

DOI: 10.32347/2412-9933.2025.64.84-93

УДК 004.42:004.9:81'33

Козак Іван Володимирович

Аспірант кафедри інформаційних систем та мереж,

<https://orcid.org/0009-0007-4953-2816>

Національний університет «Львівська політехніка», Львів

Кунанець Наталія Едуардівна

Докторка наук з соціальних комунікацій, професорка кафедри інформаційних систем та мереж,

<https://orcid.org/0000-0003-3007-2462>

Національний університет «Львівська політехніка», Львів

УПРАВЛІННЯ НАУКОВИМ ІТ-ПРОЄКТОМ «РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ АНАЛІЗУ УКРАЇНОМОВНИХ КОРПУСІВ»

Анотація. Розглянуто комплексне питання управління науковим ІТ-проєктом, орієнтованим на створення інформаційної системи для роботи з україномовними текстовими корпусами різних жанрів і структурних типів. Здійснено системний аналіз актуальних підходів до організації дослідницьких ІТ-проєктів в умовах обмежених ресурсів, змінного середовища, невизначеності технічних пріоритетів, різного рівня залученості стейкхолдерів та високої варіативності функціональних вимог, що формуються на перетині лінгвістичних і технічних задач. Запропоновано модель управління, що поєднує методології SCRUM, Lean і exploratory programming, яка забезпечує баланс між структурованістю, економічністю, гнучкістю, можливістю адаптивного планування та швидким реагуванням на зміни вимог у процесі еволюції наукового задуму. Визначено цільову аудиторію майбутньої системи, що охоплює прикладних лінгвістів, викладачів, студентів, перекладачів, дослідників цифрової гуманітаристики, а також фахівців, які працюють із корпусними методами в освіті, прикладних студіях й галузевих аналітичних задачах. Подано узагальнений соціально-професійний портрет потенційного користувача та описано типові сценарії застосування системи в навчальних, наукових і дослідницьких процесах, зокрема аналіз текстів, формування підкорпусів, розмітку, статистичний опис, підготовку даних для експериментів. Проведено ідентифікацію ключових стейкхолдерів і виконано їхню класифікацію за рівнем впливу, зацікавленості, очікуваної користі та ризиків взаємодії. Обґрунтовано використання фазової моделі розробки програмного забезпечення, що включає концептуалізацію, проєктування, імплементацію, тестування, впровадження та подальший супровід, що дає змогу формалізувати процес і водночас зберегти гнучкість. Результати дослідження можуть слугувати методологічною основою для планування, організації та реалізації наукових ІТ-проєктів у сфері цифрової лінгвістики. Запропонований підхід є релевантним для проєктів, що поєднують наукове дослідження з програмною розробкою, мають високий рівень невизначеності вимог і потребують адаптивних управлінських рішень.

Ключові слова: управління науковим проєктом; інформаційна система; українські текстові корпуси; SCRUM; Lean; exploratory programming; прикладна лінгвістика

Вступ та постановка проблеми

У сучасних лінгвістичних дослідженнях значну роль відіграють цифрові корпуси текстів, які слугують джерелом для аналізу мовних закономірностей, тестування гіпотез та побудови мовних моделей. Водночас із розширенням обсягів текстових даних та ускладненням їхньої структури постає потреба у створенні ефективних інструментів для збирання, опрацювання, фільтрації та аналізу корпусних ресурсів. Особливої уваги вимагає розроблення спеціалізованих інформаційних систем, що відповідають потребам україномовної

лінгвістичної спільноти, враховуючи специфіку мови, доступні ресурси та академічне середовище.

Розробка таких систем є складним науковим і технічним проєктом, що вимагає не лише глибокого розуміння лінгвістичних задач, а й чітко сформульованого підходу до управління проєктом. В умовах обмежених ресурсів, необхідності швидкого реагування на зворотний зв'язок користувачів та високої варіативності функціональних вимог постає потреба у застосуванні комбінованих гнучких підходів до проєктного менеджменту, які б забезпечували ефективну розробку та еволюцію продукту.

Таким чином, виникає необхідність не лише у створенні інформаційної системи для управління корпусами текстів українською мовою, але й у розробленні ефективної моделі управління науковим проєктом, яка дозволила б врахувати потреби користувачів, обмежені ресурси, ризики та специфіку академічного середовища. Це зумовлює актуальність дослідження управлінського аспекту такого проєкту та обґрунтування комбінованої стратегії його реалізації.

Аналіз останніх досліджень та публікацій

І. Л. Сазонець, Н. В. Ковшун [1] розглядають управління науковими проєктами як багаторівневу систему, що передбачає стратегічне планування, організацію, контроль і оцінку результатів у межах науково-освітніх інституцій. У роботі І. Якименко, Є. Штефана, В. Лук'янихіна [2] можемо відзначити акцент на практичних аспектах управління проєктами, зокрема плануванні, моніторингу та оцінюванні ефективності, адаптованих до потреб PhD-дослідників. О. Б. Данченко, О. Ю. Савіна, Д. І. Бедрій, О. В. Гайдаєнко, В. М. Меленчук [3] пропонують підходи до управління портфелями наукомістких проєктів, які включають інструменти ціннісно-орієнтованого підходу, ризик-менеджменту та циркулярної економіки.

В. В. Храпкіна, О. Д. Матвієнко [4] фокусуються на специфіці управління IT-проєктами, підкреслюючи важливість адаптації класичних методологій до гнучких умов галузі. Дослідження І. Мартінес, Е. Вайлес та І. Олайзола [5] аналізують чинники успіху в проєктах із аналізу даних, серед яких виокремлено роль чіткої комунікації, підтримки керівництва та гнучкості у виборі технологічного стеку. А. Л. Бакстер, Дж. Чой, І. Гортон, та С. Шен, [6] демонструють, як підходи SCRUM можуть бути адаптовані для розподілених команд, що працюють над науковим програмним забезпеченням, наголошуючи на необхідності міждисциплінарної співпраці.

У дослідженні Р. Мілевич та Е. М. Рейборн [7] проєкт розглядається як простір координації знань між учасниками з різними профілями, а ефективне управління знаннями – як критичний чинник успішності наукового програмного забезпечення. Е. К. Джонсон, Дж. Ліман, М. К. Рамамурті та Дж. Янг [8] презентують підхід SciOps – модель організаційної підтримки команд у великих дата-орієнтованих дослідженнях, яка балансує між продуктивністю, масштабованістю та відтворюваністю результатів.

Ці публікації демонструють широкий спектр підходів до управління науковими проєктами – від класичних моделей у сфері освіти до адаптивних

фреймворків у науковому програмуванні та проєктах з аналізу даних. Вони підкреслюють важливість чітко структурованих процесів, міждисциплінарної взаємодії, управління ризиками та підтримки на рівні організаційної культури.

Формулювання цілі статті та постановка задачі

Метою роботи є дослідження та обґрунтування підходів до управління науковим проєктом зі створення інформаційної системи для управління корпусами україномовних текстів із врахуванням специфіки академічних IT-проєктів.

Для досягнення поставленої мети були визначені такі завдання:

1. Виділити цільову аудиторію інформаційної системи та надати їхній психо-соціальний портрет.
2. Визначити ключових стейкхолдерів проєкту та їхні ролі в межах життєвого циклу розробки продукту.
3. Обґрунтувати доцільність застосування комбінованого підходу до управління проєктом, з урахуванням ресурсних та організаційних обмежень.

Об'єктом дослідження є науковий проєкт зі створення інформаційної системи управління корпусами текстів.

Предметом дослідження є методи та моделі управління науковими IT-проєктами в умовах обмежених ресурсів і динамічного середовища.

Виклад основного матеріалу

Концептуалізація проєкту та розробка прототипу

Основною ідеєю досліджуваного проєкту є створення інформаційної системи, спеціально розробленої для управління корпусами текстів, які є фундаментальними для емпіричного дослідження мови та виявлення закономірностей її використання. Текстовий корпус, який визначається як велика й часто неструктурована сукупність текстів, слугує основним джерелом даних для статистичного аналізу, перевірки гіпотез і валідації мовних закономірностей у межах певної мовної території.

Проєкт подається як система ітерацій:

$$P = \langle S, B, T, F, \Delta \rangle,$$

де $S = \{s_1, s_2, \dots, s_n\}$ – множина спринтів; B – беклог вимог; $T(S_i)$ – набір задач, реалізованих у спринті S_i ; F – функція оцінки виконання спринту; Δ – оператор зміни вимог на основі фідбеку користувачів.

У кожному спринті виконується:

$$T(S_i) = \operatorname{argmax}_{t \in B} (V(t) - C(t)),$$

де $V(t)$ – цінність задачі (value), оцінена разом зі стейкхолдерами, $C(t)$ – її очікувана вартість / трудомісткість.

Це подає SCRUM-пріоритизацію як оптимізацію корисності.

Дизайн і функціональні можливості програмних засобів, які використовують корпусні лінгвісти, дедалі більше впливають на методи, що застосовуються в дослідженнях, особливо у зв'язку зі зростанням обсягу корпусів і складності мовного аналізу. Саме функціональність інструментів визначає, які дослідницькі питання можуть бути поставлені й опрацьовані, що підкреслює важливість розробки потужних та зручних систем для управління й аналізу корпусів текстів. Добре спроектована інформаційна система для цих цілей здатна істотно підвищити ефективність і глибину лінгвістичних досліджень, роблячи цінний внесок у наукову спільноту.

Аналіз ключових стейкхолдерів проекту

Успіх будь-якого проекту з розроблення програмного забезпечення, особливо спрямованого на наукову спільноту, залежить від ідентифікації та ефективної взаємодії зі стейкхолдерами. Стейкхолдерами вважаються індивіди або групи, як внутрішні, так і зовнішні щодо організації, які зацікавлені в проекті та можуть впливати на його результати. Розуміння їхніх потреб, очікувань і потенційного впливу є ключовим для узгодження цілей проекту з ширшими інтересами наукової та дослідницької спільноти.

Особливо важливо залучати стейкхолдерів на ранніх етапах життєвого циклу проекту – під час збору вимог. Це сприяє врахуванню їхніх потреб у розробці функціоналу системи, підвищує рівень залучення та відчуття причетності до проекту, що зрештою підвищує ймовірність задоволення від кінцевого продукту.

Згідно з підходом Кларксона (Clarkson, 1995), стейкхолдерів доцільно поділяти на первинних та вторинних. Цей поділ деталізовано у дослідженні Mainardes, Alves & Raposo (2012), де запропоновано модель класифікації стейкхолдерів і їхніх взаємозв'язків. Для ефективного управління цими різноманітними стейкхолдерами необхідний ґрунтовний аналіз їхніх інтересів. Це передбачає оцінку рівня впливу кожного стейкхолдера, впливу проекту на них, а також їхніх пріоритетів і очікувань. Знання цих факторів дозволяє визначити належний рівень залучення кожного стейкхолдера та адаптувати стратегії комунікації й взаємодії відповідно.

Для формального опису впливу стейкхолдерів на науковий ІТ-проект використовується матрична

модель, що дозволяє кількісно оцінити внесок і значущість кожної зацікавленої сторони. Загальна структура моделі подається у вигляді матриці:

$$S = [S_{ij}],$$

де кожен елемент матриці S_{ij} описує величину впливу i -го стейкхолдера за певною категорією j .

Індекс i відповідає конкретному стейкхолдеру (наприклад, кінцевому користувачу, розробнику, науковому керівнику, адміністрації університету або грантодавцю), тоді як індекс j позначає категорію впливу, що є важливою для управління проектом. До таких категорій можуть належати:

- влада (power) – здатність стейкхолдера прямо чи опосередковано впливати на рішення, ресурси або пріоритети проекту;
- інтерес (interest) – рівень зацікавленості стейкхолдера в результатах розроблення та ступінь його залучення;
- важливість (importance) – критичність ролі стейкхолдера для успішної реалізації системи;
- ризики (risks) – потенційний негативний вплив, який може виникнути у разі зміни позиції, очікувань або поведінки стейкхолдера.

Таким чином, матриця забезпечує структуроване подання стейкхолдерів та їхніх характеристик, що дозволяє виконувати подальший аналіз взаємодії. На підставі цієї матриці можуть обчислюватися агреговані показники, зокрема загальна сила впливу кожного стейкхолдера, визначатися пріоритети комунікації та формуватися стратегічні рішення щодо управління очікуваннями та ризиками в межах життєвого циклу проекту.

Для кількісного опису впливу зацікавлених сторін на хід і результати наукового ІТ-проекту застосовується інтегральна модель, що дозволяє оцінювати значущість кожного стейкхолдера на основі сукупності релевантних критеріїв. Загальна сила впливу I_i визначається за формулою:

$$I_i = \sum_j \alpha_j S_{ij},$$

де S_{ij} є оцінкою впливу i -го стейкхолдера за j -ю категорією (зокрема, владою, інтересом, важливістю або пов'язаними ризиками), а α_j – ваговим коефіцієнтом, який відображає відносну значущість цієї категорії у загальній структурі оцінювання. Значення S_{ij} зазвичай фіксується в експертній бальній шкалі, що забезпечує можливість порівняння різних стейкхолдерів між собою. Водночас ваги α_j задаються відповідно до стратегічних пріоритетів проекту та узгоджуються зі стейкхолдерами або керівною групою, що забезпечує відображення реальної важливості кожного чинника.

Сума $\sum_j \alpha_j s_{ij}$, дає змогу агрегувати

інформацію про стейкхолдера у єдиний інтегральний показник, який характеризує загальний рівень його впливу на проєкт. Чим більше значення I_i , тим вагомішою є роль стейкхолдера та тим більшої уваги потребує стратегія взаємодії з ним.

Для підтримки прийняття рішень щодо пріоритезації комунікацій та управління очікуваннями додатково використовується класифікація стейкхолдерів за квадрантами. Вона ґрунтується на порівнянні інтегрального показника впливу та індексу залученості L_i із встановленими порогами τ_I та τ_L . Формально ключовий стейкхолдер визначається як такий, що задовольняє нерівності:

$$I_i \geq \tau_I, L_i \geq \tau_L.$$

У цьому контексті індекс L_i може відображати рівень інтересу або активності стейкхолдера у проєкті, тоді як порогові значення τ_I та τ_L визначають мінімально необхідні рівні впливу та залученості для включення стейкхолдера до критично важливої групи. Такий підхід дозволяє формувати збалансовані стратегії взаємодії, оптимізувати ресурси комунікації та ефективно управляти ризиками, пов'язаними із поведінкою стейкхолдерів протягом життєвого циклу проєкту.

Також можна подати класифікацію за квадрантами через нерівності: $I_i \geq \tau_I, L_i \geq \tau_L \Rightarrow$ – ключовий стейкхолдер.

Вплив зацікавлених сторін (стейкхолдерів) на хід і результати наукового ІТ-проєкту доцільно подати за допомогою когнітивної карти, яка дозволяє формально описати причинно-наслідкові зв'язки між чинниками. Когнітивні карти (cognitive maps, fuzzy cognitive maps, FCM) є одним із найефективніших інструментів моделювання взаємозв'язків між чинниками у складних системах — зокрема, у проєктному менеджменті, управлінні ризиками та аналізі стейкхолдерів. Когнітивна карта є орієнтованим графом, у якому:

- вузли відображають стейкхолдерів або ключові характеристики стейкхолдерів (влада, інтерес, ризики, ресурси, вплив на рішення тощо);
- дуги позначають напрям і силу впливу одних стейкхолдерів або факторів на інші;
- ваги зв'язків показують ступінь і характер впливу (позитивний або негативний).

У загальному вигляді когнітивна карта визначається як:

$$M = \langle V, E, W \rangle,$$

де $V = \{v_1, v_2, \dots, v_n\}$ – множина вузлів (стейкхолдери або їх характеристики); E – множина ребер;

$W = [w_{ij}]$ – матриця ваг зв'язків між вузлами.

Кожне ребро $v_i \rightarrow v_j$ із вагою w_{ij} означає, що зміна фактора v_i причиняє зміну фактора v_j , причому:

$$w_{ij} \in [-1, 1],$$

де $w_{ij} > 0$ – позитивний вплив, $w_{ij} < 0$ – негативний вплив, $w_{ij} = 0$ – відсутність впливу.

Наприклад концептуальна когнітивна карта для наукового ІТ-проєкту містить типовий набір факторів:

Вузли (фактори): v_1 – влада стейкхолдера; v_2 – інтерес стейкхолдера; v_3 – якість зворотного зв'язку; v_4 – своєчасність прийняття рішень; v_5 – ресурси проєкту; v_6 – ризики затримок; v_7 – успішність реалізації проєкту.

Між ними такі зв'язки:

- $v_1 \rightarrow v_4$ (вплив влади на прийняття рішень), $w_{14} = 0.8$;
- $v_2 \rightarrow v_3$ (інтерес стейкхолдера підсилює якість фідбеку), $w_{23} = 0.6$;
- $v_3 \rightarrow v_7$ (якісний фідбек покращує успішність), $w_{37} = 0.7$;
- $v_6 \rightarrow v_7$ (ризики зменшують успішність), $w_{67} = -0.8$;
- $v_5 \rightarrow v_6$ (більше ресурсів $\rightarrow$ менше ризиків), $w_{56} = -0.5$;
- $v_4 \rightarrow v_6$ (своєчасно прийняті рішення $\rightarrow$ менше ризиків), $w_{46} = -0.4$.

Це дозволяє оцінювати *непрямі ефекти*, зокрема, зростання інтересу стейкхолдерів $\rightarrow$ кращий зворотній зв'язок $\rightarrow$ нижча ймовірність дефектів $\rightarrow$ вища успішність проєкту.

Проведемо динамічне моделювання

У випадку використання нечітких когнітивних карт (FCM) можна моделювати розвиток ситуації:

$$v_i^{(t+1)} = f \left(v_i^{(t)} + \sum_j v_j^{(t)} w_{ij} \right),$$

де f – порогова або сигмоїдна функція.

Це дає можливість прогнозувати вплив окремих стейкхолдерів, оцінювати сценарії («що буде, якщо влада стейкхолдерів зросте/зменшиться?»), визначати критичні фактори успіху.

Використання когнітивної карти (рис. 1) дозволило створити формальну, візуальну й інтуїтивну модель взаємодії стейкхолдерів для аналітичного моделювання розвитку проекту.

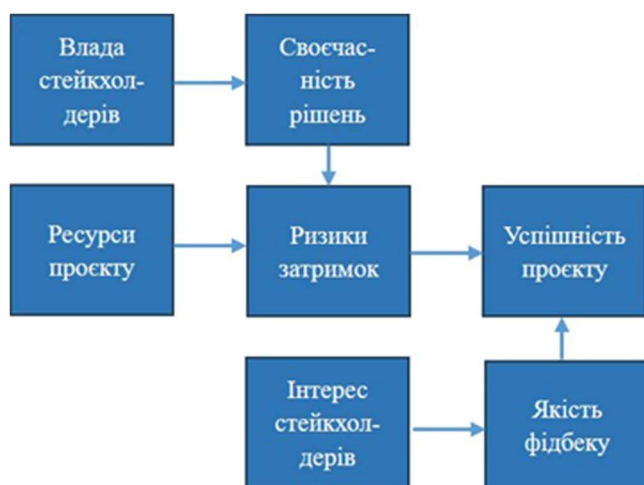


Рисунок 1 – Когнітивна карта впливу стейкхолдерів

У контексті наукового проекту створення інформаційної системи управління корпусами текстів слід виділити наступні групи стейкхолдерів:

- *Первинні стейкхолдери:*

1. Користувачі системи:

- Дослідники та науковці – основні користувачі, які працюють із текстовими корпусами.
- Студенти та викладачі – потенційні користувачі в освітньому контексті, зокрема при використанні DDL (Data Driven Learning).

2. Розробники та технічна команда, а саме програмісти та адміністратори: відповідальні за створення, підтримку та розвиток системи. У випадку наукового проекту – це аспірант або група аспірантів.

3. Керівництво проекту, а саме – керівники наукових груп чи наукові керівники, котрі здійснюють контроль за розвитком проекту та відповідність його цілям.

- *Вторинні стейкхолдери:*

1. Адміністрація наукової установи – керівництво університету або науково-дослідного інституту зацікавлене в успішності проекту з позицій наукових результатів, фінансування або іміджу установи.

2. Фінансові організації або грантодавці забезпечують фінансування проекту, але не взаємодіють із системою безпосередньо.

3. Зовнішні користувачі або партнери – до прикладу, інші наукові інститути, компанії, дослідницькі групи: потенційні користувачі результатів або даних, створених у межах проекту (наприклад, корпусів, статистик тощо).

4. Наукова спільнота та суспільство загалом, адже якщо система спрямована на створення відкритих даних або підвищення наукової обізнаності, то суспільство може бути зацікавлене в результатах, навіть не використовуючи систему безпосередньо.

Визначення цільової аудиторії в межах наукової та дослідницької спільнот

Визначення цільової аудиторії для системи управління корпусами текстів, призначеної для наукових досліджень, є ключовим етапом, що дозволяє адаптувати її функціональність, інтерфейс та супровідні матеріали відповідно до специфічних потреб користувачів. Основною цільовою аудиторією такої системи серед науковців і дослідників зазвичай є лінгвісти, які займаються різними формами мовного аналізу; лексикографи, що працюють над створенням та оновленням словників; перекладачі, які використовують корпуси для перекладознавчих досліджень і контролю якості; студенти та викладачі мовознавчих і суміжних спеціальностей; соціальні науковці, що аналізують текстові дані у своїх дослідженнях; а також гуманітарії, які працюють із великими текстовими збірками.

Серед типових профілів кінцевих користувачів варто виділити:

1. Прикладного лінгвіста (студентка, аспірантка, викладачка-дослідниця віком 20–65 років), що працює з корпусами текстів під час наукової чи кваліфікаційної роботи в межах університетів або академічних установ України. Вона використовує систему для розмітки текстів, статистичного аналізу, перевірки гіпотез або дослідження вживання мовних одиниць у контексті з метою написання дисертації, публікацій або реалізації дослідницьких проектів.

2. Викладача української мови (учительку або викладачку віком 30–60 років), яка аналізує мовні явища на основі корпусних даних під час підготовки до занять або розробки методичних матеріалів. Вона працює в межах шкіл чи університетів України й прагне підвищити ефективність навчального процесу, зокрема через використання сучасних цифрових інструментів.

Ці різноманітні групи використовують системи управління корпусами текстів у межах широкого спектра дослідницьких і освітніх завдань: від статистичного аналізу мовних даних і перевірки лінгвістичних гіпотез на емпіричному матеріалі до пошуку вживань окремих слів або фраз у контексті та верифікації мовних правил у межах певної мовної території. Аналіз кожної з груп за методикою п'яти W (Who? What? When? Where? Why?) узагальнено у таблиці.

Таблиця– Узагальнення цільової аудиторії за схемою 5W

Who?	Прикладний лінгвіст	Викладач української мови
Короткий соціально-демографічний опис	студент, PhD, викладач-науковець здебільшого жінки, віком 20-65 років	вчитель чи викладач в університеті переважно жінки, віком 30-60 років
What?	Робота з корпусами	Робота з аналітичними даними
When?	Науково-дослідна робота	Розробка методичних матеріалів
Where?	Науково-освітній заклад України	Науково-освітній заклад України
Why?	Проведення науково-дослідної роботи	Покращення навчального процесу

Стратегія розробки інформаційної системи

Обґрунтування та план реалізації комбінованого підходу до управління проєктом

З метою ефективного управління складністю та невизначеністю, що супроводжують розробку програмного забезпечення в межах науково-дослідного проєкту, доцільним є застосування комбінованого підходу, який інтегрує сильні сторони методологій SCRUM, Lean і exploratory programming (рис. 2). Методологія SCRUM забезпечує структуровану основу для ітеративного процесу розробки, що передбачає короткі цикли (спринти) тривалістю 2–4 тижні з метою поетапного створення функціональних інкрементів системи. Ключовими елементами цього підходу є регулярні спринт-рев'ю для оцінки поточних результатів, а також ретроспективи з науковим керівником проєкту, що дає змогу своєчасно ідентифікувати недоліки процесу, узагальнювати набутий досвід і адаптувати подальшу діяльність відповідно до нових викликів або уточнених вимог.

Інтеграція принципів Lean спрямована на підвищення ефективності розроблення шляхом максимального усунення втрат на всіх етапах життєвого циклу програмного забезпечення. Нехай повний час виконання спринту

$$\tau = \tau_v + \tau_w,$$

де τ_v – час, що створює цінність (value adding time);

τ_w – втрати (wastes).

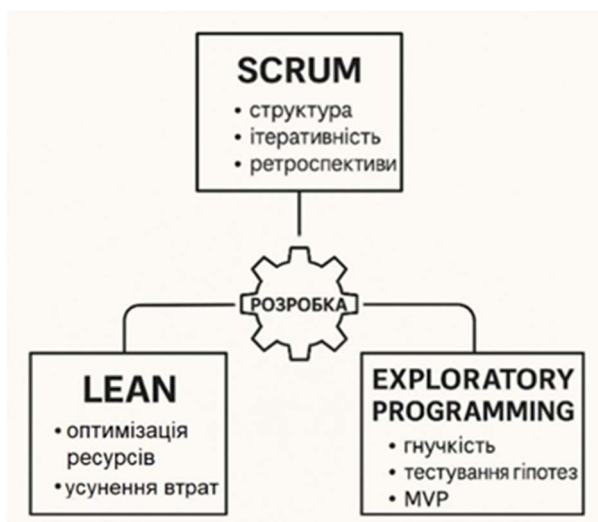


Рисунок 2 – Комбінований підхід до управління проєктом

Lean передбачає мінімізацію: $\min \tau_w$.

Показник ефективності Lean-ітерації:

$$\eta = \frac{\tau_v}{\tau_v + \tau_w}.$$

Цей KPI можна реально вимірювати на кожному спринті.

Особливу увагу приділено оптимізації витрат, що є критично важливим в умовах обмеженого фінансування, коли значна частина ресурсів, зокрема витрати на оренду технічних приміщень або хмарних обчислювальних потужностей, покривається безпосередньо виконавцем дослідження. Методологія Lean також передбачає підтримку гнучкості проєкту через відтермінування остаточних рішень, швидке отримання зворотного зв'язку, розширення повноважень розробника та вбудовування якості в систему на ранніх етапах її створення.

Користувачський зворотній зв'язок моделюється як множина сигналів:

$$F = \{f_1, f_2, \dots, f_m\}.$$

Оператор оновлення вимог:

$$B_{i+1} = \Delta(B_i, F_i),$$

де Δ може бути формалізований як

$$\Delta(B_i, F_i) = B_i \cup \{t : \sigma(t, F_i) \geq \theta \wedge t : \rho(t, F_i) \geq \gamma\},$$

де $\sigma(t, F_i)$ – функція важливості нової вимоги згідно фідбеку, $\rho(t, F_i)$ – функція виявленої зайвості / надмірності, θ , γ – пороги прийняття / видалення.

Це – формальна модель еволюції прототипу.

З урахуванням дослідницького характеру

проекту та високої варіативності вимог до кінцевого продукту, раціональним є залучення підходу *exploratory programming*. Цей підхід передбачає циклічне чергування фаз аналізу, програмування та рефлексії над отриманими результатами, що дозволяє на ранніх етапах розроблення експериментально перевіряти різні гіпотези щодо функціонування системи. Таким чином створюються MVP (мінімально життєздатні продукти), які дають змогу оперативно отримувати відгуки від потенційних користувачів та приймати обґрунтовані рішення щодо подальшого розвитку функціональності.

Фазовий підхід: етапи розробки продукту

Розроблення системи управління корпусами текстів реалізується за фазовим підходом, що передбачає проходження типових етапів життєвого циклу програмного забезпечення з метою забезпечення послідовного переходу від концептуальної моделі до повнофункціонального, підтримуваного продукту.

Початкова фаза концептуалізації зосереджена на формулюванні цілей, завдань і меж проекту, а також на досягненні узгодженого бачення його результатів серед усіх стейкхолдерів.

У наступній фазі планування передбачено побудову UML-діаграм для моделювання архітектурних рішень і ключових функцій системи, розроблення UX-сценаріїв, ідентифікацію зацікавлених сторін у межах наукової спільноти, а також проведення SWOT-аналізу для виявлення стратегічних переваг і ризиків проекту.

Фаза проектування включає деталізацію функціональних характеристик і специфікацій системи на основі зібраних вимог і результатів початкового тестування. До цього належить опис інтерфейсів користувача, логіки обробки даних та структур бази даних. У межах фази імплементації здійснюється створення програмного прототипу та основної версії інформаційної системи. На цьому етапі використовуються методи *exploratory programming* для перевірки технічних гіпотез і SCRUM як механізм ітеративної реалізації функціоналу, що доповнюється Lean-підходами для забезпечення економічності процесу.

Протягом усього циклу розробки системи передбачено проведення багатоетапного тестування, включно з юзабіліті-тестуванням із залученням представників цільової аудиторії. Це дозволяє забезпечити відповідність програмного продукту практичним потребам користувачів і здійснювати своєчасні коригування. Після завершення валідаційних процедур розпочинається фаза

впровадження, у межах якої система поширюється через відповідні канали комунікації з науковою спільнотою. Завершальним етапом є супровід і підтримка системи, що включає усунення виявлених недоліків, оновлення функціональності відповідно до змін середовища використання та адаптацію до нових дослідницьких запитів.

Для опису роботи інформаційної системи з текстовими даними застосовується формальна модель корпусу, яка подає структуру вхідних текстових ресурсів у вигляді множини документів. Корпус позначається як:

$$C = \{d_1, d_2, \dots, d_N\},$$

де C – текстовий корпус як сукупність усіх текстових документів, з якими працює інформаційна система. Корпус слугує основним джерелом даних для статистичного, лінгвістичного чи аналітичного опрацювання; d_i – текстовий документ корпусу.

Документ є елементарною одиницею корпусу – окремим текстом, який може бути реченням, статтею, новинним повідомленням, фрагментом книги, повідомленням із соціальних мереж, будь-яким іншим структурованим чи неструктурованим текстом; $i = 1..N$ – індекс, що відповідає номеру документа у корпусі; N – кількість документів у корпусі. Це загальний обсяг текстових одиниць, з яких складається корпус. Від N залежить обсяг даних, які опрацьовує система; складність пошуку; час виконання аналітичних процедур; вимоги до зберігання та індексації.

Внутрішню структуру документа формалізуємо $d_i = \{w_{i1}, w_{i2}, \dots, w_{iM}\}$, де w_{ij} – окрема текстова одиниця документа – слово, токен, лема, морфема або інший елемент, залежно від рівня попереднього опрацювання. M – кількість токенів у документі d_i . Це величина, що варіюється для кожного документа.

Модель дозволяє трактувати документ як послідовність або множину елементарних одиниць, які система може аналізувати, тегувати, лематизувати, використовувати для статистичних запитів.

Подана формалізація дозволяє чітко визначити структуру даних, із якими працює система; застосовувати математичні операції до корпусу (фільтрація, агрегація, підрахунки); формалізувати процедури пошуку та аналізу; полегшити побудову алгоритмів *NLP* та корпусної аналітики.

Ця модель є універсальною основою для подальших функцій системи – від простого підрахунку частот до складних запитів із семантичними та морфологічними фільтрами.

Для опису роботи аналітичної підсистеми інформаційної системи вводиться формалізація,

яка показує, як саме система опрацює запити користувачів на основі корпусу текстів. Запит q інтерпретується як вимога виконати певну аналітичну операцію над корпусом C . Формально це подається таким чином:

$$q \Rightarrow A(q, C),$$

де q – користувацький запит, що визначає тип операції, яку необхідно виконати. Запит може містити ключові слова або патерни для пошуку; фільтри (морфологічні, семантичні, стилістичні); умови статистичного аналізу (частотність, колокації тощо); параметри візуалізації або агрегації результатів; C – корпус текстів, що є основою для виконання всіх аналітичних операцій. Система використовує саме цей корпус для пошуку, порівняння, статистики та інших функцій.

$A(q, C)$ – функція аналітичного опрацювання, яка виконує операцію, визначену запитом q . Типи операцій можуть включати пошук – знаходження текстових входжень; фільтрацію – відбір документів або tokenів за критеріями; статистику – частотний аналіз, n -грамні моделі, колокації; обчислення метрик – TF-IDF, PMI, частотні розподіли, діаграми.

Функція A формує відповідь на запит – набір результатів, що повертається користувачу.

Час, необхідний системі для опрацювання запиту, подається як сума трьох складових:

$$T(q) = T_{IO} + T_{parse} + T_{analysis},$$

де T_{IO} – час операцій введення/виведення. Включає всі дії, пов'язані з доступом до даних завантаження корпусу або його частин; зчитування індексів; доступ до файлової системи або СУБД; отримання метаданих. Цей час залежить від швидкодії дискової підсистеми, оптимізації індексів та масштабності корпусу; T_{parse} – час розбору та попереднього опрацювання. Це етапи токенизації; лематизації; морфологічного або синтаксичного аналізу; перетворення структури запиту; підготовки даних до аналізу. Особливо значимий у системах, що працюють з неопрацьованими текстами або складною лінгвістичною розміткою; $T_{analysis}$ – час виконання аналітичної операції. Містить виконання логіки функції $A(q, C)$ – пошук у корпусі й у структурах індексів; обчислення частот або статистичних моделей; фільтрацію результатів; генерацію візуалізацій; формування вихідної відповіді.

Цей компонент найчастіше є основним чинником загального часу опрацювання.

Подана формалізація дозволяє чітко структурувати етапи опрацювання запиту; оцінювати

швидкодію системи за окремими складовими; оптимізувати «вузькі місця» (I/O, парсинг, аналітику); порівнювати різні архітектурні рішення та підходи до індексації корпусних даних.

Модель також є базою для формування вимог до продуктивності системи, наприклад $T(q) \leq T_{\max}$, де $T_{\max}$ – допустимий час відповіді для комфортної взаємодії користувача.

Виконання цієї нерівності гарантує, що система залишатиметься зручною, ефективною та придатною для використання в наукових дослідженнях або освітніх цілях.

Висновки

У цій статті проаналізовано ключові аспекти управління науковим проєктом, спрямованим на створення інформаційної системи для управління корпусами українських текстів. На основі аналізу потреб цільової аудиторії, ідентифікації стейкхолдерів обґрунтовано доцільність застосування комбінованого підходу до управління, який інтегрує методології SCRUM, Lean та exploratory programming. Запропонований підхід дозволяє поєднати структурованість, економічність і гнучкість, що є критичними у контексті академічних IT-проєктів з обмеженими ресурсами. Також сформульовано стратегічні орієнтири щодо фазової моделі розробки системи, збору зворотного зв'язку, маркетингових стратегій і забезпечення якості.

Практична значущість дослідження полягає в тому, що результати роботи можуть бути безпосередньо використані як дорожня карта для реалізації подібних інформаційних систем у галузі прикладної лінгвістики. Запропоновані моделі управління, підходи до архітектури та механізми комунікації з користувачами можуть стати основою для розроблення функціональних програмних продуктів, що враховують специфіку академічного середовища.

Подальші дослідження авторів будуть присвячені деталізації архітектури системи управління корпусами текстів, побудові її окремих компонентів і реалізації функціонального прототипу відповідно до обґрунтованої в цій роботі комбінованої управлінської методології. У межах наступних етапів планується ітеративне вдосконалення прототипу, розгортання MVP, впровадження системи в тестових наукових середовищах, а також публікація результатів емпіричних досліджень з оцінки її ефективності. Застосування SCRUM, Lean і exploratory programming дозволить адаптувати процес реалізації до змін середовища, очікувань стейкхолдерів та технологічних викликів.

Список літератури

1. Сазонець І. Л., Ковшун Н. В. Управління науковими проєктами: сучасні підходи та механізми реалізації. *Економіка та держава*. 2021. № 6. С. 67–72. DOI: <https://doi.org/10.32702/2306-6806.2021.6.67>.
2. Якименко І., Штефан Є., Лук'янихін В. Управління науковими проєктами : навчальний посібник. Київ : Центр учбової літератури, 2020. URL: <https://cutt.ly/4wZjAnu>.
3. Проєктний менеджмент: управління наукомісткими проєктами та портфелями проєктів у наукомістких галузях : монографія / О. Б. Данченко та ін. Черкаси : Видавець Пономаренко Р. В., 2023. 315 с.
4. Храпкіна В. В., Матвієнко О. Д. Трансформація процесів розвитку інформаційної економіки та особливості повосного відновлення підприємств ІТ-сектору України. *Трансформація бізнесу в контексті глобальних викликів сталого розвитку* : монографія / під заг. ред. В. В. Храпкіної, К. В. Пічик. Київ : Видавничий дім «Києво-Могилянська академія», 2025. С. 168–191. URL: <https://ekmair.ukma.edu.ua/handle/123456789/34678>.
5. Martinez I., Viles E., Olaizola I. A survey study of success factors in data science projects. *Procedia Computer Science*. 2022. 200. P. 1295–1304. DOI: <https://doi.org/10.1016/j.procs.2022.01.285>.
6. Baxter A. L., Choi J., Gorton I., Shen S. Collaborative experience between scientific software projects using agile scrum development. *2012 SC Companion: High Performance Computing, Networking Storage and Analysis*. 2012. P. 1285–1292. DOI: <https://doi.org/10.1109/SC.Companion.2012.163>.
7. Milewicz R., Raybourn E. M. Talk to Me: A Case Study on Coordinating Expertise in Large-Scale Scientific Software Projects. *Proceedings of the 2014 Annual Conference on Extreme Science and Engineering Discovery Environment*. 2014. P. 1–8. DOI: <https://doi.org/10.1145/2616498.2616552>.
8. Johnson E. C., Leeman J., Ramamurthy M. K., Young J. SciOps: Achieving productivity and reliability in data-intensive research. *Data Science Journal*. 2021. 20(1). P. 1–15. DOI: <https://doi.org/10.5334/dsj-2021-017>.
9. Clarkson M. B. E. A stakeholder framework for analyzing and evaluating corporate social performance. *Academy of Management Review*. 1995. 20(1). P. 92–117. DOI: <https://doi.org/10.5465/amr.1995.9503271994>.
10. Mainardes E. W., Alves H., Raposo M. A model for stakeholder classification and stakeholder relationships. *Management Decision*. 2012. 50 (10). P. 1861–1879. DOI: <https://doi.org/10.1108/00251741211279648>.

Стаття надійшла до редколегії 03.12.2025

Kozak Ivan

Postgraduate Student of the Department of Information Systems and Networks,
<https://orcid.org/0009-0007-4953-2816>
Lviv Polytechnic National University, Lviv

Kunanets Nataliia

D.Sc. in Social Communications, Professor of the Department of Information Systems and Networks,
<https://orcid.org/0000-0003-3007-2462>
Lviv Polytechnic National University, Lviv

MANAGEMENT OF THE SCIENTIFIC IT PROJECT “DEVELOPMENT OF AN INFORMATION SYSTEM FOR THE ANALYSIS OF UKRAINIAN-LANGUAGE CORPORA”

Abstract. The article examines the complex issue of managing a scientific IT project aimed at developing an information system for working with Ukrainian-language text corpora of various genres and structural types. A systematic analysis of current approaches to organizing research IT projects under conditions of limited resources, a changing environment, uncertainty of technical priorities, different levels of stakeholder involvement, and a high variability of functional requirements formed at the intersection of linguistic and technical tasks has been carried out. A management model combining the SCRUM, Lean, and exploratory programming methodologies is proposed, providing a balance between structuredness, cost-efficiency, flexibility, adaptive planning, and the ability to respond quickly to changes in requirements during the evolution of the scientific concept. The target audience of the future system is defined, including applied linguists, instructors, students, translators, digital humanities researchers, and specialists working with corpus methods in education, applied studies, and domain-specific analytical tasks. A generalized socio-professional profile of the potential user is presented, and typical usage scenarios of the system in educational, scientific, and research contexts are described, including text analysis, subcorpus formation, annotation, statistical description, and preparation of data for experiments. Key stakeholders have been identified, and their classification by level of influence, interest, expected benefits, and interaction risks has been carried out. The use of a phased software development model is substantiated, including conceptualization, design, implementation, testing, deployment, and further maintenance, which enables formalization of the process while preserving flexibility. The results of the study may serve as a methodological basis for planning, organizing, and implementing scientific IT projects in the field of digital linguistics. The proposed approach is relevant for projects that combine scientific research with software development, involve a high level of requirement uncertainty, and require adaptive management decisions.

Keywords: research project management; information system; Ukrainian text corpora; SCRUM; Lean; exploratory programming; applied linguistics

References

1. Sazonets, I. L., & Kovshun, N. V. (2021). Management of scientific projects: modern approaches and mechanisms of implementation. *Economy and State*, 6, 67–72. <https://doi.org/10.32702/2306-6806.2021.6.67>.
2. Yakymenko, I., Shtefan, Ye., & Lukianikhin, V. (2020). *Management of scientific projects: a study guide*. Center of Educational Literature. <https://cutt.ly/4wZjAnu>.
3. Danchenko, O. B., et al. (2023). *Project management: management of high-tech projects and project portfolios in high-tech industries: a monograph*. Publisher Ponomarenko R. V.
4. Khrapkina, V. V., & Matviienko, O. D. (2025). Transformation of development processes of the information economy and features of post-war recovery of IT sector enterprises of Ukraine. In V. V. Khrapkina & K. V. Pichyk (Eds.), *Business transformation in the context of global challenges of sustainable development: a monograph* (pp. 168–191). Kyiv-Mohyla Academy Publishing House. <https://ekmair.ukma.edu.ua/handle/123456789/34678>.
5. Martinez, I., Viles, E., & Olaizola, I. (2022). A survey study of success factors in data science projects. *Procedia Computer Science*, 200, 1295–1304. <https://doi.org/10.1016/j.procs.2022.01.285>.
6. Baxter, A. L., Choi, J., Gorton, I., & Shen, S. (2012). Collaborative experience between scientific software projects using agile scrum development. *2012 SC Companion: High Performance Computing, Networking Storage and Analysis*, 1285–1292. <https://doi.org/10.1109/SC.Companion.2012.163>.
7. Milewicz, R., & Raybourn, E. M. (2014). Talk to Me: A Case Study on Coordinating Expertise in Large-Scale Scientific Software Projects. *Proceedings of the 2014 Annual Conference on Extreme Science and Engineering Discovery Environment*, 1–8. <https://doi.org/10.1145/2616498.2616552>.
8. Johnson, E. C., Leeman, J., Ramamurthy, M. K., & Young, J. (2021). SciOps: Achieving productivity and reliability in data-intensive research. *Data Science Journal*, 20 (1), 1–15. <https://doi.org/10.5334/dsj-2021-017>.
9. Clarkson, M. B. E. (1995). A stakeholder framework for analyzing and evaluating corporate social performance. *Academy of Management Review*, 20(1), 92–117. <https://doi.org/10.5465/amr.1995.9503271994>.
10. Mainardes, E. W., Alves, H., & Raposo, M. (2012). A model for stakeholder classification and stakeholder relationships. *Management Decision*, 50 (10), 1861–1879. <https://doi.org/10.1108/00251741211279648>.

Посилання на публікацію

- | | |
|------|--|
| APA | Kozak, I., & Kunanets, N. (2025). Management of the scientific IT project “Development of an information system for the analysis of Ukrainian-language corpora.” <i>Management of Development of Complex Systems</i> , 64, 84–93, dx.doi.org/10.32347/2412-9933.2025.64.84-93 . |
| ДСТУ | Козак І. В., Кунанець Н. Е. Управління науковим ІТ-проєктом «Розробка інформаційної системи аналізу україномовних корпусів». <i>Управління розвитком складних систем</i> . Київ, 2025. № 64. С. 84 – 93, dx.doi.org/10.32347/2412-9933.2025.64.84-93 . |