ризики та підвищуючи ймовірність досягнення поставлених цілей.

Для успішної розробки конкретного програмного продукту необхідна детальна деталізація його життєвого циклу та побудова адекватної моделі, яка відповідає специфіці проєкту.

Життєвий цикл програмного продукту базується на моделі, яка визначає концептуальний підхід до його організації, а також основні фази та принципи переходу між ними. У свою чергу, методологія (або метод) задає конкретний

набір робіт, їхній зміст та розподіл ролей серед фахівців на кожному етапі. Методологія зазвичай містить рекомендації з використання відповідних практик (best practices) для досягнення максимальної ефективності [33].

Традиційно першими були розроблені каскадна (водоспадна) модель, вперше описана в 1970 році Вінстоном Ройсом, та спіральна модель, запропонована Баррі Боемом у 1986 році. Згодом, із розвитком інформаційних технологій і ускладненням функціоналу програмних продуктів, з'явилися гнучкі моделі (Agile), які нині стали найпоширенішими.

Каскадна модель є однією з найстаріших і найвідоміших (рис. 1.3.). Вона побудована за принципом послідовності: кожен етап розробки починається лише після завершення попереднього. Це забезпечує строгий контроль процесу, що було великою перевагою в умовах стабільних вимог.

Основні риси каскадної моделі:

– простота та зрозумілість: чітка структура робіт полегшує планування і контроль.

– логічна послідовність: кожна фаза завершується, перш ніж розпочинається наступна.

– строгий контроль якості на кожному етапі.

Однак, у сучасних умовах, коли вимоги до програмного продукту можуть змінюватися навіть у процесі його розробки, каскадна модель втрачає свою ефективність. Її строгий і статичний підхід може стати перешкодою для гнучкого реагування на нові вимоги або виправлення помилок на ранніх етапах.

Незважаючи на це, каскадна модель продовжує застосовуватися у великих компаніях для масштабних і складних проєктів, де необхідний всеосяжний контроль ризиків [27].

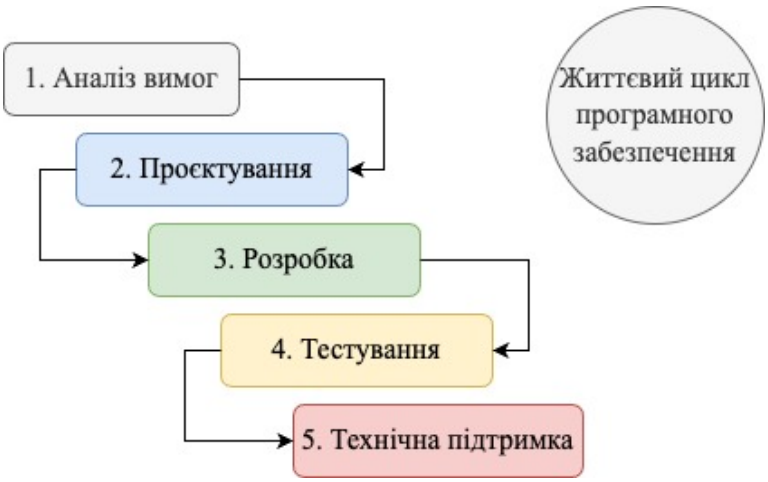



Рисунок 1.3 – Каскадна модель життєвого циклу

Переваги каскадної моделі.

1. Проста та зрозуміла структура робочого процесу. Завдяки чіткій послідовності етапів, ця модель легко сприймається як фахівцями, так і замовниками, що сприяє кращому розумінню завдань.

2. Можливість точного прогнозування ресурсів. Завчасне планування дозволяє визначити обсяг необхідних ресурсів і строків виконання завдань.

3. Мінімальні витрати на комунікацію. Оскільки кожен етап виконується послідовно, немає потреби у постійній координації між усіма членами команди.

Недоліки каскадної моделі.

1. Формалізований підхід. Строга послідовність етапів може стати перешкодою для адаптації до нових вимог або змін у процесі розробки.

2. Відсутність гнучкості. Замовник не має можливості вносити зміни в специфікацію продукту до завершення всього проєкту, що може призвести до невідповідності кінцевого продукту очікуванням.

3. Зниження якості через обмеження ресурсів. У разі нестачі часу чи бюджету скорочення етапу тестування може суттєво вплинути на якість кінцевого продукту.

Хоча каскадна модель і досі застосовується, особливо у великих і масштабних проєктах, її популярність значно знизилася. На зміну їй прийшли

сучасніші та більш адаптивні моделі, що враховують динамічні зміни у вимогах до програмного забезпечення.

2. Спіральна модель (рис. 1.4).

Спіральна модель розробки базується на ітераційному підході, де проєкт проходить кілька повторюваних циклів. Особлива увага приділяється початковим етапам, таким як аналіз вимог та проєктування.

Основні особливості спіральної моделі:

– початковий етап спрямований на обґрунтування доцільності розробки;

– створюється прототип майбутнього продукту, який дозволяє виявити та врахувати ризики ще до повної реалізації рішення [15].

Такий підхід забезпечує більшу гнучкість та адаптивність у порівнянні з каскадною моделлю, дозволяючи мінімізувати ризики і підвищити якість кінцевого продукту.

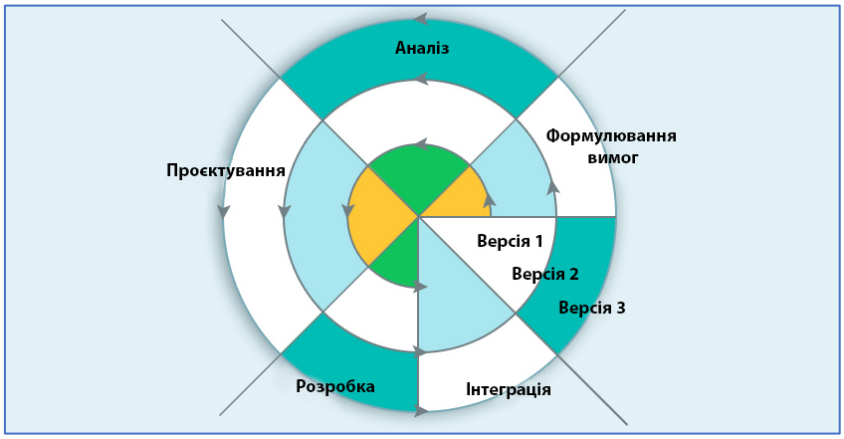



Рисунок 1.4 – Спіральна модель життєвого циклу

Свою назву спіральна модель отримала завдяки специфічному ітераційному підходу, що створює візуальний образ спіралі. Кожен цикл цієї моделі наближає програмний продукт до фінальної форми та змісту, поступово додаючи нові функції та уточнення.

Цей підхід дозволяє на кожній ітерації:

– уточнювати цілі та параметри проєкту;

– виявляти та вирішувати проблеми, які можуть виникати на різних етапах розробки;

– планувати наступні етапи роботи, враховуючи досвід попередніх циклів.

Завдяки ітеративній структурі спіральна модель дозволяє зменшити загальний час виконання завдань та підвищити якість кінцевого продукту.

Переваги спіральної моделі.

1. Гнучкість у розробці. Спіральна модель усуває обмеження жорсткого, лінійного процесу каскадної моделі, надаючи можливість адаптації до нових вимог.

2. Легке внесення змін. Якщо замовник змінює свої вимоги, модель дозволяє оперативно внести корективи без шкоди для основного процесу.

3. Тісна взаємодія з користувачем. Постійний зв’язок між замовником і розробником допомагає швидко виявляти помилки або слабкі місця проєкту.

4. Ефективніше управління проєктом. Завдяки ітераціям можна краще контролювати хід виконання завдань і оперативно реагувати на ризики.

5. Зниження ризиків. Ітераційний підхід дозволяє вчасно виявити і мінімізувати ризики, які можуть вплинути на успіх проєкту.

Недоліки спіральної моделі.

1. Відсутність чітких часових рамок. Оскільки етапи життєвого циклу не мають суворих обмежень за часом, це може ускладнювати контроль.

2. Ризик затягування проєкту. Можливість постійного внесення змін до вимог може призводити до збільшення кількості ітерацій, що подовжує терміни розробки.

3. Складність реалізації. Модель має досить складну структуру, що створює труднощі для розробників і менеджерів, особливо у разі недостатньої підготовки або відсутності досвіду роботи з цією методологією.

Спіральна модель є ефективним інструментом для розробки складних проєктів, особливо в умовах змінних вимог. Вона забезпечує гнучкість і адаптивність, однак потребує чіткої організації та досвідченого управління для успішного застосування.

3. Гнучкі (ітеративні) моделі (Agile) (рис. 1.5)

Окрім послідовних підходів, широко застосовуються ітеративні (інкрементальні) моделі, які належать до сімейства методологій Agile. Ці методології базуються на спільному підході до управління проєктами, що акцентує увагу на:

1. Людях і взаємодії як пріоритеті над процесами та інструментами.
2. Робочому продукті, який важливіший за вичерпну документацію.
3. Співпраці із замовником замість суворого дотримання умов контракту.
4. Готовності до змін на противагу сліпому дотриманню початкового плану [26, 28].

У ітеративних моделях життєвий цикл проєкту поділяється на серію міні-циклів або ітерацій. Кожна ітерація складається з базових етапів життєвого циклу (планування, розробка, тестування, впровадження). Протягом ітерації відбувається розробка окремих компонентів системи, які потім додаються до вже створеного функціоналу.

Однією з ключових переваг ітеративного підходу є те, що для початку роботи над продуктом не потрібно повного обсягу вимог. Спочатку визначаються лише базові потреби, а згодом вимоги можуть уточнюватися та доповнюватися.

Кожна ітерація завершується створенням нової версії продукту з оновленим функціоналом. Після завершення кожного міні-циклу оцінюються результати, і приймається рішення про використання цих результатів як основи для наступної ітерації. Цей підхід відомий як інкрементальне прототипування.

Процес повторюється до того моменту, поки всі вимоги не будуть повністю реалізовані в продукті. У підсумку відбувається реліз – випуск готового продукту, який відповідає всім поставленим цілям і потребам замовника.

Переваги ітеративного підходу.

Гнучкість у розробці. Зміни в вимогах легко впроваджуються на будь-якому етапі проєкту.

Постійний прогрес. Кожна ітерація приносить видимий результат, що дозволяє замовнику оцінити хід виконання проєкту.

Ризик-менеджмент. Регулярні перевірки та тестування знижують ризик виникнення критичних помилок у фінальній версії продукту.

Ефективна співпраця з замовником. Замовник має змогу активно брати участь у процесі розробки, надаючи зворотний зв’язок після кожної ітерації.

Ітеративні моделі, зокрема Agile, сьогодні стали стандартом у багатьох галузях, особливо в розробці програмного забезпечення, завдяки своїй гнучкості та орієнтації на результат.




Рисунок 1.5 – Ітеративна модель життєвого циклу

З точки зору математичної термінології, ітеративна модель реалізує процес послідовної апроксимації, тобто поступового наближення до заданих параметрів і, врешті-решт, до готового продукту.

Успіх застосування цієї моделі значною мірою залежить від строгої валідації вимог і верифікації функціональності на кожній ітерації, що дозволяє вчасно виявляти і виправляти помилки.

Процес розробки в ітеративній моделі повторює ключові етапи водоспадної моделі: планування, розробка, тестування та впровадження. Проте кожна ітерація завершується створенням робочої версії продукту, яка проходить ретельне тестування на всіх рівнях.

Переваги ітеративної моделі.

1. Високий рівень взаємодії команди. Постійна комунікація між учасниками проєкту сприяє ефективній роботі.

2. Швидкий результат. Завдяки коротким циклам («спринтам») створюється робочий код, який можна швидко протестувати і продемонструвати.

3. Стимулювання покращень. Під час розробки продукту з’являється можливість вносити зміни та вдосконалення на основі зворотного зв’язку.

4. Безпосередня участь замовника. Замовник активно залучається до процесу, що зменшує ризик непорозумінь і забезпечує відповідність продукту його очікуванням.

5. Висока адаптивність. Модель легко пристосовується до змін у вимогах та умовах проєкту.

6. Швидка реакція на зміни. Проєктна команда може оперативно реагувати на внутрішні та зовнішні фактори.

7. Ідеальна для інновацій. Модель підходить для розробки продуктів із високим ступенем невизначеності та обмеженою інформацією на початкових етапах.

Недоліки ітеративної моделі.

1. Ризик нескінченних змін. Постійні корективи вимог можуть затягувати завершення проєкту.

2. Залежність від команди. Рівень успіху значною мірою залежить від кваліфікації та досвіду команди.

3. Складність оцінки вартості. Через постійні зміни та ітерації важко точно спрогнозувати остаточну вартість проєкту.

4. Вимоги до менеджменту. Ефективність застосування Agile-методик залежить від компетентності управлінців. Для спрощення роботи використовуються спеціальні підходи, такі як Scrum і Kanban.

До найпоширеніших методологій сімейства Agile належать:

- Scrum;
- Lean;
- Kanban;
- Six Sigma;
- PRINCE2.

Розглянемо детальніше основні з них.

3.1. Scrum – найбільш структурована методологія в групі *Agile* (рис. 1.6).

Scrum виділяється своєю чіткою організацією процесів і підходить для комплексних проєктів з динамічними вимогами. Ця методологія забезпечує максимальну ефективність роботи команди завдяки розподілу ролей, чітким цілям на кожен спринт і постійному зворотному зв'язку.




Рисунок 1.6 – *Scrum* модель життєвого циклу

У Scrum проєкт розбивається на окремі частини – завдання, які поступово реалізуються протягом усього циклу розробки. Завдання отримують пріоритетність залежно від їхньої важливості.

Проєкт поділяється на ітерації – визначені проміжки часу, які встановлює команда і погоджує із замовником. Ітерації використовуються для контролю та управління ходом виконання.

Основні принципи Scrum.

– пріоритезація завдань. Найважливіші завдання виконуються першими.

– робочий результат після кожної ітерації. Після завершення ітерації замовник отримує робочий проміжний варіант продукту.

– постійний перегляд вимог. Перед початком кожного спринту проводиться переоцінка залишкових завдань, і, за необхідності, вносяться зміни до проєкту.

– розподіл відповідальності. У процесі беруть участь проєктний менеджер, команда виконавців і замовник.

Процес організовано навколо п’яти основних нарад:

1. Нарада-упорядкування. У перший день ітерації команда аналізує, що вже зроблено, і визначає завдання, які потрібно виконати.

2. Нарада-планування. Команда обирає із пріоритетного списку ті завдання, які будуть виконуватися під час поточної ітерації.

3. Щоденні летючки. Короткі (до 15 хвилин) щоденні зустрічі, де члени команди обмінюються інформацією про стан виконання своїх завдань.

4. Підсумкова нарада. Команда демонструє результати роботи всім зацікавленим сторонам і перевіряє відповідність продукту поставленим цілям та очікуванням.

5. Нарада-ретроспектива. Проводиться аналіз командної роботи, виявляються проблеми в методах або комунікації, і формулюються висновки для поліпшення процесів у майбутніх ітераціях.

Переваги Scrum.

1. Швидкий результат. Підходить для проєктів, де потрібні швидкі та придатні для використання проміжні версії продукту.

2. Ефективна комунікація. Постійний зв’язок між членами команди дозволяє оперативно виявляти і виправляти помилки.

3. Гнучкість у роботі з командою. Scrum допускає наявність менш досвідчених розробників, оскільки завдяки тісній співпраці вони можуть швидко адаптуватися.

Недоліки Scrum.

1. Вимоги до соціальних навичок. Постійна і тісна взаємодія між учасниками проєкту може бути стресовою для людей, які не мають розвинених комунікативних навичок або схильні до інтровертності.

2. Непридатність для строго регламентованих сфер. Scrum не підходить для проєктів у галузях, де робота строго нормована і вимагає дотримання жорстких стандартів, наприклад, у юриспруденції.

Scrum ідеально підходить для адаптивних проєктів з постійно змінними вимогами, забезпечуючи швидкий результат і високу залученість замовника, але вимагає від команди високого рівня комунікації та взаємодії.

3.2. Kanban – метод із групи Agile.

На відміну від Scrum, метод Kanban не використовує чітко визначених ітерацій. Процес створення продукту базується на обмеженні кількості одночасно виконуваних завдань. У Kanban відсутні часові рамки для етапів розробки, що дає команді більше гнучкості.

Особливості Kanban:

– процес роботи не розбивається на чіткі ітерації;

– немає регламентованих виробничих нарад;

– кожен член команди може працювати над кількома завданнями одночасно;

– завдання можна призупиняти і повертатися до них пізніше, якщо змінилися пріоритети або з’явилися важливіші завдання.

Переваги Kanban.

1. Відсутність жорстких дедлайнів. Це знижує стрес і дозволяє команді зосередитися на якості виконання завдань.

2. Економічна ефективність. Завдяки розумному розподілу завдань між членами команди ресурсами користуються максимально ефективно.

Недоліки Kanban.

1. Потреба у взаємозамінності. Метод найбільш ефективний у командах, де кожен учасник може виконувати завдання різного типу.

2. Непридатність для проєктів зі строгими термінами. Відсутність часових рамок робить цей метод непридатним для проєктів, де жорстко обумовлені строки виконання.

3.3. CRISP-DM – модель для проєктів, пов’язаних із ШІ

З розвитком штучного інтелекту (ШІ) постала необхідність адаптації життєвих циклів для проєктів, що включають роботу з великими даними та інтелектуальними системами. CRISP-DM (Cross Industry Standard Process for Data Mining) спочатку була створена для проєктів Data Mining, але згодом виявилася ефективною для розробки ШІ-вмісних програмних продуктів.

Особливості CRISP-DM:

1. Організація структури життєвого циклу. Продукти зі штучним інтелектом мають відмінну структуру життєвого циклу, що враховує специфіку роботи з великими обсягами даних.

2. Ітеративність із оновленням результатів. У CRISP-DM кожен новий цикл оновлює або навіть перезапускає результати попереднього, на відміну від моделей, що базуються на поступовому накопиченні досягнень.

3. Зосередженість на обробці та інтерпретації даних. Основний акцент робиться на глибинний аналіз і використання даних для створення прогнозних моделей.

Етапи CRISP-DM:

– бізнес-розуміння. Визначення цілей і завдань проєкту.

– розуміння даних. Збір, дослідження та підготовка даних для аналізу.

– підготовка даних. Очищення, трансформація і вибір даних для побудови моделей.

– моделювання. Створення математичних і алгоритмічних моделей.

– оцінка. Аналіз отриманих моделей і їхня відповідність бізнес-цілям.

– впровадження. Реалізація готового продукту на основі моделі.

Переваги CRISP-DM.

1. Гнучкість і адаптивність до різних типів проєктів.

2. Можливість багаторазового перегляду та вдосконалення результатів.

3. Чітка структура, яка охоплює всі аспекти роботи з даними.

Недоліки CRISP-DM.

1. Висока залежність від якості початкових даних.

2. Складність управління в умовах великої кількості ітерацій.

Схема моделі CRISP-DM представлена на рисунку 1.7.




Рисунок 1.7 – Модель життєвого циклу дослідження даних *CRISP-DM*

CRISP-DM (Cross Industry Standard Process for Data Mining) складається з шести основних фаз, які охоплюють увесь процес роботи з даними: від визначення бізнес-цілей до впровадження отриманих результатів.

1. Розуміння бізнес-цілей (Business Understanding).

На цьому етапі визначаються основні цілі аналізу, які мають відповідати потребам бізнесу.

Завдання фази: дослідити бізнес-процеси компанії, сформулювати цілі та вимоги до аналізу даних.

Учасники: залучаються всі зацікавлені сторони, щоб виробити спільне бачення та створити план аналітичного проєкту.

Результат: вирішення питання доцільності проєкту та розробка попереднього плану дій.

2. Початкове вивчення даних (Data Understanding).

На цьому етапі проводиться детальний аналіз доступних даних для виявлення можливих проблем і їх вирішення на наступних стадіях.

Завдання: отримати доступ до даних, провести їхній аналіз за допомогою таблиць і графіків, оцінити якість даних.

Результат: створення документації, яка описує доступні дані та їхню відповідність цілям проєкту.

3. Підготовка даних (Data Preparation).

Цей етап є найбільш трудомістким і може займати від 50% до 70% усього часу проєкту.

Завдання: консолідація даних із різних джерел; формування вибірок для аналізу; агрегування, збагачення та очищення даних; розподіл даних на навчальні та тестові набори.

Результат: створення готового до аналізу набору даних, який відповідає поставленим цілям.

4. Моделювання (Modeling).

На цьому етапі створюються аналітичні моделі для обробки даних і досягнення поставлених бізнес-цілей.

Особливості: моделі будуються протягом кількох ітерацій для досягнення найкращих результатів.

Результат: побудова та тестування математичних моделей, які можуть прогнозувати або пояснювати закономірності в даних.

5. Оцінка (Evaluation).

Ця фаза включає перевірку результатів моделювання на відповідність бізнес-цілям.

Завдання: оцінити, наскільки отримані моделі задовольняють вимоги проєкту, та виявити можливі недоліки.

Учасники: ключові особи компанії приймають рішення щодо доцільності впровадження результатів.

Результат: затвердження або коригування моделей для подальшого використання.

6. Впровадження (Deployment).

Фінальна фаза, на якій результати аналізу інтегруються в бізнес-процеси компанії для підвищення їхньої ефективності.

Завдання: застосувати нові ідеї, знання та моделі у повсякденній діяльності підприємства.

Результат: впровадження рішення, яке забезпечує підвищення продуктивності та конкурентоспроможності компанії.

Модель CRISP-DM забезпечує системний підхід до аналізу даних, дозволяючи ефективно управляти всіма етапами життєвого циклу проєкту. Її структурованість і гнучкість роблять її універсальною для використання в різних галузях.

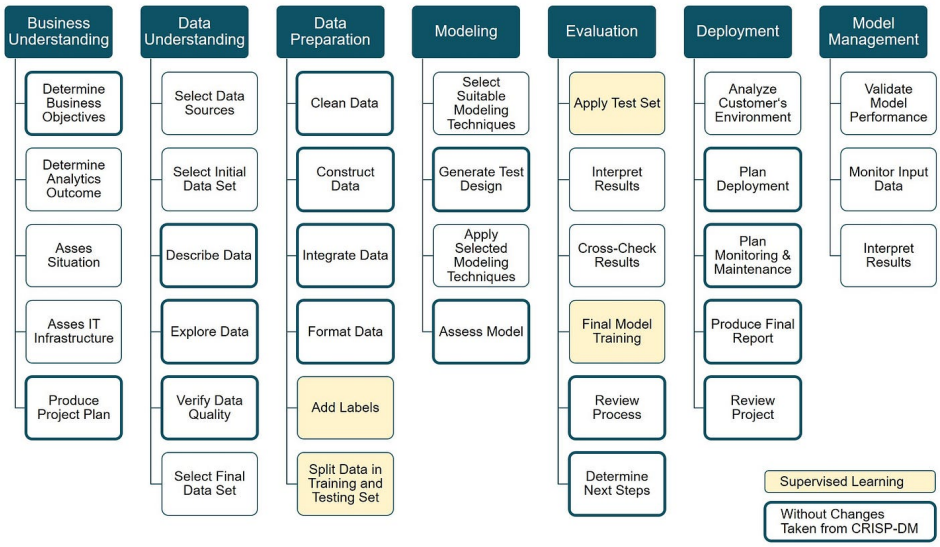



Рисунок 1.8 – Завдання фаз *Crisp DM*

На рисунку 1.8 зображено основні завдання кожної фази життєвого циклу проєкту, розробленого із застосуванням нейронних мереж. Ця модель відображає послідовність кроків, необхідних для створення ефективного продукту на основі штучного інтелекту (ШІ).

Переваги моделі.

1. Нейтральність до предметної області. Модель може використовуватися у різних галузях, від медицини до фінансів, без необхідності значних адаптацій.

2. Зручність для проєктів із ШІ та машинним навчанням (МО). Модель враховує специфіку роботи з інтелектуальними системами та алгоритмами.

3. Інтеграція інтелектуального аналізу даних. Це одна з небагатьох моделей, яка дозволяє використовувати data mining та інші методи обробки великих масивів даних.

4. Релевантність для технологій великих даних (Big Data). Модель підходить для аналізу, обробки та інтеграції великих обсягів інформації, що робить її важливим інструментом у сучасному світі даних.

Недоліки моделі.

1. Відсутність інструментів управління проєктами. Модель не передбачає готових рішень для організації командної роботи, планування або моніторингу проєкту.

2. Відсутність чітких часових меж для кожного циклу. Це може ускладнити контроль і планування етапів реалізації.

3. Змінна тривалість циклів. Оскільки кожна фаза може займати різний час, це може створювати труднощі у прогнозуванні строків завершення проєкту.

Модель життєвого циклу на основі нейронної мережі є потужним інструментом для реалізації проєктів, що базуються на технологіях ШІ та великих даних. Вона пропонує високу гнучкість і адаптивність, проте потребує вдосконалення в аспектах управління проєктами та часової структури для підвищення ефективності використання.

## 1.3. Аналіз інструментів управління проєктами

Інструменти управління проєктами можуть застосовуватися як на всіх етапах життєвого циклу проєкту, так і на окремих його фазах, залежно від специфіки та вимог проєкту. Під такими інструментами розуміють формалізовані методики, процедури, а також шаблони необхідних документів, які забезпечують структурованість і ефективність управління.

Особливості інструментів управління:

1. Методики та процедури. Це стандартизовані підходи до управління проєктами, які допомагають забезпечити послідовність дій та мінімізувати ризики.

2. Шаблони документації. Використання уніфікованих форм документів сприяє ефективній комунікації між членами команди, скорочує час на підготовку звітів і полегшує контроль виконання завдань.

Одним із популярних інструментів для управління Agile-проєктами є Trello – хмарна платформа, розроблена компанією Fog Creek Software.

Основні характеристики Trello:

– багатоплатформність. Система доступна з різних пристроїв і операційних систем, що забезпечує гнучкість роботи.

– простота використання. Інтуїтивно зрозумілий інтерфейс дозволяє швидко адаптуватися до роботи з платформою навіть новачкам.

– організація завдань. Завдяки використанню дошок, списків і карток можна легко відстежувати статус кожного завдання.

– командна взаємодія. Trello дозволяє об’єднати всіх учасників проєкту на одній платформі, забезпечуючи зручну комунікацію та розподіл обов’язків.

– безкоштовний доступ. Для невеликих команд система надає базовий функціонал безкоштовно, що робить її ідеальною для малих проєктів.

Приклади використання Trello (рис. 1.9, 1.10):

Планування ітерацій. Завдання розподіляються за списками (наприклад, «До виконання», «В процесі», «Готово»), що дозволяє візуально відстежувати прогрес.

Пріоритезація завдань. Картки можна легко переміщати між списками, змінюючи їхній статус або пріоритет.

Спільна робота. Члени команди можуть додавати коментарі, вкладення, встановлювати дедлайни та призначати відповідальних за виконання завдань.

Використання сучасних інструментів управління, таких як Trello, значно спрощує координацію проєктних процесів і підвищує ефективність комунікації. Завдяки хмарному формату і простому інтерфейсу, ця платформа ідеально підходить для невеликих груп і Agile-проєктів, забезпечуючи прозорість і контроль на всіх етапах реалізації.

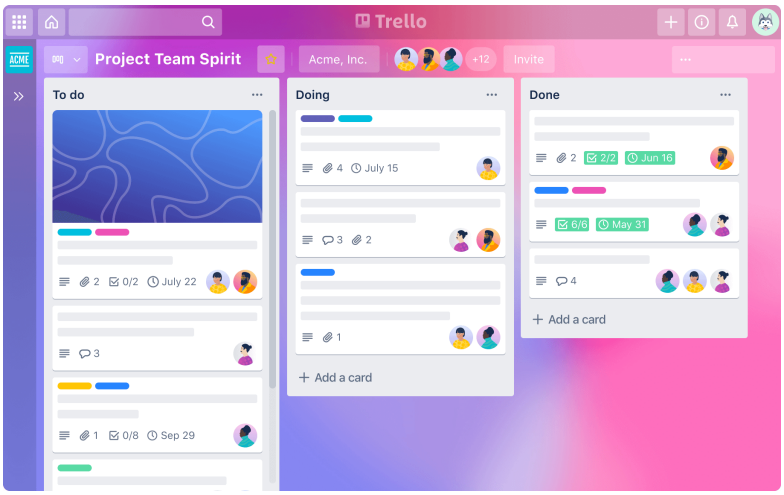



Рисунок 1.9 – *Trello Board*

Trello базується на парадигмі управління проєктами, відомій як Kanban. Цей підхід забезпечує візуалізацію робочого процесу та полегшує управління завданнями.

Основні елементи Trello:

Дошки проєктів. Кожен проєкт представлений у вигляді дошки, що відображає загальний стан завдань.

Списки. На дошці створюються списки, які представляють етапи роботи (наприклад, «Заплановано», «В роботі», «Готово»).

Картки. Завдання відображаються у вигляді карток, які розміщуються у відповідних списках.

Картки можна перетягувати між списками, що відображає рух завдання від однієї стадії до іншої (наприклад, від розробки до тестування).

Кожна картка може містити:

– назву завдання;

– призначених відповідальних користувачів;

– коментарі та вкладення;

– термін виконання;

– чек-листи або списки підзавдань.

Додаткові можливості:

Команди. Користувачі можуть об’єднуватися в команди для роботи над спільними дошками, що полегшує координацію та обмін інформацією.

Кольорові мітки. Trello пропонує шість кольорових тегів для позначення карток, що дозволяє швидко категоризувати завдання.

Markdown. Картки підтримують розмітку Markdown, що дозволяє форматувати текст, наприклад, виділяти жирним, курсивом або створювати списки.

Переваги Kanban у Trello.

1. Простота використання. Інтуїтивно зрозумілий інтерфейс, що дозволяє легко адаптуватися до роботи.

2. Гнучкість. Можливість швидко змінювати статус завдань і додавати нові елементи.

3. Командна співпраця. Зручна система призначення відповідальних і комунікації в межах карток.

4. Візуалізація процесів. Чітко видно, на якому етапі перебуває кожне завдання, що сприяє ефективному управлінню проєктом.

Trello, завдяки використанню Kanban, є універсальним інструментом для управління завданнями в команді. Його простота, гнучкість і широкий функціонал роблять його ідеальним рішенням для малого та середнього бізнесу, а також для особистого планування.

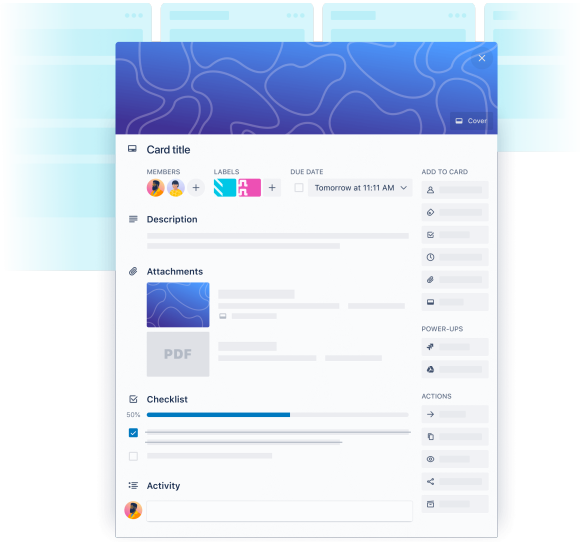



Рисунок 1.10 – Картки завдання *Trello*

Trello – це зручний і багатофункціональний інструмент для управління проєктами, який пропонує низку переваг:

1. Широкий функціонал для спрощення управління проєктом.

Обмін файлами. Ви можете завантажувати документи, фотографії та відео, якими легко ділитися з членами команди.

Коментарі до карток. Можливість залишати зауваження або уточнення безпосередньо в завданнях.

To-do списки. Зручне управління списками завдань і контроль виконання.

Кольорові мітки. Позначення карток за пріоритетністю, що спрощує навігацію.

Необмежене створення карток. Користувачі можуть створювати стільки завдань, скільки потрібно для проєкту.

2. Доступна вартість.

Безкоштовна версія. Пропонує базовий функціонал із можливістю створювати необмежену кількість дошок, карток і списків, а також запрошувати до роботи необмежену кількість учасників.

Платна версія (Business Class). За 25 доларів США на місяць надається розширений функціонал, зокрема інтеграція з Google Службами, об'ємний експорт даних і управління всіма дошками в одному інтерфейсі [35].

3. Ефективний контроль термінів виконання завдань.

Дедлайни та сповіщення. При створенні карток можна додавати дедлайни, що дозволяє контролювати строки виконання завдань.

Колірна індикація. За 1-2 дні до завершення строку картка змінює колір на жовтий, а в день дедлайну – на червоний, нагадуючи про необхідність завершити завдання.

4. Мобільна сумісність (Mobile Friendly).

Trello доступна на всіх платформах і пристроях: комп'ютерах, планшетах і смартфонах. Інструмент автоматично адаптується до розміру екрана, забезпечуючи комфортну роботу незалежно від пристрою.

5. Миттєві сповіщення (Instant Notifications).

Ви завжди будете в курсі змін завдяки функції миттєвих сповіщень про оновлення, коментарі або видалення завдань.

Також є можливість отримувати сповіщення електронною поштою, що забезпечує додатковий контроль над робочим процесом.

Trello – це ефективний інструмент, який спрощує управління проєктами, забезпечуючи прозорість, контроль строків і зручність комунікації. Завдяки доступній вартості та адаптивності він ідеально підходить для різних типів команд і проєктів.

Недоліки роботи з Trello. Попри численні переваги, Trello має низку недоліків, які обмежують його використання для певних завдань:

1. Відсутність offline-доступу. Trello потребує стабільного інтернет-з'єднання для роботи. У разі відсутності Wi-Fi або при використанні пристрою в

режимі «в літаку» доступ до платформи стає неможливим. Це створює незручності для користувачів, які працюють у віддалених місцях або в умовах, де немає інтернету.

Таблиця 1.3 – Огляд Trello

| Призначення продукту | Відстеження завдань проєкту і їх статусів.<br>Управління невеликими проєктами |
|---|---|
| Users | Software developers<br>Project managers<br>SCRUM masters |
| Use Cases | Управління проектами<br>Менеджмент продукту<br>Управління завданнями<br>Розробка програмного забезпечення<br>Agile розробка програмного забезпечення |
| Інтеграція | Confluence<br>JIRA<br>Slack<br>GitHub |
| Hosting options | Cloud |
| | |

2. Обмежена придатність для великих проєктів. Trello оптимально підходить для стартапів, домашніх проєктів або невеликих ідей. Для управління масштабними проєктами, що включають кілька команд із різних країн, цей інструмент є недостатньо функціональним.

3. Обмежений обсяг вкладень. Картки Trello дозволяють прикріплювати файли, проте максимальний розмір кожного файлу в безкоштовній версії становить лише 10 МБ. Для користувачів Trello Gold і Business Class ліміт збільшується до 250 МБ.

4. Відсутність редагування коментарів. Коментарі в картках Trello не можна редагувати після збереження. У разі помилки або необхідності внесення змін доведеться створювати новий коментар.

5. Недостатня візуалізація процесів. Trello не забезпечує повної візуалізації роботи всіх членів команди одночасно, що ускладнює координацію завдань у великих проєктах.

6. Відсутність можливості створення довгострокових планів. Інструмент зручний для короткострокових завдань, проте створення дорожніх карт для проєктів із тривалістю більше двох тижнів є складним.

7. Неможливість перегляду ітерацій та ретроспективи. Trello не має вбудованих інструментів для аналізу виконаних завдань і планування майбутніх. Це ускладнює проведення ретроспектив та оцінку результатів.

8. Відсутність функції обліку часу. У Trello немає можливості відстежувати час, витрачений на виконання завдань. Це створює труднощі в складанні щомісячних звітів і контролі прогресу.

JIRA – це система управління проєктами, розроблена компанією Atlassian, яка використовується для відстеження помилок і координації роботи команд.

Основні переваги JIRA:

1. Веб-інтерфейс. Зручний доступ до платформи через браузер.

2. Широкий функціонал. Платформа пропонує різні продукти для розробки програмного забезпечення, IT, бізнесу та інших команд.

3. Гнучкість. JIRA дозволяє ефективно координувати роботу кількох команд, інтегруючи різні процеси в одну систему.

4. Можливість розгортання. Пропонує варіанти використання як у хмарному середовищі, так і на локальних серверах.

JIRA є надійним інструментом для великих і складних проєктів, що вимагають інтегрованого підходу до управління командною роботою [16] (рис.1.11.).

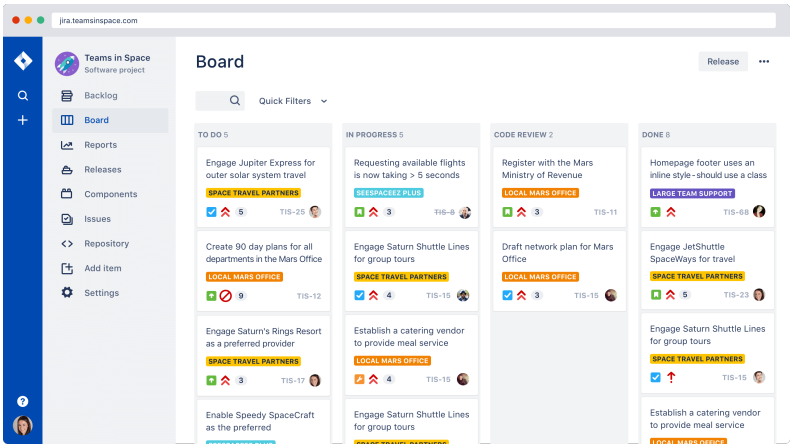



Рисунок 1.11 – Інтерфейс користувача у *JIRA*

Програмні продукти, створені на платформі JIRA, забезпечують командам можливість ефективно планувати, призначати, відстежувати, звітувати та керувати робочими процесами. Платформа охоплює широкий спектр завдань – від гнучкої розробки програмного забезпечення та підтримки клієнтів до управління особистими справами, такими як списки покупок чи сімейні обов'язки.

Платформа включає чотири основні продукти:

1. JIRA Software – орієнтована на команди розробників.
2. JIRA Service Desk – для управління клієнтською підтримкою.
3. JIRA Ops – для оперативного вирішення інцидентів.
4. JIRA Core – базове рішення для бізнес-команд.

Кожен із продуктів постачається з набором вбудованих шаблонів, адаптованих до різних сценаріїв використання. Крім того, всі продукти легко інтегруються, що забезпечує ефективну співпрацю між командами різних відділів і навіть організацій.

JIRA Software – це інструмент, створений для організації робочих процесів у командах, які займаються розробкою програмного забезпечення. Основою його функціональності є робочі процеси, які представляють послідовний шлях від виникнення ідеї до її реалізації.

На рисунку 1.12 представлена базова схема робочого процесу, яка відображає основні етапи роботи: від постановки завдання до його завершення.

Ця схема дозволяє командам структурувати свої процеси, забезпечуючи прозорість та ефективність виконання завдань.

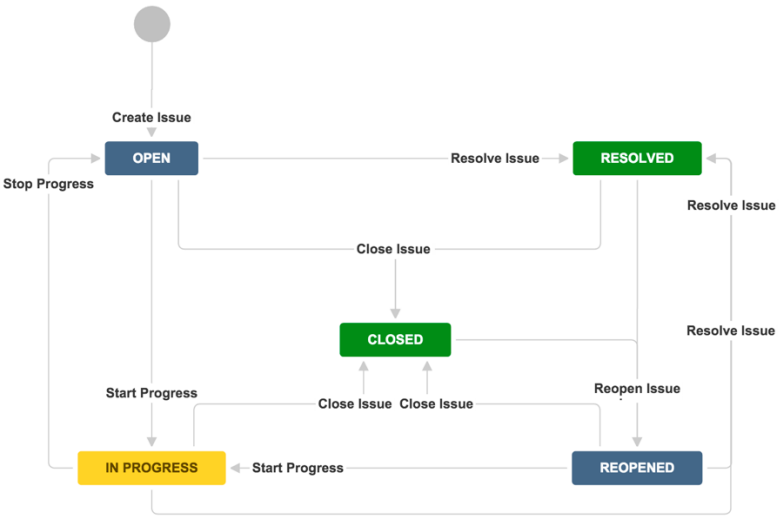



Рисунок 1.12 – *Jira Workflow*

У JIRA Software робочі процеси відображають зміну статусу завдань від початку до завершення. Наприклад, статуси Open і Closed, а також проміжні етапи між ними, вказують на стадії, через які проходить задача. Стрілки між статусами представляють можливі переходи з одного стану в інший.

Робочі процеси можуть бути як простими, так і складними, включаючи додаткові функції, такі як умови, тригери, валідатори та сповіщення. Однак для новачків у JIRA Software рекомендується починати з максимально спрощених робочих процесів, поступово ускладнюючи їх лише за необхідності під час реалізації проєкту [38].

У таблиці 1.5 наведено ключові характеристики продукту JIRA Software, які роблять його універсальним інструментом для управління проєктами:

– призначення: забезпечення ефективного управління завданнями та проєктами.

– потенційні користувачі: розробники, тестувальники, менеджери проєктів, аналітики та інші учасники команд.

– сфери застосування: розробка програмного забезпечення, управління IT-проєктами, підтримка клієнтів, бізнес-аналітика.

– інтеграція з іншими системами: можливість з’єднання з популярними інструментами, такими як Confluence, Slack, GitHub, Bitbucket та іншими.

– опції хостингу: доступність як у хмарному середовищі, так і на власних серверах компанії.

–

Таблиця 1.5 – Огляд *JIRA Software*

| Призначення продукту | Планування, відстеження і випуск програмного забезпечення світового рівня. |
|---|---|
| *Users* | *Software developers*<br>*Project managers*<br>*SCRUM masters* |
| *Use Cases* | Відстеження помилок<br>Управління проєктами<br>Менеджмент продукту<br>Управління процесами<br>Управління завданнями<br>Розробка програмного забезпечення Agile<br>розробка програмного забезпечення |
| Інтеграція | Confluence<br>Bitbucket<br>Slack GitHub |
| Hosting options | Cloud Server, Data Center |

Переваги використання JIRA Software.

1. Чітка видимість процесу виконання завдань. Однією з ключових переваг JIRA є висока видимість робочих процесів. Часто затримки у виконанні проєктів спричиняються нечітким формулюванням завдань або відсутністю прозорої ієрархії їх виконання. JIRA вирішує цю проблему, надаючи можливість усім членам команди бачити прогрес у режимі реального часу завдяки позначкам started і completed. Це дозволяє швидко оцінювати стан проєкту, що сприяє пришвидшенню роботи.

2. Зручне визначення пріоритетів. JIRA дозволяє легко встановлювати пріоритети для завдань. Це допомагає команді чітко розуміти, які завдання потребують негайного виконання, а які можуть бути відкладені. Такий підхід особливо корисний при роботі над кількома проєктами з різними термінами виконання, оскільки забезпечує дотримання дедлайнів.

3. Підвищення продуктивності. Функціонал JIRA дозволяє кожному члену команди в будь-який момент ознайомитися з чергою завдань (backlog). Це зменшує кількість обговорень і простоїв, що виникають під час уточнення деталей, тим самим збільшуючи продуктивність. Незначні простої, на перший погляд, можуть накопичуватись і суттєво впливати на строки виконання завдань. JIRA допомагає уникнути цього, оптимізуючи робочий процес.

4. Безперервна взаємодія команди. JIRA забезпечує безперебійний зв’язок між членами команди незалежно від їхнього місцезнаходження. Завдяки мобільним додаткам співробітники можуть працювати не лише з офісу чи дому, але й з будь-якого місця за допомогою смартфонів або планшетів.

5. Понад 1000 доступних плагінів. Платформа JIRA пропонує більше 1000 плагінів для адаптації під потреби користувачів. Найпопулярнішими з них є GreenHopper і Bonfire, які значно полегшують роботу команд, що дотримуються принципів Agile-розробки [36].

Недоліки використання JIRA Software.

1. Надмірна функціональність. Велика кількість функцій у JIRA може викликати труднощі як у новачків, так і в досвідчених користувачів. Процес налаштування системи потребує значних зусиль через численні параметри, що вимагає часу для розуміння її архітектури.

2. Складний інтерфейс UX/UI. JIRA має перенасичений інтерфейс, що містить багато візуальної інформації та елементів, які не завжди інтуїтивно зрозумілі. Це створює труднощі для користувачів у процесі взаємодії з платформою.

3. Вимога технічних знань. Для ефективного використання JIRA потрібні базові навички в DevOps, розуміння принципів Scrum та вміння налаштовувати спринти для побудови робочої структури.

4. Відсутність функції управління календарем. JIRA не має вбудованого календаря для планування завдань, що може стати недоліком для команд, яким важливо відстежувати часові рамки проєкту.

5. Висока вартість. JIRA є відносно дорогим інструментом. Для команди до 10 осіб вартість складає 10 доларів США на місяць, тоді як для більшої команди (до 100 осіб) ціна становить 7 доларів на кожного члена команди щомісяця.

У сучасних IT-компаніях для ефективної організації спільної роботи з Jira часто використовують Confluence, оскільки обидва продукти розроблені компанією Atlassian і забезпечують зручну інтеграцію між собою.

Confluence – це платформа, яка дозволяє створювати та публікувати вебсторінки й документи у стилі wiki, обмінюватися контентом між членами команди та організовувати обговорення. Основні функції цієї платформи включають:

– централізація інформаційних потоків: всі дані зберігаються в одному місці, що спрощує доступ до інформації.

– файлова система та документація: створення, зберігання, перегляд, обмін і редагування документів стають простими й доступними.

– функція пошуку: швидкий доступ до документів, робочої інформації або навіть електронних листів.

– вбудований корпоративний календар: можливість планування та відстеження подій.

– постановка завдань і контроль виконання: зручний інструмент для управління проєктами.

– корпоративні чати: забезпечують миттєвий обмін повідомленнями між членами команди.

Недоліки Confluence.

Попри широкий функціонал, користувачі вказують на такі недоліки:

– незручна система коментарів: коментарі складно структурувати, що ускладнює обговорення.

– проблеми з одночасним редагуванням текстів: система не підтримує повноцінну роботу декількох користувачів над одним документом.

– недосконале відстеження прогресу проєкту: відсутність чітких інструментів для моніторингу етапів виконання.

– неінтуїтивний тайм-менеджмент: планування та відстеження часу залишають бажати кращого.

– відсутність управління ресурсами: система не передбачає контролю за використанням ресурсів.

– система ціноутворення: вартість продукту зростає зі збільшенням кількості користувачів, що може бути проблемою для великих команд [30].

Confluence має потужний набір функцій, який охоплює більшість аспектів офісної роботи. Хоча існують деякі недоліки, інструмент залишається одним із найбільш популярних для організації командної співпраці завдяки його інтеграції з Jira та можливостям централізації інформації.

## 1.4. Висновки до розділу 1

На основі аналізу різних моделей життєвого циклу проєктів та інструментів управління ними можна зробити кілька ключових висновків.

Модель життєвого циклу CRISP-DM найкраще відображає специфіку процесу створення нейронних мереж. Вона забезпечує структурований підхід до роботи з великими обсягами даних, що є важливим елементом у проєктах, пов’язаних із штучним інтелектом.

Для ефективного управління розробкою програмного забезпечення із застосуванням ШІ доцільно використовувати поєднання гнучких моделей (Agile, Scrum) та CRISP-DM. Гнучка методологія дозволяє швидко адаптуватися до змін

і працювати в умовах невизначеності, а CRISP-DM забезпечує чітку структуру для дослідницької діяльності та аналізу даних.

Гнучкі моделі є ідеальними для стартапів, інноваційних сфер і IT-компаній, де важливі швидкість адаптації та мінімізація ризиків. CRISP-DM найбільш ефективна для управління науково-дослідницькими процесами у відділах R&D. У свою чергу, каскадна модель (Waterfall) оптимальна для проєктів із чіткими вимогами та жорсткими строками, де ключовим обмеженням є час, а не фінанси.

Синхронізація різних підходів до управління дозволяє ефективніше розподіляти завдання між командами, які працюють над одним продуктом. Наприклад, використання гнучкої методології в основному процесі та CRISP-DM у дослідницьких підпроцесах сприяє оптимізації роботи та забезпечує чітку координацію між відділами.

Поєднання гнучких моделей і CRISP-DM дозволяє адаптувати процеси управління до особливостей конкретного проєкту, зберігаючи баланс між гнучкістю, структурованістю та ефективністю. Такий підхід є особливо актуальним для IT-компаній, які працюють у сфері ШІ та інноваційних технологій.

# РОЗДІЛ 2
# МЕТОДИ ТА АЛГОРИТМИ УПРАВЛІННЯ ПРОЄКТАМИ ІТ ПІДПРИЄМСТВА

## 2.1 Алгоритми процесу розробки програмних продуктів та методи управління проєктами

ТОВ «Антс» – це молода, але перспективна продуктова компанія на ринку ІТ-послуг, яка була заснована в Запоріжжі у 2019 році. Її поява обумовлена зростаючим попитом на інтелектуальні програмні рішення, включаючи платформи та сервіси на основі штучного інтелекту (ШІ) та машинного навчання (МН, Machine Learning).

За короткий час свого існування компанія реалізувала низку успішних проєктів, які складають її портфоліо:

1. Рекомендаційна система для купівлі супутніх товарів. Замовником виступила українська мережа аптек.
2. Аналітичний сервіс прогнозування відтоку клієнтів. Цей сервіс використовується в мережах фітнес-клубів, спа-салонів та інших підприємств сфери обслуговування.
3. Платформа для збільшення продажів в електронній комерції.

На даний момент компанія працює над проєктом для банківської та біржової сфер, розробляючи інноваційні рішення для аналізу й оптимізації фінансових операцій.

Всі програмні продукти компанії засновані на методах ШІ та МН. Наприклад, платформа для збільшення продажів в електронній комерції є самонавчальною системою, яка аналізує поведінку користувачів та їхні покупки для постійного вдосконалення. Її основою є унікальна система свідомих рекомендацій (CRS), що працює за оригінальними алгоритмами машинного навчання.

Як працює CRS:

– аналізує покупки, які об'єднані спільною метою;

– пропонує товари у відповідний момент, враховуючи індивідуальні вподобання користувача;

– завдяки глибокому аналізу ймовірність покупки пропонованого товару значно підвищується;

– розробка програмних продуктів із використанням ШІ має більш складну структуру порівняно зі створенням типового програмного забезпечення. Це обумовлено необхідністю глибокого аналізу даних і складністю навчання моделей.

На рисунку 2.1 наведено блок-схему процесу створення програмного продукту, яка відображає основні етапи розробки.

Процес створення будь-якого програмного забезпечення, включаючи продукти на основі ШІ, складається з п’яти ключових стадій:

1. Аналіз завдання. Вивчення умов і вимог замовника.

2. Проєктування. Створення архітектури майбутньої системи.

3. Розробка. Програмування основного функціоналу.

4. Тестування. Перевірка продукту на помилки та відповідність технічному завданню.

5. Супровід. Підтримка та вдосконалення продукту під час його експлуатації замовником.

Ця структура забезпечує системний підхід до розробки та впровадження інноваційних програмних рішень.

На відміну від традиційної розробки програмного забезпечення, створення сервісів, програм і платформ із використанням штучного інтелекту (ШІ), зокрема нейронних мереж, потребує додаткового етапу. Як показано на рисунку 2.2, перед проєктуванням програмного рішення виділяється окремий етап – дослідницька діяльність (Research and Development, R&D), що позначений синім кольором.

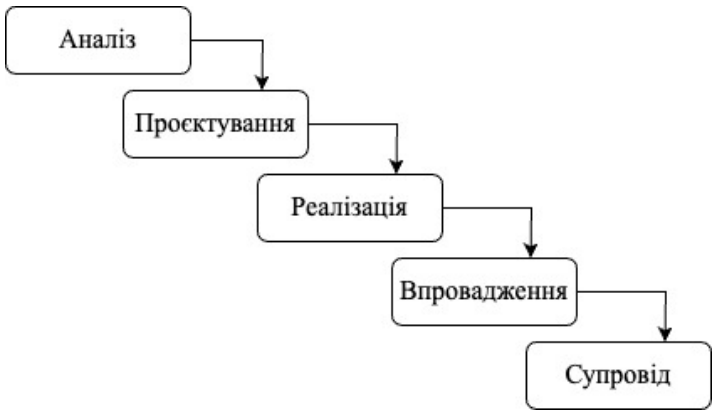



Рисунок 2.1 – Блок-схема робочого процесу розробки програмних продуктів

R&D є невід'ємною частиною Data Science-проєктів і виконує кілька ключових завдань:

1. Аналіз проблеми. Вивчення поставленого завдання для визначення основних викликів і потреб.

2. Дослідження можливих рішень. Пошук і тестування різних підходів до розв’язання проблеми.

3. Розробка моделей ШІ. Створення та навчання моделей для кожного з обраних варіантів.

4. Оцінка ефективності. Порівняння результатів різних моделей для вибору найбільш оптимального рішення.

Дослідницька діяльність є критично важливою для успіху проєктів на основі ШІ з кількох причин:

– індивідуальність кожного завдання. Кожен проєкт має свої унікальні особливості, що вимагають розробки кастомізованих моделей.

– висока складність моделей. Процес навчання нейронних мереж і оптимізації їхніх параметрів потребує глибокого аналізу та експериментів.

– ітеративний підхід. R&D дозволяє багаторазово вдосконалювати моделі, поступово досягаючи більшої точності та ефективності.

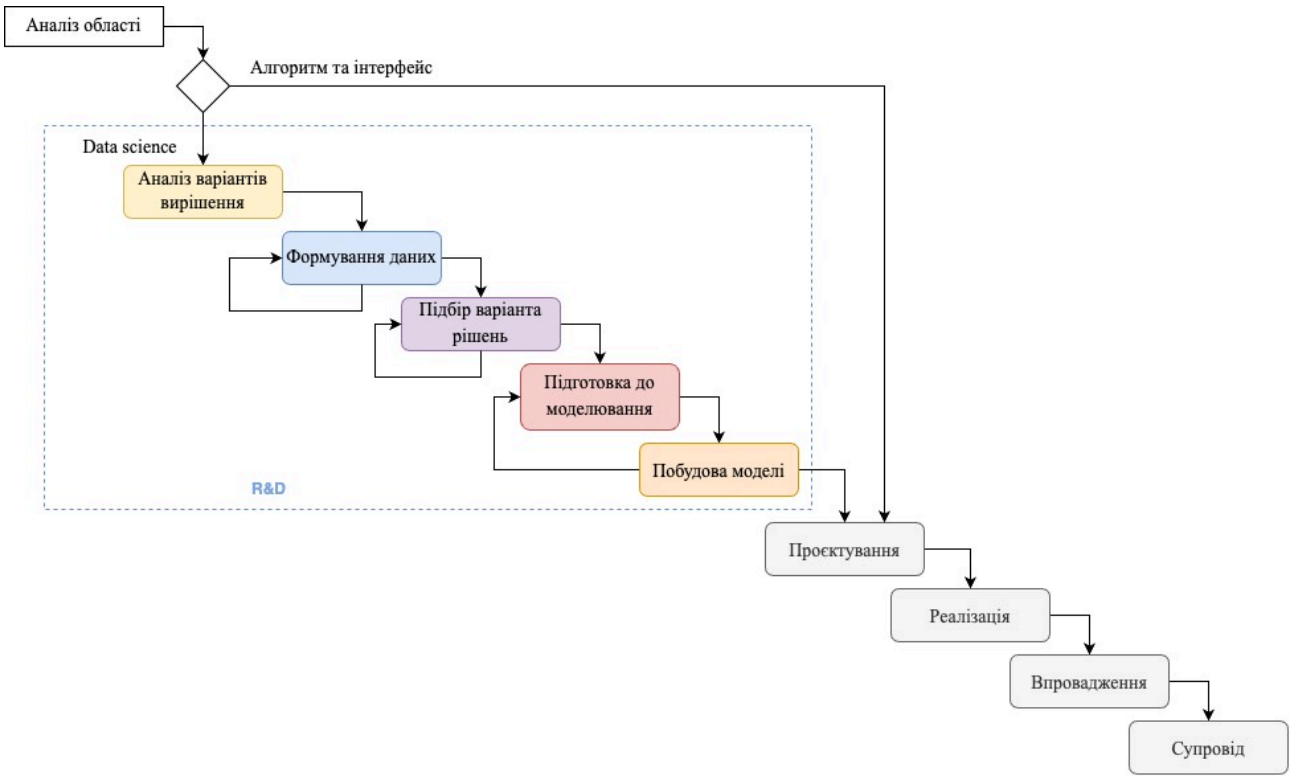



Рисунок 2.2 – Блок-схема робочого процесу розробки програмного продукту, що містить ШІ

Розробка програмних продуктів на основі штучного інтелекту (ШІ) має принципові відмінності від традиційних підходів. Зокрема, після етапу аналізу предметної області неможливо одразу переходити до проєктування кінцевого продукту, як це зазвичай робиться при вирішенні стандартних завдань. Вихідні дані, отримані на етапі аналізу, часто потребують багаторазового перегляду, уточнення та доопрацювання, що значно ускладнює процес розробки.

На рисунку 2.3 представлено уточнений алгоритм етапу R&D, який демонструє циклічний характер підготовчих робіт у ШІ-проєктах. Цей підхід відрізняється від традиційних методів тим, що життєвий цикл таких проєктів включає багаторазові ітерації. Кількість цих циклів заздалегідь передбачити неможливо, оскільки вона залежить від двох ключових факторів:

1. Складності поставленого завдання. Чим складніше завдання, тим більше ітерацій може знадобитися для досягнення оптимального результату.

2. Кваліфікації команди. Рівень підготовки та творчий підхід співробітників безпосередньо впливають на швидкість і якість пошуку рішень.

Важливо враховувати, що детальне дослідження поставленого завдання іноді може показати, що воно не має рішення у межах існуючих ресурсів чи технологій.

На рисунку 2.3 вузол, позначений *1, відображає ключовий момент дослідницького етапу, коли команда ухвалює рішення щодо подальших дій. Це може включати:

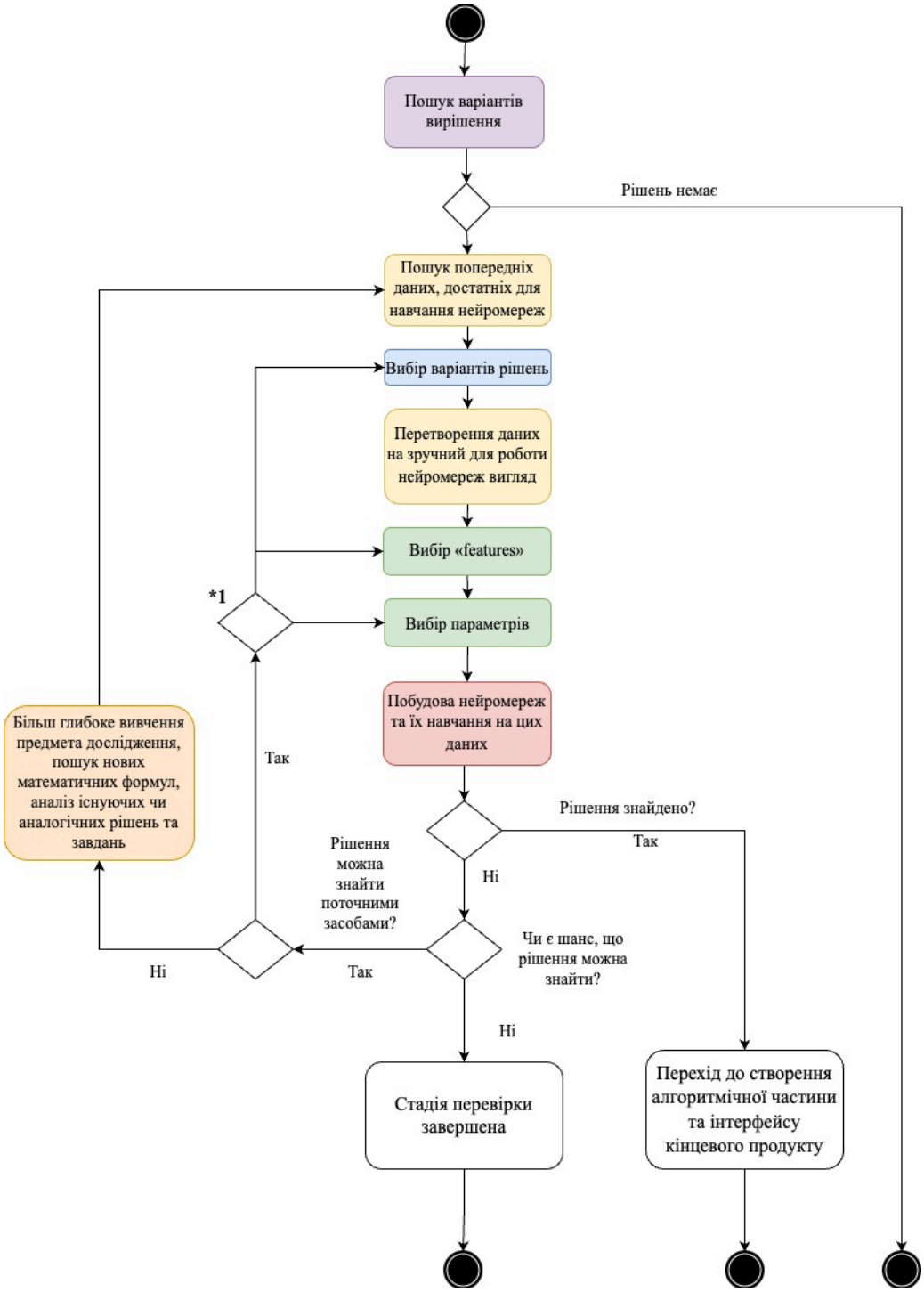



Рисунок 2.3 – Блок-схема алгоритму *R&D*

- зміну параметрів моделі;
- модифікацію ознак (features);
- пошук нового підходу до вирішення завдання.

Цей етап вимагає від команди не лише технічних знань, а й досвіду та інтуїції, які допомагають у виборі найефективнішого шляху.

Процес розробки ШІ-продуктів суттєво відрізняється від традиційних Agile-моделей (наприклад, Scrum або Kanban), у яких кожна ітерація завершується створенням певної функціональної частини кінцевого продукту. У випадку з ШІ-проєктами, завершення окремого циклу не завжди означає готовність частини або функції продукту. Навпаки, часто виникає необхідність повернутися до початкових даних і почати роботу з нуля.

Дослідницька діяльність (R&D) у ШІ-продуктах базується на опрацюванні великих масивів даних протягом усього життєвого циклу проєкту. Це вимагає використання спеціалізованих методів для роботи з Big Data. У відповідь на ці потреби компанії, які спеціалізуються на Data Science, найчастіше застосовують методологію CRISP-DM для дослідження та аналізу даних.

Розробка ШІ-продуктів поєднує два основні типи діяльності:

- дослідницькі дії, які мають унікальний і творчий характер;
- стандартні дії, пов’язані з проєктуванням, реалізацією та впровадженням.

Ця комбінація потребує гнучкого підходу до управління, оскільки дослідницькі етапи за своєю суттю значно відрізняються від традиційних інженерних процесів.

Для управління проєктами у компанії ТОВ «Антс» використовуються Jira та Confluence. Ці інструменти забезпечують повний набір функцій для управління завданнями, персоналом та документацією на стандартних проєктах. Вони чудово підходять для етапів проєктування та реалізації, але їх можливості виявляються обмеженими для ефективного управління дослідницькою діяльністю на етапі R&D.

Як показало проведене автором дослідження, використання Jira на етапі R&D у компанії є фрагментарним і неефективним. Причина полягає в тому, що Jira не має інструментів, необхідних для підтримки гнучкої та ітеративної природи дослідницьких процесів, властивих ШІ-проєктам.

## 2.2 Організація процесу дослідницької діяльності на IT підприємстві

Компанія ТОВ «Антс» має компактну, але ефективну організаційну структуру, яка включає 15 співробітників, розподілених між різними відділами:

Фінансовий відділ: 1 співробітник.

Відділ DevOps: 2 співробітники.

Фахівці з даних (Data Engineering): 2 співробітники.

Відділ R&D: 6 співробітників.

Технічний директор (CTO), виконавчий директор (CEO), Product Owner.

Розробник сайтів готового продукту: 1 співробітник.

Співробітники мають високу кваліфікацію та володіють такими знаннями та навичками.

Освіта: ступінь PhD з математики (у керівника відділу), технічна освіта в галузі прикладної математики.

Технічні знання: алгоритми та структури даних, програмування на Python, Flask, Docker Swarm, Kubernetes, TensorFlow, Cuda, C++, NodeJS, ClickHouse, MySQL, Redis, Kafka, R, Bash Script.

Фахівці DevOps володіють:

Освіта: вища математична освіта.

Навички: Docker Swarm, Kubernetes, Kibana, Grafana, Git, NginX, Lua, Bash Script.

Директори компанії мають технічну освіту та добре володіють деякими з перерахованих мов програмування та фреймворків.

Технічний директор (CTO) також виконує роль архітектора проєктів і керує всіма технічними відділами. Виконавчий директор (CEO) поєднує обов’язки Project Manager, серед яких:

- постановка завдань,
- розподіл обов’язків,
- моніторинг виконання,
- створення проєктної документації.

R&D-процес вимагає активної залученості всієї команди, включно з менеджментом, що відрізняє його від традиційних Agile-методологій. Співробітники компанії мають перехресні навички, що є важливою перевагою у розробці ШІ-продуктів.

Усі члени команди підтримують тісний зв’язок між собою та із замовником. Щоранку проводяться стенд-ап мітинги, де аналізуються результати попереднього дня і плануються завдання на поточний. На цих зустрічах також приймаються рішення щодо відхилення проміжних завдань або зміни підходу до їх вирішення.

Щотижня організовуються брейнштормі для розв’язання складних проблем. У таких обговореннях бере участь уся команда, а в разі потреби залучаються зовнішні експерти.

Одним із недоліків процесів управління є недостатнє документування результатів зустрічей. Ідеї, пропозиції та висновки часто фіксуються на стікерах, аркушах паперу або взагалі не записуються. Хоча це не критично для невеликої компанії, подібна практика може призвести до проблем із відстеженням прийнятих рішень і планування в майбутньому.

Схема організаційної структури підприємства у загальному вигляді представлена рисунку 2.4.

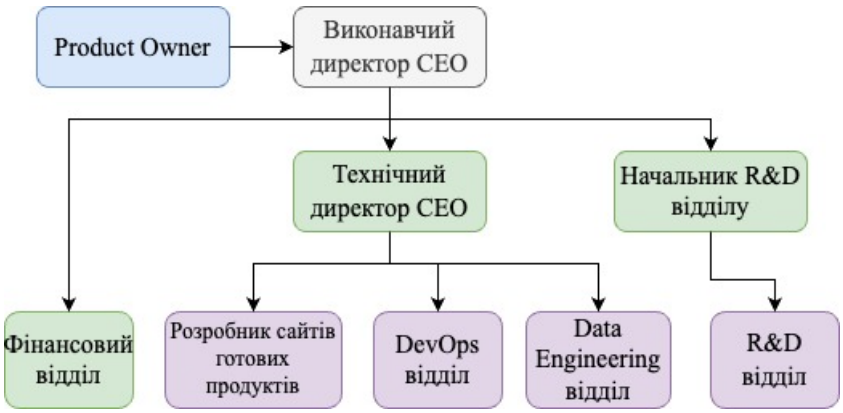



Рисунок 2.4 – Загальна організаційна структура підприємства ТОВ «Антс»

Організація процесу створення програмного продукту в компанії ТОВ «Антс» базується на чітко визначеному поділі праці між відділами та фахівцями. Особливу увагу варто звернути на етап R&D, який є ключовим у розробці інноваційних рішень.

Аналіз організації роботи показує, що процес розробки на етапі R&D має складну нелінійну структуру, що відрізняється від традиційних підходів. На рисунку 2.5 представлено детальну схему цієї структури, яка демонструє взаємозв’язок між різними етапами роботи та ролями учасників проєкту.




Рисунок 2.5 – Модель бізнес-процесів етапу *R&D* підприємства ТОВ «Антс»

Модель бізнес-процесів у компанії ТОВ «Антс» демонструє, як розподіляються робочі завдання в часі та між виконавцями, а також відображає зв’язки між ними. Рисунок 2.5 є деталізованим рівнем декомпозиції блок-схем, представлених на рисунках 2.2 і 2.3. Через складність і нелінійність процесу

дослідницької діяльності виділення чітких етапів стає проблематичним, оскільки робочі підпроцеси можуть відбуватися паралельно або перетинатися.

Процес створення та навчання нейронної мережі розподіляється між трьома основними сторонами:

1. Product Owner – приймає ключові рішення зі сторони замовника.

2. Фахівці з Data Science – відповідають за моделювання та аналітику.

3. Data Engineers – займаються обробкою та підготовкою даних.

Залежно від потреб проєкту, члени команди можуть тимчасово змінювати свої ролі, виконуючи обов'язки інших.

Як показано на рисунку 2.3, R&D-процес розпочинається з аналізу предметної області. Основними результатами цього етапу є:

– визначення мети продукту;

– формування бізнес-вимог;

– встановлення ключових показників ефективності (KPI) для кожної вимоги;

– розробка стратегії вирішення проблеми за допомогою ШІ або алгоритмічних методів.

Попередня орієнтовна діяльність є надзвичайно важливою для проєктів, що базуються на ШІ, оскільки на цьому етапі визначаються критерії, яким повинен відповідати кінцевий продукт. Формулювання потреб замовника здійснюється через постановку проблеми: наприклад, «запобігти відтоку клієнтів» або «стимулювати продажі». Виходячи з цих потреб, команда визначає функціональність продукту.

Цей етап передбачає колективне генерування ідей, де важлива участь усіх членів команди, включаючи менеджмент. Кінцевий результат цього етапу – розуміння масштабу і глибини необхідного дослідження.

Наступний крок – визначення кількісних та якісних показників, які дозволяють оцінити досягнення бізнес-вимог. Наприклад, показник CTR може бути визначений з точністю до певного відсотка, або якісні метрики можуть

відображати скорочення часу на виконання певних завдань. Усе це уточнюється під час збору даних і може коригуватися залежно від обмежень часу та ресурсів.

Чітко визначений обсяг роботи на цьому етапі є критичним, оскільки дослідницькі проєкти мають тенденцію до неконтрольованого розширення. Відсутність чітких меж і цілей може призвести до витрачання часу на розробку моделей, які не відповідають реальним потребам.

Етап аналізу рішення (бузковий колір на схемах) передбачає вивчення академічної літератури та існуючих програмних рішень. Глибина аналізу залежить від специфіки проєкту: чи є він фундаментальним, чи спрямований на вирішення конкретної проблеми.

Обрані рішення оцінюються за технічними характеристиками, складністю впровадження та відповідністю бізнес-вимогам. Це дозволяє уникнути непродуктивних витрат на розробку і забезпечує вибір оптимального варіанту.

Останнім етапом є підготовка даних (синій і жовтий колір на схемах). На цьому етапі збираються та обробляються первинні дані, які очищуються та перетворюються у формат, необхідний для навчання нейронних мереж. Ці дані зберігаються у базах даних компанії для подальшого використання.

Модель бізнес-процесів у ТОВ «Антс» поєднує нелінійну структуру роботи, гнучкий підхід до розподілу завдань та адаптацію до змінних умов. Такий підхід забезпечує ефективну реалізацію складних ШІ-продуктів, але вимагає чіткої координації на кожному етапі.

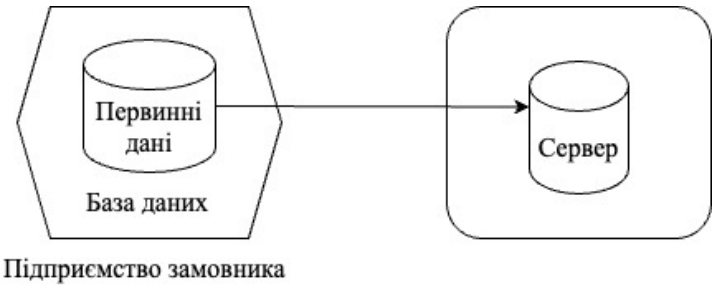



Рисунок 2.6 – Схема взаємодії баз даних

Підготовка даних. Процес підготовки даних для проєктів із використанням штучного інтелекту (ШІ) часто починається з перенесення великих обсягів інформації (Big Data) із баз даних замовника на сервер компанії, якщо це дозволяють умови конфіденційності. Після цього дані обробляються у формат, оптимальний для швидких запитів та виконання складних обчислень. Цей етап є підготовчим до процесу аналізу даних і нерідко займає більше часу, ніж передбачалося.

Після обробки розпочинається дослідження даних, на якому спеціаліст з Data Science використовує визначені раніше KPI та показники моделі. Проте важливо підтримувати баланс між дослідженням і розробкою: навіть маючи чіткі критерії, корисно розглядати альтернативні підходи, які на перший погляд можуть здаватися неактуальними.

Дані, що пройшли обробку, повинні бути зручними для використання командою Data Engineering. У процесі можуть виникнути труднощі, наприклад, недостатність або неповнота наданих замовником даних. У таких випадках інженери з даних мають знайти додаткові джерела інформації.

Слід зазначити, що хоча підготовка даних здається окремим процесом, вона часто виконується паралельно або чергується з теоретичним аналізом і дослідженням можливих рішень.

Підготовка до моделювання.

На цьому етапі багато залежить від наявного технічного забезпечення та обсягу технічної підтримки, яка доступна спеціалістам Data Science. Наприклад, вони можуть створити новий репозиторій коду, налаштувати локальний сервер Jupyter Notebook, або ж отримати доступ до потужнішого хмарного середовища для виконання складних обчислень.

Додатково спеціалісти можуть розробити код для управління версіями даних і моделей або для відстеження й керування експериментами. Якщо така функціональність уже доступна в використовуваних сервісах, необхідно лише налаштувати розподіл ресурсів і користувацькі пакети.

Цей етап також включає підготовку параметрів (features) для майбутньої нейромережі, що є важливою частиною роботи перед початком моделювання.

Розробка та навчання моделі.

На етапі розробки моделі відбувається її розгортання, навчання та постійний моніторинг результатів. У середовищі, підготовленому на попередньому етапі, створюється модель нейромережі. Data Engineering передає набір даних із запитами, сформованими відповідно до заданих параметрів, а Data Science завантажує ці дані в навчальне середовище.

Навчання моделі супроводжується моніторингом її роботи за допомогою відповідного функціоналу. Після завершення навчання модель проходить тестування.

На кожному етапі моделювання всі версії моделі та супутні масиви даних перевіряються на відповідність визначеним KPI. Це дає змогу оцінити прогрес навчання й визначити, чи досягнуто необхідного рівня точності для повної відповідності початковим бізнес-вимогам.

Такий підхід забезпечує системний контроль за якістю роботи моделі, дозволяючи визначити, коли її можна вважати готовою до впровадження.

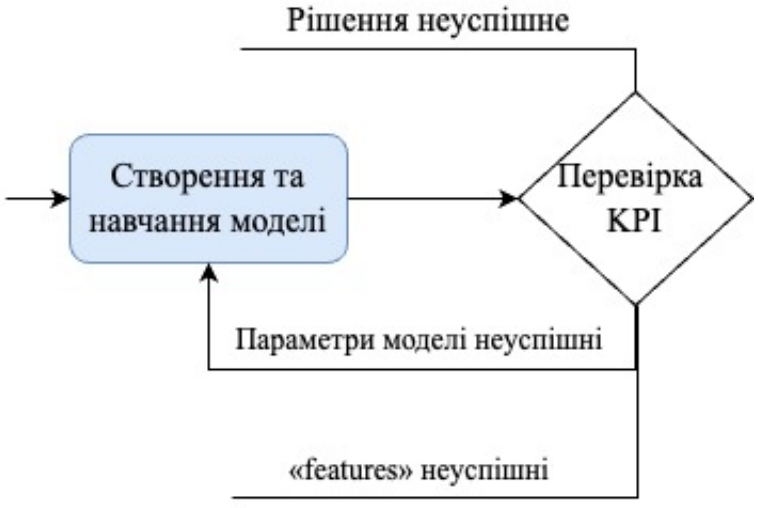



Рисунок 2.7 – Алгоритм ухвалення рішення про успішність моделі

Якщо результати навчання нейронної мережі не досягають очікуваних показників, це може бути зумовлено кількома причинами. Спеціаліст відділу

Data Science аналізує результати та приймає рішення щодо подальших дій: модифікувати параметри моделі, змінити набір характеристик (features) або перейти до альтернативного варіанту моделі, якщо поточна не здатна вирішити поставлену задачу.

У деяких випадках невдале навчання моделі вимагає перегляду цілей продукту. Невеликі зміни у формулюванні цілей можуть спричинити незначні корективи технічної реалізації, але загалом це може означати повернення до фази аналізу предметної області. У найекстремальніших випадках може бути прийняте рішення про повне скасування проєкту. Це відбувається тоді, коли фахівці впевнені, що всі можливі напрями досліджень були вичерпані, а проєкт більше не має перспектив. У таких випадках команда переходить до роботи над новим проєктом.

Якщо модель успішно досягла заданих KPI, етап R&D можна вважати завершеним. Після цього команда переходить до розробки алгоритмічної частини продукту та створення користувацького інтерфейсу.

У ситуаціях, коли отримані значення KPI відхиляються від заданих, але залишаються в межах допустимого, необхідно провести їх детальний перегляд та узгодження з Product Owner. Важливо підтвердити, що показники перебувають у межах прийнятного розкиду, враховуючи технічні вимоги до продукту та інтереси клієнта. Коли Product Owner вважає, що модель відповідає заявленим цілям проєкту, команда може розпочати створення кінцевого програмного продукту.

Аналіз процесу дослідницької діяльності у компанії ТОВ «Антс» показав, що його структура відрізняється нелінійністю. Хоча в основі життєвого циклу проєкту використовується Agile-модель, вона не завжди має лінійну архітектуру. Це підтверджує схема, наведена на рисунку 2.8, яка ілюструє складність і багатогранність процесів R&D.

Таке відхилення від традиційної лінійної моделі свідчить про необхідність гнучкого підходу до управління проєктами, що дозволяє ефективно адаптуватися до змін і вирішувати складні завдання у сфері штучного інтелекту.

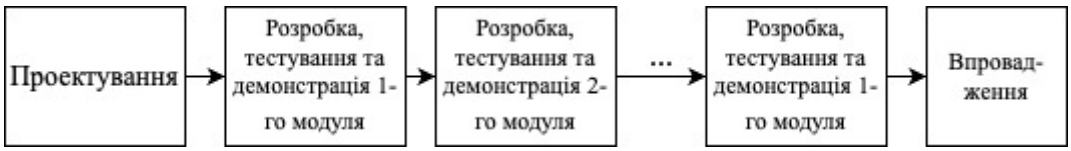



Рисунок 2.8 – Розгортка *Agile* -моделі життєвого циклу проекту

На відміну від Agile-моделей, життєвий цикл проєкту в моделі CRISP-DM можна представити у вигляді дерева рішень, де кожна гілка символізує новий цикл розробки (рис. 2.9). У цій структурі кореневий вузол представляє основні межі проєкту, включаючи визначені цілі, бізнес-вимоги та набір ключових показників ефективності (KPI).

На другому рівні дерева розташовані вузли, що відображають знайдені або адаптовані можливі рішення для поставленої задачі. Від цих вузлів відходять гілки, що представляють розроблені характеристики (features), необхідні для створення моделі. Кожен вузол, що відповідає конкретній характеристиці, розгалужується на параметри, які необхідно вибрати для оптимального налаштування моделі.

Кожен вузол у дереві рішень можна співвіднести з однією з фаз етапу R&D, розглянутих раніше. Наприклад, на першій фазі формуються бізнес-вимоги та KPI, які складають основу кореневого вузла. Далі, на наступних фазах, розробляються можливі рішення та адаптуються їхні характеристики, що створює відповідні вузли другого рівня та розгалуження. Така структура дозволяє систематизувати процеси розробки та забезпечити послідовний перехід між фазами.

Представлення моделі CRISP-DM у вигляді дерева рішень дає змогу краще візуалізувати процес дослідження та розробки. Це допомагає ефективно координувати команду, забезпечує прозорість прийняття рішень і дозволяє швидко вносити корективи на будь-якому етапі проєкту.

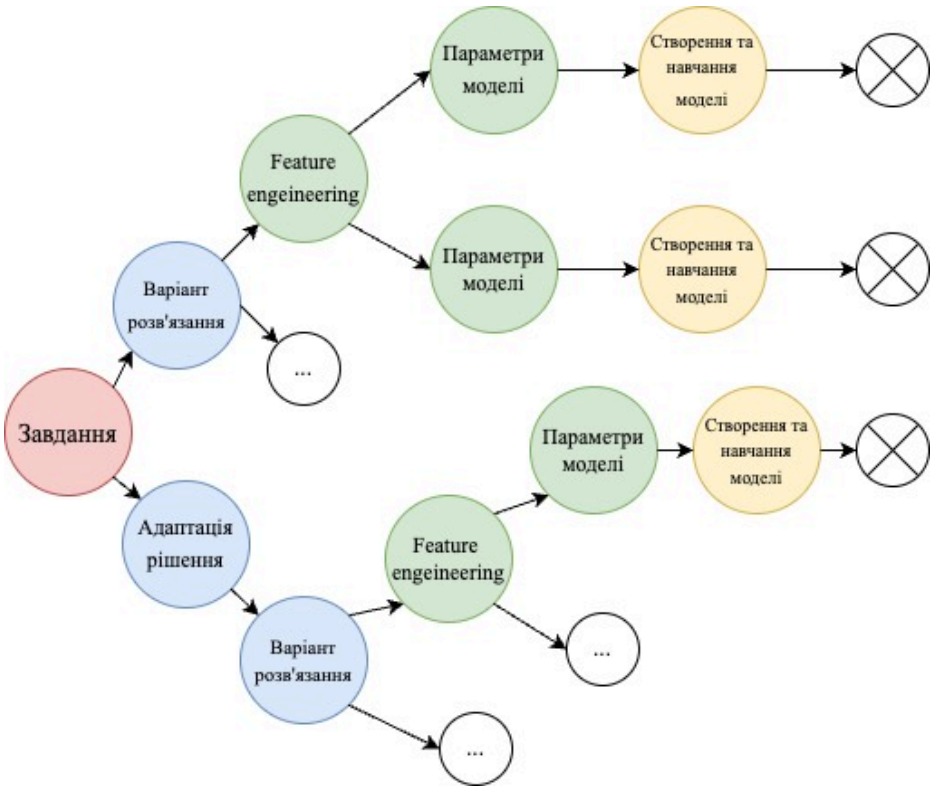



Рисунок 2.9 – Дерево рішень моделі *Crisp DM*

## 2.3 Модель системи управління проєктами «*As Is*»

Аналіз процесу розробки програмного продукту з елементами штучного інтелекту та організаційної структури компанії ТОВ «Антс» дозволив створити модель «As Is» («Як є»), що відображає поточний стан процесів управління.

Побудована модель демонструє управлінські дії, які здійснюються за допомогою інструментів Jira і Confluence. Вона також враховує дії інших співробітників компанії, які виконують функції Project Manager у залежності від вимог конкретного проєкту. Модель охоплює всі аспекти управління, що супроводжують фази R&D-діяльності.

Процеси управління в моделі «As Is» розподіляються між трьома ключовими учасниками:

1. Project Manager – відповідає за координацію проєкту, визначення завдань і контроль їх виконання.

2. Відділ Data Science – займається створенням і навчанням моделей штучного інтелекту, а також аналізом даних.

3. Відділ Data Engineering – забезпечує підготовку та обробку даних, створює інфраструктуру для зберігання та обробки великих обсягів даних.

Модель «As Is» (рис. 2.10) дозволяє візуалізувати поточний стан процесів управління, виявити сильні та слабкі сторони організаційної структури та знайти шляхи для оптимізації. Вона також забезпечує прозорість і спрощує комунікацію між учасниками проєкту на всіх етапах розробки продукту.

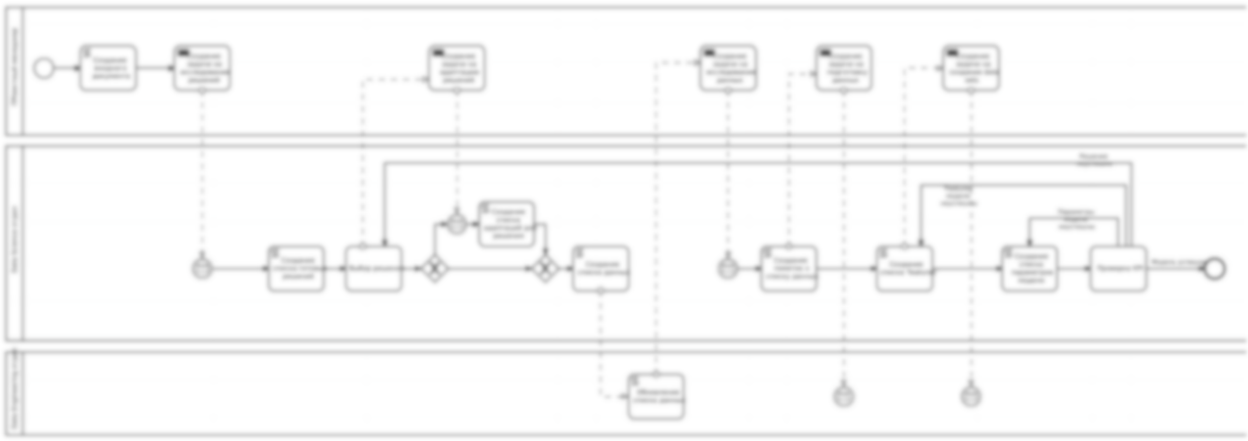


Рисунок 2.10 – Модель управління процесами *As Is*

Управління етапом R&D у ТОВ «Антс» можна описати через модель, що відображає послідовність управлінських дій і документообіг.

Основні етапи управління:

1. Створення вхідного документа. Цей етап належить до фази аналізу предметної області. На нараді команда визначає цілі, бізнес-вимоги та KPI, а також уточнює, які з них вирішуватимуться за допомогою ШІ чи алгоритмів. Усі записи з дошки чи листка PM переносить у Confluence.

2. Створення завдання з вивчення рішень. Після наради PM створює завдання у Jira для співробітника Data Science, який починає дослідження.

3. Складання списку готових рішень. Data Scientist готує презентацію з можливими рішеннями, яку команда обговорює на наступній зустрічі.

4. Вибір перспективного рішення. Команда обговорює презентацію, визначає перспективні рішення, відхиляє непридатні та залишає деякі «про запас».

5. Адаптація обраного рішення. Якщо жодне з рішень не підходить у готовому вигляді, PM створює завдання в Jira на адаптацію, яке виконує Data Scientist.

6. Створення списку адаптацій. Співробітник формує список адаптацій і зберігає його в Confluence.

7. Складання списку необхідних даних. Команда створює перелік даних для проєкту, який також зберігається в Confluence.

8. Оновлення списку даних. Data Engineer займається збором даних, робить позначки про наявність інформації.

9. Дослідження даних. Начальник Data Science аналізує зібрані дані, створює документ із позначками та зберігає його в Confluence.

10. Підготовка даних. PM створює завдання в Jira, яке Data Engineer виконує, очищуючи дані та готуючи їх для моделювання.

11. Формування списку «features». Команда визначає перелік «features» для навчання моделі та додає його до Confluence.

12. Розробка параметрів моделі. Data Science розробляє список параметрів для моделі на основі зібраних даних.

13. Перевірка KPI. Після навчання моделі команда перевіряє її результати за допомогою Jupyter Notebook, однак показники поточного циклу часто не документуються.

Проблеми та недоліки.

Неструктурована документація: більшість документів створюється на розсуд виконавців, без дотримання єдиної структури.

Недостатнє використання корпоративних систем: частина документації ведеться поза Confluence або у паперовому вигляді.

Відсутність централізованого контролю: PM не має можливості оперативно отримати повну картину стану проєкту. Більшість даних передаються в усній формі.

Інформаційна безпека: через відсутність централізованого документообігу інформація може втрачатися або залишатися незахищеною.

Організація процесів управління у ТОВ «Антс» потребує оптимізації. Впровадження жорсткіших правил документообігу, використання Confluence та Jira для збереження всіх даних і завдань, а також підвищення прозорості процесів допоможе покращити управління проєктами.

## 2.4. Висновки до розділу 2

Дослідження методів управління проєктами та алгоритму R&D-діяльності у компанії ТОВ «Антс» дозволило виокремити низку проблем:

1. Неповна документація. У процесі дослідницької діяльності документування часто залишається поза увагою через специфіку обов’язків. Зазвичай створення документації покладається на менеджерів та бізнес-аналітиків, однак технічна документація потребує залучення спеціалістів Data Science та Data Engineering. Через брак часу або небажання розробників виконувати нетипові для них завдання документація створюється неякісно або взагалі не створюється. У результаті цінні ідеї та творчі рішення можуть бути втрачені, що призводить до необхідності повторних обговорень, додаткових досліджень та витрат часу.

2. Неуніфікований формат документації. Через відсутність єдиних стандартів оформлення документація створюється у різних форматах, що ускладнює її спільне використання та аналіз. Це може впливати на ефективність командної роботи та доступність інформації.

3. Слабка візуалізація процесів. У компанії відсутня система візуалізації дерева рішень та їх зв’язків. Це створює труднощі для відстеження прогресу проєкту, ускладнює підготовку звітів для замовників та загальне розуміння стану справ.

4. Неповний облік часу. Через повторюваність завдань, їхню незначну тривалість та компактність команди частина задач не фіксується у Jira. Це ускладнює точний облік витраченого часу, що може спричинити «втрату» завдань і проблеми у звітуванні.

Модель CRISP-DM, яка є основою для Data Science проєктів, демонструє свою ефективність, однак має певні обмеження:

1. Відсутність підтримуючих засобів. Ця модель адаптована до Data Science порівняно недавно, і наразі не існує достатньо інструментів для ефективного управління проєктами такого типу. Існуючі платформи ще не адаптовані, а нові рішення не встигли з’явитися на ринку.

2. Відсутність обліку часу. Модель не враховує часові рамки виконання окремих циклів, що може ускладнювати планування та контроль за дотриманням термінів.

Для ефективного управління проектами у ТОВ «Антс» необхідно вдосконалити процес документування, стандартизувати формати, впровадити інструменти візуалізації процесів та забезпечити точний облік часу. Додатково варто розглянути можливість адаптації існуючих інструментів або створення нових, які відповідатимуть потребам Data Science-проєктів.

# РОЗДІЛ 3
# ПРОЄКТУВАННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ УПРАВЛІННЯ ПРОЄКТАМИ КОМПАНІЇ ТОВ «АНТС»

## 3.1. Модель автоматизованої системи управління проєктами підприємства ТОВ «Антс»

Дослідження організації управління проектами в ТОВ «Антс» виявило значний потенціал для підвищення ефективності. Чинна система, яка базується на ручному підході, не повністю відповідає сучасним вимогам створення програмних продуктів.

Для успішної роботи проектної команди необхідний єдиний інформаційний центр, який дозволить вирішувати такі управлінські завдання:

- зберігати всю проектну документацію;
- вести протоколи нарад, стендапів, мозкових штурмів та виробничих зустрічей;
- зберігати презентації доповідей співробітників;
- інформувати команду про плани та події;
- створювати реєстри завдань, бізнес-процесів та власних розробок;
- формувати бібліотеку готових рішень для майбутнього використання;
- зберігати резерв ідей, методів та процедур;
- призначати завдання і контролювати їх виконання;
- забезпечувати зворотний зв’язок щодо виконання завдань.

Метою даного етапу дослідження є розробка моделі такого інформаційного центру у вигляді автоматизованої системи управління проектами (АСУП). Ця система буде інтегрована в існуючі інструменти Jira та Confluence для оптимізації процесів та підвищення загальної ефективності управління проектами.

Аналіз показав, що управління на етапі R&D суттєво відрізняється від етапу реалізації програмного продукту. Тому автоматизована система повинна враховувати специфіку R&D-процесів, зберігаючи сумісність з існуючими інструментами.

Запропонована система не є самостійним продуктом, а виступає плагіном до Jira, що дозволить автоматизувати рутинні процеси без порушення поточної структури.

Для інтеграції АСУП можна застосовувати різні методи адаптації:

1. Параметрична адаптація. Передбачає налаштування параметрів програми, що дозволяє обирати стратегії поведінки із запропонованого набору.

2. Функціональна адаптація. Зміна функціональності в межах існуючої структури програмного забезпечення.

3. Організаційна адаптація. Перерозподіл внутрішніх ресурсів системи та зміна потоків процесів без зміни структури.

4. Структурна адаптація. Модифікація структури системи або заміна окремих її елементів.

5. Розмноження. Створення нових версій системи з оптимізованими ресурсами та здатністю до змін.

6. Розвиток. Еволюційний процес адаптації програмного забезпечення через накопичення знань про середовище, завдання та власну структуру.

Система управління проектами в ТОВ «Антс» має стати окремим функціональним модулем, інтегрованим у Jira та Confluence. Ця система буде використовувати їхні функціональні можливості, що дозволить значно вдосконалити існуючу модель управління, яка наразі поєднує ручні та автоматизовані елементи.

У нинішній моделі «As Is» значна частина документування та постановки завдань покладається на проектну команду та проектного менеджера. Впровадження R&D-плагіну автоматизує ці процеси. Завдяки вбудованому алгоритму R&D, система автоматично генеруватиме шаблони документів та створюватиме завдання на кожному етапі управління. Відстеження дій проектної

команди дозволить генерувати відповідні доручення без додаткових ручних операцій.

Таким чином, нова система «To Be» розширює функціональні можливості, не порушуючи загальної структури існуючої моделі.

Із впровадженням R&D-плагіну змінюється траєкторія інформаційних потоків. Взаємодія між проектним менеджером і відділами компанії тепер проходитиме через єдиний інформаційний центр. При цьому точки входу та виходу даних залишаються незмінними, але інформаційні процеси стають більш ефективними.

Структура системи управління проектами також зазнає змін. Методи ручного управління більше не є обов'язковими, а в системі Jira з'являється новий компонент – R&D-плагін.

На основі структурно-функціонального методу адаптації було розроблено модель автоматизованої системи управління (АСУ) для ТОВ «Антс». Ця модель представлена кількома діаграмами:

- верхньорівнева діаграма екосистеми R&D-плагіну;
- діаграма бізнес-процесів «To Be»;
- діаграма прецедентів;
- ER-діаграма.

Верхньорівнева діаграма екосистеми R&D-плагіну (рис. 3.1) демонструє, як плагін взаємодіє з існуючими підсистемами та показує зв'язки між ними.

На рисунку 3.2 представлено модель взаємодії системи управління проектами «To Be» із учасниками R&D-етапу. У цій моделі з'являється новий учасник – R&D-плагін. Процеси, які раніше виконувалися учасниками проекту, тепер автоматизуються і виконуються плагіном.

R&D-плагін бере на себе частину обов'язків проектної команди, що значно скорочує час, необхідний для створення завдань, підготовки документації та відстеження прогресу. Це дозволяє команді зосередитися на ключових аспектах розробки, підвищуючи загальну продуктивність і якість проекту.

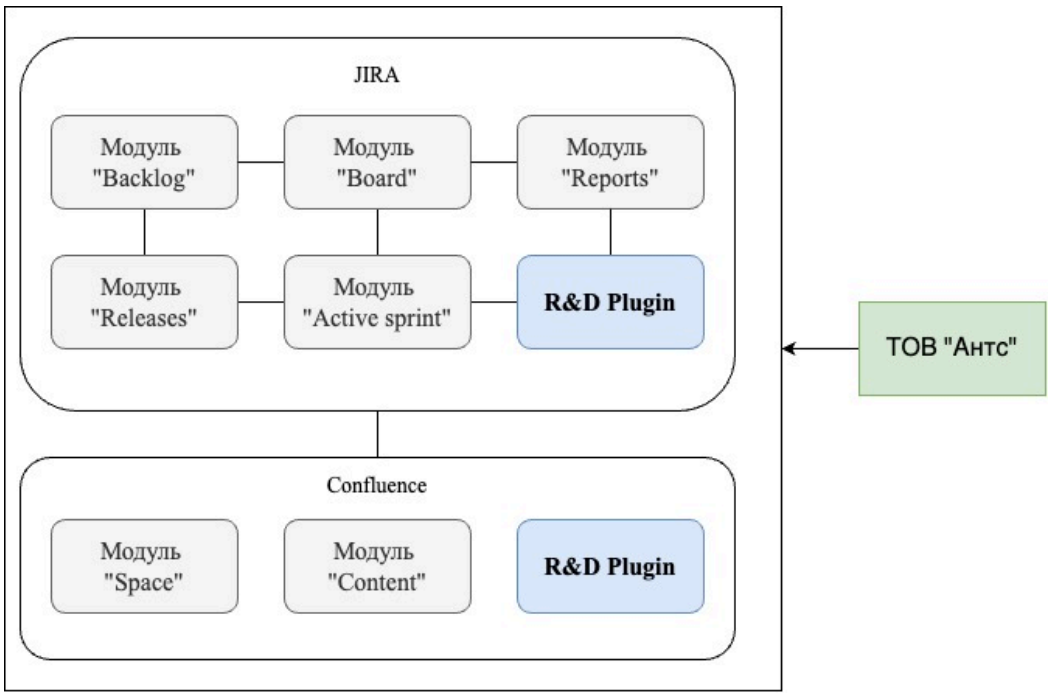



Рисунок 3.1 – Діаграма екосистеми плагіна, що вбудовується, для управління проєктами на етапі *R&D*

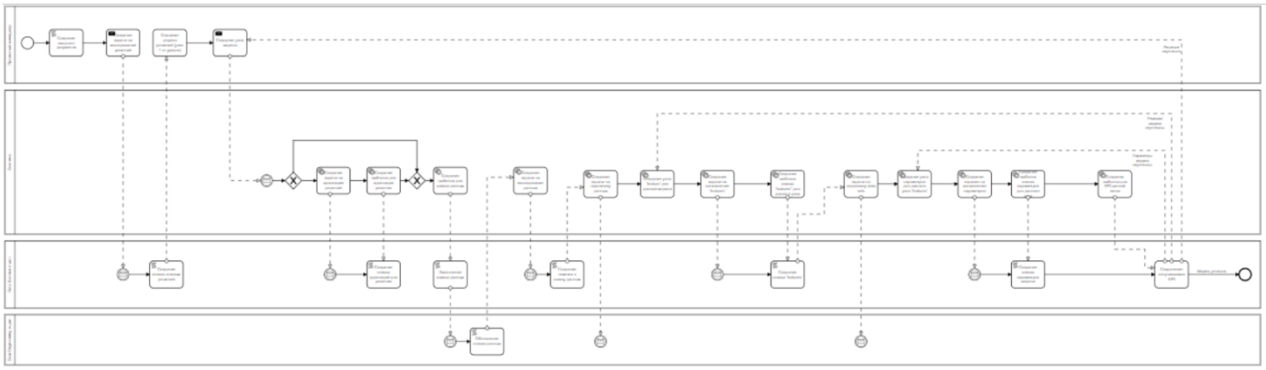


Рисунок 3.2 – Діаграма бізнес-процесів моделі *To Be* управління проєктами на етапі R&D

Діаграма бізнес-процесів управління проектами R&D To Be включає такі етапи:

1. Створення вхідного документа. На першому етапі проєктний менеджер (PM) ініціює створення нового проекту. Плагін пропонує заповнити шаблон вхідного документа, в якому фіксуються цілі проєкту, список бізнес-вимог, коментарі щодо їх реалізації (за допомогою II чи алгоритмічного підходу), а також набір KPI для майбутнього продукту.

2. Завдання на вивчення рішень**.** Після завершення наради PM завершує створення вхідного документа, перевіряє його на коректність і підтверджує новий проект у Jira. Плагін автоматично створює проєкт у Jira, додає завдання на дослідження рішень і зберігає протокол у Confluence. PM призначає відповідального з відділу Data Science, який розпочинає виконання завдання.

3. Шаблон списку рішень. Плагін автоматично створює шаблон для переліку можливих рішень у Confluence. Посилання на цей шаблон додається до завдання, створеного у попередньому етапі.

4. Перелік готових рішень. Співробітник Data Science заповнює шаблон, формуючи список знайдених рішень. Після обговорення на нараді пріоритети кожного рішення уточнюються, а також визначаються ті, що потребують адаптації.

5-6. Створення кореневого вузла та вузлів 1-го рівня. PM створює кореневий вузол дерева рішень у плагіні. Потім, на основі пріоритетів, плагін формує вузли першого рівня для моделі.

7. Завдання на адаптацію рішення. Якщо рішення з вузла не може бути використане у готовому вигляді, плагін створює завдання в Jira на його адаптацію, яке PM закріплює за співробітником Data Science.

8-9. Шаблон та список адаптацій. Плагін генерує шаблон для документа з адаптації рішення у Confluence**.** Співробітник заповнює цей документ, відзначаючи всі необхідні адаптації.

10-12. Шаблон та список даних. Після вибору рішення плагін створює шаблон для переліку необхідних даних. Співробітники Data Science заповнюють список, а Data Engineering відповідає за збирання даних, відзначаючи прогрес у документі.

13-14. Завдання на дослідження даних. Плагін створює завдання у Jira для начальника Data Science на дослідження зібраних даних. Начальник заповнює документ у Confluence із коментарями до кожного набору даних.

15. Завдання на підготовку даних. АСУ автоматично створює завдання на підготовку даних. PM закріплює завдання за співробітником Data Engineering, який працює з даними на основі створеного документа.

16-19. Вузол «features» та список параметрів. Плагін генерує вузол «features» для моделі. Завдання зі створення списку «features» додається до Jira, а співробітники Data Science заповнюють шаблон у Confluence. Після завершення цього етапу начальник Data Science підтверджує готовність списку у плагіні.

20. Завдання на підготовку Data Sets**.** Після формування списку «features» плагін створює завдання на підготовку Data Sets, яке додається до Jira разом із відповідним документом. PM призначає виконавця для виконання цього завдання.

21. Створення вузла параметрів для вузла «features». Після створення завдання підготовки Data Sets (п.20), плагін автоматично генерує новий вузол у системі – вузол параметрів. Цей вузол стає основою для подальшого налаштування моделі.

22. Завдання на складання параметрів. На основі створеного вузла параметрів АСУ формує нове завдання в Jira для складання списку параметрів моделі. PM може призначити відповідального за виконання цього завдання.

23. Шаблон списку параметрів. Плагін створює шаблон списку параметрів у Confluence. Посилання на цей шаблон автоматично додається до завдання, створеного у п.22, що спрощує доступ до необхідного формату документації.

24. Заповнення списку параметрів. Співробітники відділу Data Science заповнюють шаблон із параметрами моделі, що відповідають поточному етапу розробки. Цей список стає основою для створення моделі та її навчання на підготовлених наборах даних (Data Sets). Після завершення навчання модель аналізується на відповідність KPI продукту.

25. Шаблон для KPI. Після завершення роботи над параметрами моделі (п.24), плагін автоматично створює шаблон у Confluence для збереження показників KPI, що відображають результати поточного етапу навчання.

26. Збереження KPI. Співробітник Data Science заповнює шаблон із показниками KPI, отриманими після навчання моделі. Цей документ використовується для аналізу відповідності результатів встановленим бізнес-вимогам.

Після виконання п.26 проектна команда аналізує досягнуті KPI. Якщо модель відповідає вимогам, етап R&D вважається завершеним. В іншому випадку команда вирішує, до якого етапу потрібно повернутися:

– створити нові параметри для моделі, використовуючи поточний набір «features» (повернення до п.21);

– сформувати новий набір «features» (повернення до п.16);

– вибрати інше можливе рішення (повернення до п.6).

На рисунку 3.3 представлено діаграму прецедентів (Use Case), яка демонструє взаємодію учасників етапу R&D із R&D Plugin. Система автоматизує процеси, що раніше виконувалися вручну:

– створення та структурування документів;

– генерування завдань;

– відстеження виконання етапів R&D;

– своєчасне створення шаблонів документації.

Це дозволяє не тільки спростити управління документацією, але й забезпечити її уніфікованість. Завдяки візуалізації фаз дослідження та статусів завдань система підвищує прозорість і ефективність управління проектами.

Для виконання структурного аналізу моделюваної системи було використано метод «сутність-зв'язок» (Entity-Relationship, ER-діаграма). Цей підхід дозволяє наочно відобразити об’єкти, що використовуються в системі, та способи їхньої взаємодії. На рисунку 3.4 представлена ER-діаграма автоматизованої системи управління (АСУ) ШІ-проєктами на етапі R&D.

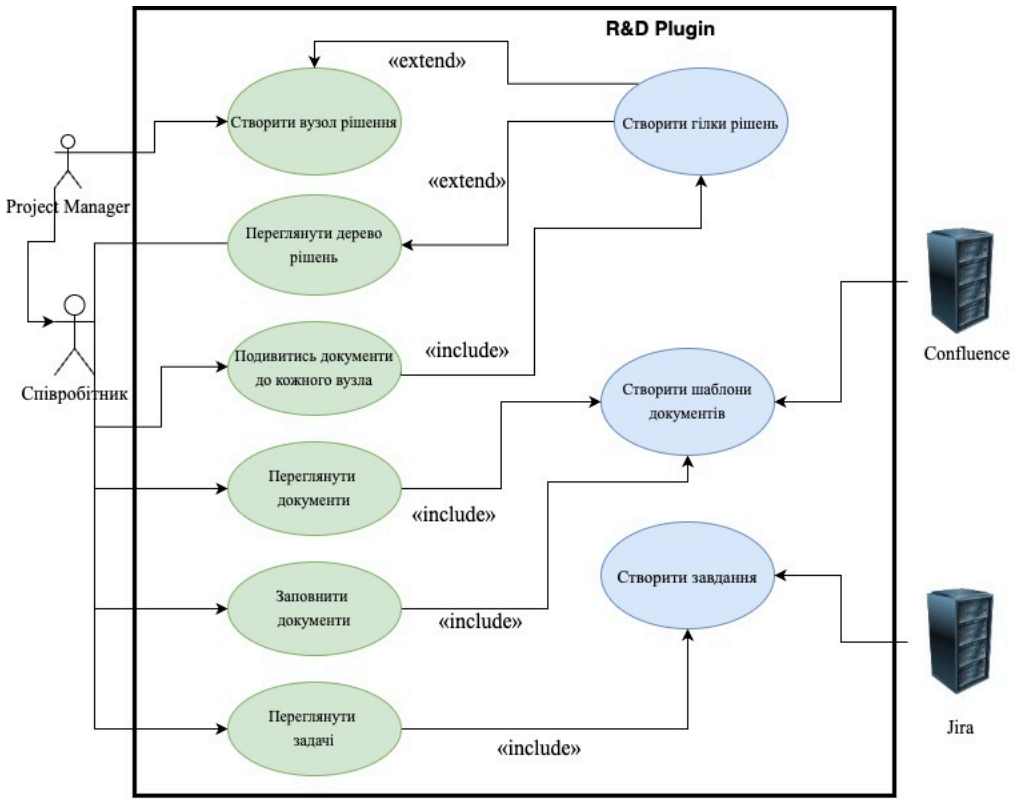



Рисунок 3.3 – Діаграма прецедентів для моделі *To Be* управління проєктами на етапі *R&D*

ER-діаграма ілюструє статичну взаємодію між ключовими елементами системи: проєктами**,** документами**,** завданнями та гілками рішень. Ці сутності є основою системи, і їхні зв'язки формують логічну структуру, необхідну для ефективного функціонування управлінських процесів.

Дана модель дозволяє систематизувати інформацію про кожен елемент, визначити їхні атрибути та встановити зв'язки між ними, що сприяє оптимізації роботи системи. Це забезпечує цілісність даних, спрощує відстеження статусу завдань та актуальність документації, а також дозволяє швидко переходити між етапами розробки.

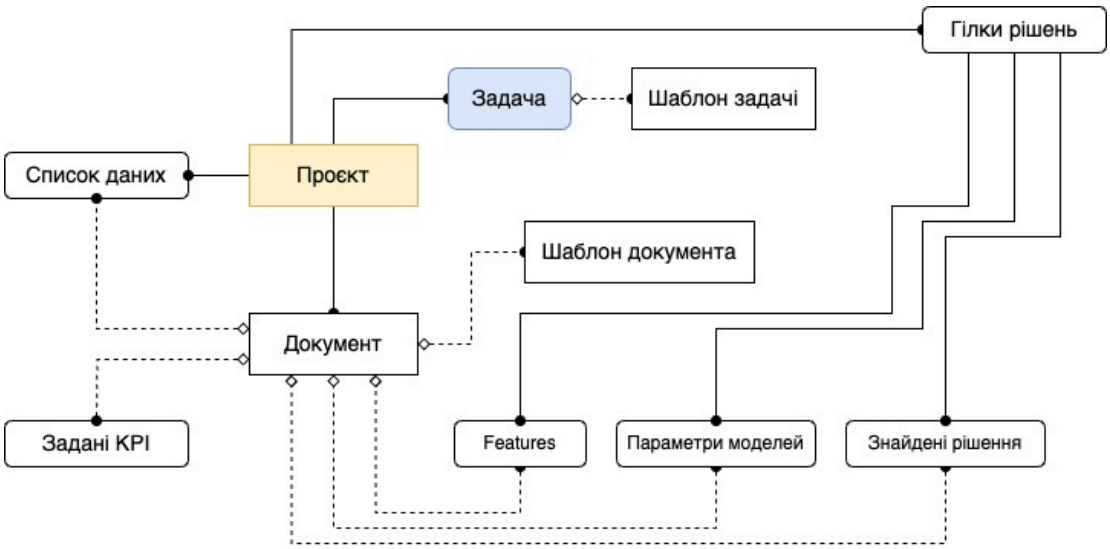



Рисунок 3.4 – *ER*-діаграма моделі «*To Be*» управління проєктами на етапі *R&D*

Сукупність представлених діаграм – діаграми екосистеми, бізнес-процесів, прецедентів та ER-діаграми – забезпечує комплексний опис моделі To Be інформаційної системи управління ШІ-проєктами на етапі R&D у компанії ТОВ «Антс».

Діаграма екосистеми наочно демонструє взаємодію між новою системою та вже існуючими підсистемами, відображаючи загальну архітектуру інтегрованого середовища. Діаграма бізнес-процесів деталізує основні управлінські процедури, необхідні для ефективного управління проєктами, і відображає їхній розподіл між учасниками.

Діаграма прецедентів показує функціональне призначення системи, акцентуючи увагу на взаємодії користувачів із плагіном. Вона також дозволяє ідентифікувати межі системи та автоматизовані процеси, що спрощують виконання завдань і створення документації.

ER-діаграма, у свою чергу, забезпечує структурний аналіз системи, відображаючи взаємозв'язки між ключовими об'єктами, такими як проєкти, документи, завдання та гілки рішень. Ця діаграма створює статичну основу для подальшого вдосконалення процесів управління.

У сукупності ці діаграми формують цілісну модель системи управління, яка сприяє автоматизації рутинних процесів, оптимізує діяльність проектної команди та забезпечує прозорість управління етапом R&D.

## 3.2. Якісна оцінка моделі автоматизованої системи управління проєктами підприємства ТОВ «Антс»

Для якісної оцінки запропонованої моделі автоматизованої системи управління проектами (АСУ) було використано такі методи:

– аналіз ступеня автоматизації бізнес-процесів шляхом порівняння моделей *As Is* і *To Be*;

– експертні оцінки для оцінки практичності та ефективності запропонованої моделі.

Для визначення сильних і слабких сторін розробленої моделі, оцінки її потенціалу та можливих ризиків також проведено SWOT-аналіз.

Оцінка ступеня автоматизації виконувалася за методикою, описаною Корабльовим І.Г. Ця методика передбачає підрахунок кількості рутинних або ручних операцій, які виконуються в поточній моделі управління (*As Is*), та їх кількості після впровадження плагіна (*To Be*).

Результати аналізу показали, що найбільша кількість рутинних дій пов’язана з двома процесами:

1. Створення завдань у системі Jira**.** Цей процес є трудомістким, оскільки на етапі R&D більшість завдань є короткостроковими, але їх обсяг значний. Витрачати час на створення таких завдань у ручному режимі іноді недоцільно.

2. Створення документації у Confluence. Формування документації є не лише рутинним, а й трудомістким завданням, особливо зважаючи на те, що система Confluence не є оптимально зручною для таких процесів.

Автоматизація цих процесів за допомогою запропонованого плагіна дозволить значно знизити навантаження на проектного менеджера та інші члени команди, підвищуючи ефективність управління проектами на етапі R&D.

Таблиця 3.1 – Шкала рівнів автоматизації

| Рівень автоматизації (LOA) | Опис |
|---|---|
| 1 | Комп'ютер не пропонує допомога: людина повинен приймати все рішення і виконувати все дії сам |
| 2 | Комп'ютер пропонує людині повний набір рішень/дій,альтернативи (Приклад – робота з електронним довідником) |
| 3 | Комп'ютер пропонує повний набір рішень/дій, альтернативи і звужує вибір до кількох варіантів |
| 4 | Комп'ютер пропонує одну альтернативу |
| 5 | Комп'ютер пропонує одну альтернативу і автоматично виконує це пропозиція, якщо людина погоджується |
| 6 | Комп'ютер пропонує одну альтернативу і виконує це пропозиція,якщо людина протягом обмеженого часу не накладає вето на автоматичне виконання операції |
| 7 | Комп'ютер виконує операції автоматично, обов'язково інформуючи людини |
| 8 | Комп'ютер виконує операції автоматично інформує людини,тільки якщо комп'ютер запитають |
| 9 | Комп'ютер виконує операції автоматично інформує людини,тільки якщо він (комп'ютер) вирішить |
| 10 | Комп'ютер вирішує все і діє автономно, не зважаючи налюдини |

Таблиця 3.2 – Ступінь автоматизації процесів створення задачі та створення документів *As Is*

| Назва процесу | Опис операцій в даному процесі | LOA |
|---|---|---|
| Створення документа | Відкрити Confluence | 1 |
| | Відкрити потрібну папку | 1 |
| | Створити документ | 1 |
| | Створити структуру документа | 1 |
| | Заповнити документ | 1 |
| | Опублікувати документ | 1 |
| Створення завдання | Відкрити Jira | 1 |
| | Відкрити Backlog | 1 |
| | Створити завдання | 1 |
| | Описати завдання | 1 |
| | Призначити виконавця | 1 |
| | Перенести в поточний спринт | 1 |

Таблиця 3.3 – Ступінь автоматизації для процесу *As Is* на одній повній гілці дерева рішень

| Вид операції | Кількість операцій в задачі | Кількість завдань на одній гілці | Загальна кількість операцій на етапі R&D для однієї гілки | Відсоток операцій на етапі R&D |
|---|---|---|---|---|
| Процес створення завдання | | | | |
| Ручні операції | 6 | 8 | 48 | 100% |
| Автоматичні операції | - | - | - | - |
| Процес створення документа | | | | |
| Ручні операції | 6 | 6 | 36 | 100% |
| Автоматичні операції | - | - | - | - |

Як видно з таблиці 3.3, процеси створення завдання та створення документа є ручними на 100%.

Побудуємо таблиці, аналогічні таблицям 3.2 та 3.3, для моделі *To Be*.

Таблиця 3.4 – Ступінь автоматизації процесів створення завдання та створення документів *To Be*

| Назва процесу | Опис операцій в даному процесі | LOA |
|---|---|---|
| Створення документа | Відкрити Confluence | 1 |
| | Відкрити потрібну папку | 10 |
| | Створити документ | 10 |
| | Створити структуру документа | 10 |
| | Заповнити документ | 1 |
| | Опублікувати документ | 1 |
| Створення завдання | Відкрити Jira | 10 |
| | Відкрити Backlog | 10 |
| | Створити завдання | 10 |
| | Описати завдання | 10 |
| | Призначити виконавця | 1 |
| | Перенести в поточний спринт | 10 |

Таблиця 3.5 – Ступінь автоматизації процесу *To Be* на одній повній гілці дерева рішень

| | Кількість операцій в задачі | Кількість завдань на одній гілці | Загальна кількість операцій на етапі R&D для однієї гілки | Відсоток операцій на етапі R&D |
|---|---|---|---|---|
| Процес створення завдання | | | | |
| Ручні операції | 1 | 8 | 8 | 14,3% |
| Автоматичні операції | 6 | 8 | 48 | 85,7% |
| Процес створення документа | | | | |
| Ручні операції | 3 | 5 | 15 | 50% |
| Автоматичні операції | 3 | 5 | 15 | 50% |

Аналіз таблиць 3.4 і 3.5 демонструє значний приріст автоматизації управлінських процесів. Зокрема, автоматизація створення завдань зросла на 85,7%, а створення документації – на 50%. Це підвищення сприяє не лише оптимізації рутинних операцій, а й покращенню ряду якісних характеристик управління. Наприклад, вдосконалюється візуалізація циклів прийняття рішень, збільшується обсяг збережених даних про програмні рішення, що створюються, а також забезпечується більш ефективна робота з інформацією.

Практична цінність та доцільність запропонованої моделі автоматизованої системи управління проектами (АСУ) підтверджуються відгуками експертів, зокрема технічного та виконавчого директорів компанії ТОВ «Антс».

Для комплексної оцінки ефективності розробленої моделі АСУ було проведено SWOT-аналіз. Цей підхід дозволив виявити ключові переваги системи, які можуть забезпечити конкурентні переваги, а також недоліки, що потребують вдосконалення в процесі подальшої розробки. SWOT-аналіз також допоміг визначити перспективні напрями розвитку моделі та потенційні способи її монетизації. Окрім цього, було ідентифіковано можливі ризики, які можуть виникнути під час розробки та впровадження моделюваного програмного продукту.

| Сильні сторони | Слабкі сторони |
|---|---|
| • Відстеження рішень<br>• Повнота документації<br>• Документація має уніфікований формат<br>• Збереження показників кожної випробуваної моделі<br>• Створення своєї бази знань<br>• Зменшення часу витрат проєктного менеджера шляхом автоматичного створення завдань<br>• Зменшення часу витрат команди розробки шляхом автоматичного створення шаблону документацій<br>• Зменшення часу витрат проєктною командою шляхом систематизації процесів розробки<br>• Готова документація для портфоліо команди компанії | • Витрати на розробку програмного продукту<br>• Залежність від компанії розробника<br>• Програмний продукт є не самостійним, а надбудовою на Jira<br>• Цільова аудиторія обмежена користувачами Jira |
| **Потенційні можливості** | **Потенційні загрози** |
| • Вихід на загальний ринок IT компаній<br>• Створення загальної бази знань<br>• Зменшення часових витрат на дослідження зовнішніх джерел<br>• Збільшення ймовірності успішного пошуку готових рішень<br>• Додатковий прибуток від продажу доступу до бази знань<br>• Створення мережі компанії на ринку штучного інтелекту | • Складно уніфікувати, оскільки процеси в компаніях відрізняються<br>• Потенційно великі витрати на вивчення процесу інших компаній<br>• Компанії можуть не дозволити доступ до їх напрацювань |

Рисунок 3.5 *SWOT*-аналіз *АСУ* проєктами ТОВ «Антс»

SWOT-аналіз виявив, що запропонована система має низку важливих переваг у порівнянні з попередньою моделлю управління. Серед основних сильних сторін моделі To Be можна виділити впорядкування документообігу, що зменшує хаос у зберіганні інформації. Система також сприяє покращенню візуального представлення етапів та задач, що робить роботу команди більш зручною та зрозумілою. Крім того, новий підхід підвищує ефективність комунікації між учасниками проекту, зменшуючи ризик втрати важливої інформації.

Однак, були виявлені й слабкі сторони. Головний недолік пов’язаний із залежністю системи від продуктів Atlassian, таких як Jira та Confluence. Це означає, що функціональність та стабільність системи залежатимуть від цінової

політики Atlassian і технічної підтримки їхніх продуктів. Ще одним викликом є обмеження цільової аудиторії: запропонований плагін може бути корисним лише для компаній, які вже використовують Atlassian. Це звужує потенційний ринок, адже далеко не всі Data Science-компанії працюють із Jira та Confluence.

Таким чином, хоча модель To Be значно покращує внутрішні процеси в компанії, її впровадження потребує врахування цих обмежень для зменшення ризиків і підвищення конкурентоспроможності продукту.

## 3.3. Висновки до розділу 3

Дослідження існуючих методів, інструментів і систем управління проєктами, детальний аналіз діяльності членів команди на етапі R&D, а також розробка алгоритму процесу управління на цьому етапі дозволили сформувати модель To Be автоматизованої системи управління проєктами. Вона враховує наявні в компанії ТОВ «Антс» недоліки, зокрема неорганізований документообіг і виконання рутинних завдань.

Модель To Be передбачає адаптацію поточної системи управління проєктами, яка базується на Jira і Confluence, шляхом інтеграції плагіна для автоматизації процесів етапу R&D. У встановлених межах це спричиняє певні зміни в структурі, функціональності та організації інформаційних потоків системи.

Модель To Be описується за допомогою чотирьох ключових діаграм:

– діаграма екосистеми вбудованого плагіна для управління R&D-процесами;

– діаграма бізнес-процесів моделі To Be;

– діаграма прецедентів, що ілюструє взаємодію учасників з системою;

– ER-діаграма, яка відображає сутності та зв’язки між ними в системі.

Цей комплекс діаграм забезпечує повне розуміння моделі To Be та створює основу для розробки технічного завдання (SRS) і подальшої реалізації плагіна.

# ВИСНОВКИ

У ході дослідження було проведено теоретичне обґрунтування моделі автоматизованої системи управління проєктами для компанії ТОВ «Антс», яка спеціалізується на створенні програмних продуктів із використанням штучного інтелекту та методів машинного навчання. Аналіз наукових статей, навчальних посібників з управління проєктами та матеріалів із практичним досвідом впровадження інформаційних систем управління дозволив підтвердити необхідність і доцільність використання сучасних інструментів для побудови ефективної системи управління ШІ-проєктами.

Для досягнення поставленої мети було проаналізовано сучасні методи, засоби та інформаційні системи управління проєктами, виявлено їхні переваги та недоліки. Спостереження за діяльністю проєктної команди компанії «Антс» на етапі розробки ШІ-продуктів показали наявність рутинних і дублюючих дій, недоліків у документообігу та управлінських процесах. Оскільки компанія вже використовує систему Jira, було вирішено розробити модель автоматизованої системи управління проєктами з інтеграцією плагіна для етапу R&D.

Розроблена модель To Be складається з чотирьох ключових діаграм:

- діаграма екосистеми плагіна для управління проєктами на етапі R&D;
- діаграма бізнес-процесів моделі To Be;
- діаграма прецедентів для моделі To Be;
- ER-діаграма, що відображає структуру сутностей і їх взаємозв’язки.

Якісна оцінка моделі проводилася методами експертної оцінки та порівняльного аналізу рівня автоматизації бізнес-процесів між моделями As Is та To Be. Результати показали, що модель To Be збільшує ступінь автоматизації створення завдань на 85,7%, а створення документації – на 50%.

SWOT-аналіз дозволив визначити сильні та слабкі сторони запропонованої моделі, а також потенційні ризики й перспективи її подальшого розвитку.

Новизна дослідження полягає у фокусі на етапі R&D, який є найбільш складним для управління через специфіку дослідницької діяльності. Запропонована модель включає алгоритм управління процесами на цьому етапі та подання життєвого циклу проєкту у вигляді дерева рішень. Це забезпечує візуалізацію точок переходу між циклами, що є її значною перевагою.

Рекомендації щодо практичного застосування моделі та подальших досліджень:

1. Реалізація моделі To Be у вигляді R&D-плагіна з подальшою інтеграцією в систему управління проєктами компанії.

2. Розробка механізмів управління часом для врахування часових обмежень та ефективного планування етапу R&D.

3. Розширення бібліотеки готових рішень із можливістю інтеграції зі сторонніми базами даних.

4. Створення рекомендаційної підсистеми, яка допомагатиме дослідникам у пошуку рішень на основі аналізу їхніх запитів.

Економічність і результативність моделі забезпечуються такими властивостями:

– автоматизація рутинної роботи;

– збереження всіх супровідних документів;

– організація інформаційних потоків через єдиний центр управління;

– скорочення часу на комунікацію;

– незалежність від територіального розташування учасників команди;

– підвищення швидкості й ефективності взаємодії між співробітниками.

Достовірність отриманих результатів підтверджується аргументованими висновками, оцінками експертів і відповідністю методів дослідження поставленій меті та завданням.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Адлер О., Лесько О., Лесько О. До питання актуальності та ефективності управління іт-проєктами в бізнес-процесах сучасного підприємства. *Економіка та суспільство*. 2024. № 59. URL: https://doi.org/10.32782/2524-0072/2024-59-1 (дата звернення: 15.09.2024).

2. Бабич К. В. Моделювання процесу створення ІТ-підприємства : thesis. 2016. URL: http://essuir.sumdu.edu.ua/handle/123456789/46739 (дата звернення: 15.09.2024).

3. Безугла Ю., Грицай О. Цифрові технології моделювання системи управління фінансовими ресурсами підприємства. *Adaptive management theory and practice economics*. 2024. Т. 18, № 36. URL: https://doi.org/10.33296/2707-0654-18(36)-07 (дата звернення: 15.09.2024).

4. Бочарова Н., Тихоненко О. Розробка документообігу щодо управління проєктами в роботі окремих відділів та служб підприємства. *InterConf*. 2023. № 31(147). С. 24–34. URL: https://doi.org/10.51582/interconf.19-20.03.2023.003 (дата звернення: 15.09.2024).

5. Везденецький С. В. Моделювання управління соціально-економічною ефективністю підприємства. *Економіка і організація управління*. 2022. № 1. С. 76–86. URL: https://doi.org/10.31558/2307-2318.2022.1.8 (дата звернення: 15.09.2024).

6. Венажиндене М., Чорна Л., Коваленко О. Моделювання бізнес-процесів для створення інформаційної системи управління якістю підприємства біоекономіки. *Herald of Khmelnytskyi National University. Economic sciences*. 2023. Т. 318, № 3. С. 244–248. URL: https://doi.org/10.31891/2307-5740-2023-318-3-38 (дата звернення: 15.09.2024).

7. Едуардович В. Теоретичні та практичні сегменти моделювання управління збалансованого розвитку підприємства. *Grail of science*. 2024. № 39.

С. 156–162. URL: https://doi.org/10.36074/grail-of-science.10.05.2024.019 (дата звернення: 15.09.2024).

8. Захарченко В., Акулюшина М., Лінгур Л. Моделювання процесів управління витратами підприємства в умовах невизначеності. *Економіка та суспільство*. 2024. № 64. URL: https://doi.org/10.32782/2524-0072/2024-64-2 (дата звернення: 15.09.2024).

9. Інструментарій управління проєктами фінансового забезпечення підприємства / І. Дячок та ін. *Development service industry management*. 2024. № 2. С. 149–154. URL: https://doi.org/10.31891/dsim-2024-6(22) (дата звернення: 15.09.2024).

10. Книрік Н. Р. Управління портфелем проектів аутсорсингової ІТ-компанії на основі імітаційного моделювання : автореф. дис. … канд. техн. наук. Миколаїв, 2016. 20 с.

11. Колодінська Я. Мережеві методи моделювання процесів управління іт-проєктами в умовах цифрової економіки. *Herald of Khmelnytskyi National University. Economic sciences*. 2024. Т. 326, № 1. С. 289–296. URL: https://doi.org/10.31891/2307-5740-2024-326-45 (дата звернення: 15.09.2024).

12. Командровська В. Моделювання механізму управління інноваційним забезпеченням сталого розвитку підприємства. *Економіка та суспільство*. 2024. № 65. URL: https://doi.org/10.32782/2524-0072/2024-65-141 (дата звернення: 15.09.2024).

13. Кравчук О. Метод використання метрик продуктивності для оптимізації процесу управління іт-проєктами. *Measuring and computing devices in technological processes*. 2023. № 2. С. 28–33. URL: https://doi.org/10.31891/2219-9365-2023-74-4 (дата звернення: 15.09.2024).

14. Кузьмін О., Свірський Ю. Еклектичне імітаційне моделювання розвитку управління бізнес-процесами підприємства. *Věda a perspektivy*. 2024. № 11(42). URL: https://doi.org/10.52058/2695-1592-2024-11(42)-298-311 (дата звернення: 15.09.2024).

15. Лелюк С., Манасян В. Візуалізація результатів моделювання процесу управління прибутковістю підприємства. *Молодий вчений*. 2021. № 12 (100). С. 272–277. URL: https://doi.org/10.32839/2304-5809/2021-12-100-55 (дата звернення: 15.09.2024).

16. Литовченко О., Кузенко Т. Структурно-функціональне моделювання процесу управління інвестиційною привабливістю підприємства. *Економіка та суспільство*. 2021. № 34. URL: https://doi.org/10.32782/2524-0072/2021-34-101 (дата звернення: 15.09.2024).

17. Мискін Ю. І., Романюк А. П., Денещук К. С. Моделювання обліково-аналітичного забезпечення управління цінністю підприємства. *Український економічний часопис*. 2023. № 1. С. 34–37. URL: https://doi.org/10.32782/2786-8273/2023-1-6 (дата звернення: 15.09.2024).

18. Мінухін С. В. Моделювання управління бізнес-процесами підприємства: стратегічний аспект. *Актуальні проблеми економіки*. 2007. № 10. С. 156–164.

19. Міщенко В. Ю., Мищенко В. Ю., Mishchenko V. Y. Моделювання управління ризиками туристичного підприємства : автореф. Автореферат магістерської роботи. 2016. URL: https://dspace.znu.edu.ua/jspui/handle/12345/112 (дата звернення: 15.09.2024).

20. Овецька О. В., Кукудяк Н. В. Управління проєктами: стан та перспективи розвитку підприємства. *Економіка і організація управління*. 2022. № 2. С. 139–147. URL: https://doi.org/10.31558/2307-2318.2022.2.14 (дата звернення: 15.09.2024).

21. Петренко Л. М. Моделювання процесів управління фінансовою безпекою підприємства : автореф. дис. … канд. екон. наук. Київ, 2010. 19 с.

22. Сагер Л. Ю. Моделювання та оптимізація процесу управління внутрішніми комунікаціями підприємства: методичний підхід1. *Науковий вісник ужгородського університету. серія «економіка»*. 2017. № 2(50). С. 218–222. URL: https://doi.org/10.24144/2409-6857.2017.2(50).218-222 (дата звернення: 15.09.2024).

23. Свінцицька О. М., Пулеко І. В. Інтеграція Jira, Bitbucket та Sourcetree в системі управління ІТ-проєктами. *Технічна інженерія*. 2023. № 2(92). С. 102–108. URL: https://doi.org/10.26642/ten-2023-2(92)-102-108 (дата звернення: 15.09.2024).

24. Симонов В. Вплив agile-методологій на ефективність управління іт-проєктами: аналіз практик та результатів. *Grail of science*. 2023. № 34. С. 185–187. URL: https://doi.org/10.36074/grail-of-science.08.12.2023.38 (дата звернення: 15.09.2024).

25. Соляник Л. Г., Цуркан І. М., Галіба С. В. Економіко-математичне моделювання оптимізації управління оборотними активами промислового підприємства. *Efektyvna ekonomika*. 2022. № 11. URL: https://doi.org/10.32702/2307-2105.2022.11.46 (дата звернення: 15.09.2024).

26. Управління бізнес-проєктами щодо транспортного забезпечення зовнішньоекономічної діяльності підприємства / П. Пашко та ін. *Development service industry management*. 2024. № 2. С. 265–269. URL: https://doi.org/10.31891/dsim-2024-6(42) (дата звернення: 15.09.2024).

27. Фендьо О. Управління іт-проєктами з використанням гнучких методологій agile, scrum, kanban. *Distance education in ukraine: innovative, normative-legal, pedagogical aspects*. 2023. № 2. С. 419–425. URL: https://doi.org/10.18372/2786-5495.1.17349 (дата звернення: 15.09.2024).

28. Фонар Л. С., Лотіс О. С. Аналіз технологій управління іт-проєктами в умовах негативного впливу вигорання. *Інфокомунікаційні та комп'ютерні технології*. 2024. Т. 1, № 07. С. 21–24. URL: https://doi.org/10.36994/2788-5518-2024-01-07-02 (дата звернення: 15.09.2024).

29. Храпкін О., Кіндрат О., Чопей Р. Управління проєктами в іт-галузі: методики, інструменти та керування ризиками. *Економіка та суспільство*. 2023. № 55. URL: https://doi.org/10.32782/2524-0072/2023-55-110 (дата звернення: 15.09.2024).

30. Хринюк О., Очеретяна О. Особливості моделювання бізнес-процесу «управління інвестиційною привабливістю підприємства». *Економіка та*

*суспільство*. 2021. № 27. URL: https://doi.org/10.32782/2524-0072/2021-27-20 (дата звернення: 15.09.2024).

31. Черноусова Ж. Т., Фартушний І. Д., Лісовська О. М. Економіко-математичне моделювання управління грошовими активами підприємства в сфері фінансового посередництва. *Підприємництво та інновації*. 2023. № 28. С. 79–86. URL: https://doi.org/10.32782/2415-3583/28.13 (дата звернення: 15.09.2024).

32. Шитікова Л. В., Головатенко С. С., Головатенко О. С. Моделювання перспективного розвитку системи управління на шляху удосконалення інноваційного потенціалу підприємства. *Управління змінами та інновації*. 2023. № 7. С. 92–95. URL: https://doi.org/10.32782/cmi/2023-7-13 (дата звернення: 15.09.2024).

33. Юрчук Н. П., Кіпоренко С. С. Методології управління іт-проєктами: критерії вибору для ефективного управління. *Efektyvna ekonomika*. 2024. № 11. URL: https://doi.org/10.32702/2307-2105.2024.11.76 (дата звернення: 15.09.2024).

34. Chaikina A., Kravchenko O., Pokhidnia B. Innovative marketing technologies of it-project management. *Market infrastructure*. 2022. No. 69. URL: https://doi.org/10.32782/infrastruct69-21 (date of access: 15.09.2024).

35. Dudnyk O. O., Sokolovska Z. M. Modern technologies for management of outsourcing IT projects. *Business inform*. 2021. Vol. 2, no. 517. P. 62–72. URL: https://doi.org/10.32983/2222-4459-2021-2-62-72 (date of access: 15.09.2024).

36. Hud O. O., Kunanets N. E. Доцільність застосування систем прогнозування як складової частини процесу управління ІТ-проєктами. *Scientific Bulletin of UNFU*. 2023. Т. 33, № 3. С. 71–76. URL: https://doi.org/10.36930/40330310 (дата звернення: 15.09.2024).

37. Kiyko S., Druzhinin E., Prokhorov O. Імітаційне моделювання процесів управління енергоресурсами металургійного підприємства. *Системи управління, навігації та зв'язку. Збірник наукових праць*. 2019. Т. 6, № 58. С. 20–27. URL: https://doi.org/10.26906/sunz.2019.6.020 (дата звернення: 15.09.2024).

38. Modeling optimization tasks in it project management / I. Y. Ivchenko et al. *Electrical and computer systems*. 2024. No. 40(116). P. 25–36. URL: https://doi.org/10.15276/eltecs.40.116.2024.3 (date of access: 15.09.2024).

39. Stankevych I. V., Borysevych E. H., Klikich S. D. Developing the IT project management models of bank on an innovation basis. *Business inform*. 2022. Vol. 2, no. 529. P. 117–125. URL: https://doi.org/10.32983/2222-4459-2022-2-117-125 (date of access: 15.09.2024).

40. Tyurina N., Nazarchuk T., Shkabara N. Formation of information and analytical support of business project management of the enterprise. *Innovation and sustainability*. 2022. No. 2. P. 68–77. URL: https://doi.org/10.31649/ins.2022.2.68.77 (date of access: 15.09.2024).