

**МІНІСТЕРСТВО ВНУТРІШНІХ СПРАВ УКРАЇНИ  
ЛЬВІВСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ ВНУТРІШНІХ СПРАВ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ УПРАВЛІННЯ, ПСИХОЛОГІЇ  
ТА БЕЗПЕКИ**

**Кафедра інформаційних технологій**

**ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ МОДУЛЯ ІНФОРМАЦІЙНОЇ  
СИСТЕМИ З ПІДТРИМКИ ІТ-ПРОЄКТІВ**

**Кваліфікаційна робота**  
здобувача вищої освіти  
4 курсу денної форми навчання  
**Данила ЛАЗОРА**

**Науковий керівник:**  
Старший викладач  
**Наталія ГАЛАЙКО**

**Рецензент:**

---

---

*Кваліфікаційна робота допущена до захисту*

«\_\_» «\_\_\_\_\_» 2026 р., протокол № \_\_\_\_\_

Завідувач кафедри інформаційних технологій

\_\_\_\_\_ Олег ЗАЧЕК

(підпис)

Львів  
2026

## СПИСОК УМОВНИХ СКОРОЧЕНЬ

API	Application Programming Interface (інтерфейс програмування застосунків)
PM	Product manager
SQL	Structured query language (мова структурованих запитів)
БД	База даних
ІС	Інформаційна система
ІТ	Інформаційні технології
ПЗ	Програмне забезпечення
ШІ	Штучний інтелект
СУБД	Система управління базами даних

## АНОТАЦІЯ

**ЛАЗОР Д.** Проектування та реалізація модуля інформаційної системи з підтримки ІТ-проектів. – Рукопис.

Кваліфікаційна робота на здобуття освітнього ступеня «бакалавр» за спеціальністю F6 – «інформаційні системи та технології». – Львівський державний університет внутрішніх справ, МВС України. Львів. 2026.

У кваліфікаційній роботі спроектовано та реалізовано модуль інформаційної системи, який призначений для підтримання процесів управління ІТ-проектами, що забезпечує ефективну організацію роботи колективу, управління завданнями та контроль виконання проектних робіт. З цією метою у роботі проаналізовано особливості управління ІТ-проектами та визначено основні вимоги до ІС підтримки таких процесів; досліджено існуючі програмні рішення і підходи до автоматизації управління ІТ-проектами та проведено їх порівняння; розроблено структурну й функціональну модель модуля інформаційної системи; реалізовано програмний модуль підтримки ІТ-проектів із використанням сучасних засобів і технологій розроблення програмного забезпечення; проведено тестування й оцінку ефективності роботи розробленого модуля.

Отримані результати розширюють підходи до створення модульних інформаційних систем управління ІТ-проектами та можуть бути використані під час проектування програмних засобів підтримки проектної діяльності.

**Ключові слова:** управління ІТ-проектами, інформаційна система, база даних, вебдодаток, технології управління ІТ-проектами.

## ABSTRACT

**LAZOR D.** Design and Implementation of a Module of an Information System for IT Project Support. – Manuscript.

Bachelor's qualification thesis in specialty F6 – *Information Systems and Technologies*. – Lviv State University of Internal Affairs, Ministry of Internal Affairs of Ukraine. Lviv, 2026.

In the qualification thesis, a module of an information system designed to support IT project management processes has been designed and implemented. The module ensures effective team collaboration, task management, and control over project execution. To achieve this goal, the thesis analyzes the specifics of IT project management and defines the main requirements for information systems supporting such processes; examines existing software solutions and approaches to IT project management automation and provides their comparison; develops structural and functional models of the information system module; implements a software module for IT project support using modern software development tools and technologies; and conducts testing and evaluates the efficiency of the developed module.

The obtained results extend the approaches to the development of modular information systems for IT project management and can be used in the design of software tools supporting project activities.

**Keywords:** IT project management, information system, database, web application, IT project management technologies.

## ЗМІСТ

ВСТУП.....	7
Розділ 1. ОКРЕМІ АСПЕКТИ АДМІНІСТРУВАННЯ ПРОЦЕСІВ	
УПРАВЛІННЯ ІТ-ПРОЄКТАМИ.....	13
1.1. Термінологія та фундаментальні поняття у галузі управління ІТ- проєктами.....	13
1.2. Етапи та організаційні аспекти управління ІТ-проєктами.....	18
1.3. Значення управління ІТ-проєктами та роль менеджера проєкту в команді.....	22
1.4. Функціональні аспекти та інформаційні підсистеми управління ІТ- проєктами.....	25
1.5. Окремі аспекти колективної діяльності під час підготовки ІТ-проєктів.....	27
Розділ 2. АНАЛІЗ МЕТОДОЛОГІЧНОГО ІНСТРУМЕНТАРІЮ	
УПРАВЛІННЯ ІТ-ПРОЄКТАМИ.....	30
2.1. Найпопулярніші методологічні інструменти управління ІТ-проєктами.....	30
2.1.1. Технологія «Водоспаду» («Waterfall»).....	32
2.1.2. Технологія «Agile».....	34
2.1.3. Технологія «Scrum».....	36
2.1.4. Технології «Kanban» і «Lean».....	38
2.2. Програмний інструментарій ІТ-проєктів.....	40
2.2.1. «Jira».....	42
2.2.2. «Redmine».....	44
2.2.3. «Trello».....	45
Розділ 3. РОЗРОБЛЕННЯ МОДУЛЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ	
ПІДТРИМКИ ІТ-ПРОЄКТІВ.....	47
3.1. Вебдодаток як елемент ІС.....	47
3.2. Основні переваги спроектованого модуля.....	48

	6
3.3. Проектування БД для модуля ІС управління ІТ-проєктами.....	49
3.4. Моделювання додатку ІС управління ІТ-проєктами.....	52
3.5 Архітектура модуля ІС управління ІТ-проєктами.....	55
3.6. Розроблення модуля управління ІТ-проєктами.....	57
ВИСНОВКИ.....	62
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	64

## ВСТУП

**Актуальність теми дослідження.** Управління ІТ-проєктами передбачає вирішення широкого спектра завдань, пов'язаних із плануванням, розподілом ресурсів, координацією діяльності учасників проєкту, відстеженням виконання завдань, контролем термінів й управлінням змінами. Особливої актуальності ці процеси набувають у середовищах, де над одним проєктом працює декілька фахівців або колективів, а сам процес проєктування потребує постійної взаємодії між різними ролями – керівниками, розробниками, тестувальниками й іншими учасниками. За відсутності належних інструментів підтримки такі процеси можуть супроводжуватися неузгодженістю дій, затримкою виконання завдань, неефективним використанням ресурсів та зниженням якості кінцевого результату.

Одним із шляхів підвищення ефективності управління ІТ-проєктами є використання спеціалізованих ІС, що забезпечують підтримку процесів планування, організації й моніторингу виконання проєктних робіт. Такі системи дозволяють структурувати інформацію про проєкти, формувати та керувати переліком завдань, контролювати строки виконання робіт, а також отримувати аналітичні дані щодо стану реалізації проєкту. Разом із тим існуючі програмні продукти у сфері управління проєктами часто орієнтовані на універсальне використання, що не завжди відповідає специфічним потребам окремих організацій або навчально-дослідницьких середовищ. Крім того, деякі з них характеризуються складністю впровадження, надмірною функціональністю або високою вартістю використання.

У зв'язку із сказаним **актуальним** є створення і вдосконалення окремих модулів ІС, спрямованих на підтримання процесів управління ІТ-проєктами. Модульна архітектура ІС дозволяє гнучко інтегрувати нові функціональні компоненти у вже існуюче інформаційне середовище,

забезпечуючи при цьому можливість адаптації системи до конкретних потреб користувачів. Реалізація спеціалізованого модуля з підтримки ІТ-проектів дає змогу автоматизувати процеси формування структури проекту, управління завданнями, розподілу відповідальності між учасниками у колективі, відстеження прогресу виконання робіт й аналізу стану реалізації проектів. Крім того, використання такого модуля сприяє підвищенню прозорості процесів управління проектами, покращенню комунікації між учасниками команди та ефективнішому прийняттю управлінських рішень. Автоматизація ключових процедур управління дозволяє зменшити кількість помилок, підвищити оперативність доступу до актуальних даних та забезпечити централізоване зберігання інформації про проекти.

Таким чином, розроблення та реалізація модуля інформаційної системи з підтримки ІТ-проектів є **актуальним** науково-практичним завданням, спрямованим на підвищення ефективності організації роботи команд розробників, оптимізацію процесів управління проектами та покращення якості інформаційної підтримки управлінської діяльності.

**Аналіз останніх досліджень і публікацій.** Питання управління ІТ-проектами та створення ІС підтримки проектної діяльності привертає значну увагу як українських, так і закордонних науковців. У сучасних умовах ІС управління проектами розглядаються не лише як інструменти планування, а як комплексні програмні середовища, що забезпечують підтримку процесів організації, контролю й аналізу виконання проектів.

Одним із напрямів сучасних досліджень є вивчення функціональних можливостей та ролі ІС управління проектами. У роботі [1] проведено систематизований огляд досліджень у розглядуваній сфері за останні роки. Автори зазначають, що сучасні системи підтримки управління проектами поступово трансформуються у комплексні платформи, які інтегрують інструменти планування, контролю виконання завдань, управління ресурсами, аналізу даних та комунікації між учасниками проекту. Дослідники

підкреслюють, що ефективність таких систем визначається не лише їх функціональністю, а й можливістю інтеграції з іншими сервісами організації.

Подібні результати наведено у дослідженні [2], у якому проведено статистичний аналіз функціональних можливостей сучасних програмних систем управління проєктами. У роботі визначено основні функціональні компоненти управління проєктами, до яких належать інструменти планування робіт, декомпозиції проєкту на завдання, управління ресурсами, контролю термінів виконання, моніторингу прогресу та формування звітності. Автори наголошують, що саме поєднання цих функцій у межах єдиного програмного середовища забезпечує ефективну підтримку управління проєктами.

Значна частина сучасних досліджень присвячена впливу гнучких методологій розроблення ПЗ на процеси управління ІТ-проєктами. У огляді [3] досліджено особливості використання agile-підходів у розробленні програмних систем. Автори відзначають, що поширення гнучких методологій призвело до зміни підходів до організації роботи проєктних команд, зокрема до переходу від жорсткого планування до ітеративної моделі управління, що передбачає короткі цикли розроблення, постійний зворотний зв'язок та швидке реагування на зміни вимог.

Окремі дослідження розглядають процеси інтеграції цифрових технологій у сферу управління проєктами. Зокрема у [4] проаналізовано вплив цифрової трансформації на розвиток методологій та інструментів проєктного менеджменту. Автори зазначають, що сучасні ІС повинні забезпечувати не лише підтримку базових функцій управління проєктами, а й інтеграцію з іншими цифровими платформами, що використовуються в організації.

Перспективним напрямом досліджень є використання методів ІІІ та аналітики даних у системах управління проєктами. У роботі [5] розглянуто можливості застосування ІІІ у сфері проєктного менеджменту. Дослідники зазначають, що сучасні ІС можуть використовувати алгоритми аналізу даних для прогнозування ризиків, оцінювання ймовірності затримки виконання

завдань та оптимізації розподілу ресурсів. Подібні підходи підвищують ефективність прийняття управлінських рішень у межах ІТ-проектів.

Розвиток інтелектуальних систем підтримки прийняття рішень у проєктному менеджменті також розглянуто у роботі [6]. Автор досліджує можливості використання систем підтримки прийняття рішень, що базуються на методах ШІ, для оптимізації управління ризиками та розподілу ресурсів у гнучких програмних проєктах. Отримані результати свідчать про перспективність використання аналітичних інструментів у системах підтримки управління ІТ-проєктами.

Окрему увагу в наукових дослідженнях приділено оцінюванню результативності інформаційних та програмних проєктів. У роботі [7] проаналізовано результати досліджень успішності ІТ-проєктів за останні десятиліття. Автори зазначають, що традиційні критерії успішності, такі як дотримання строків і бюджету, поступово доповнюються новими показниками, зокрема рівнем задоволеності користувачів, ефективністю використання системи та її впливом на діяльність організації.

В українських наукових дослідженнях також активно розглядаються питання управління ІТ-проєктами та використання програмних інструментів підтримки проєктної діяльності. Так, у роботі [8] проаналізовано сучасні методики управління ІТ-проєктами, інструменти їх реалізації та підходи до управління ризиками. Автори наголошують на необхідності використання спеціалізованих програмних систем для підтримки управління проєктами, що дозволяє підвищити ефективність планування та контролю виконання робіт.

У роботі [9] представлено огляд сучасних програмних систем управління проєктами, зокрема таких інструментів, як Jira, Trello та Asana. У дослідженні проведено порівняльний аналіз їх функціональних можливостей та визначено основні переваги й недоліки використання цих систем у процесі організації роботи проєктних команд. Результати дослідження свідчать про те, що універсальні програмні рішення не завжди враховують специфічні потреби

окремих організацій, що обумовлює необхідність створення адаптованих ІС або окремих програмних модулів.

Таким чином, аналіз сучасних наукових досліджень показує, що незважаючи на наявність великої кількості програмних інструментів управління проектами, існує потреба у створенні гнучких програмних рішень, які можуть бути адаптовані до специфічних умов роботи організації або проектною командою. Це обумовлює доцільність дослідження та розроблення модуля ІС з підтримки ІТ-проектів, що забезпечуватиме ефективну організацію проектною діяльності, управління завданнями та контроль виконання робіт.

**Мета** дослідження полягає у проектуванні та реалізації модуля інформаційної системи, призначеного для підтримання процесів управління ІТ-проектами, що забезпечує ефективну організацію роботи колективу, управління завданнями та контроль виконання проектних робіт.

Для досягнення означеної мети слід виконати наступні **завдання**:

- проаналізувати особливості управління ІТ-проектами та визначити основні вимоги до інформаційних систем підтримки таких процесів;
- дослідити існуючі програмні рішення та підходи до автоматизації управління ІТ-проектами та провести їх порівняння;
- розробити структурну й функціональну модель модуля ІС;
- реалізувати програмний модуль підтримки ІТ-проектів із використанням сучасних засобів та технологій розроблення ПЗ;
- провести тестування та оцінити ефективність роботи розробленого модуля.

**Об'єктом** дослідження у роботі є процеси управління ІТ-проектами в інформаційних системах.

**Предметом** дослідження є методи, моделі та програмні засоби проектування і реалізації модуля інформаційної системи, призначеного для підтримки управління ІТ-проектами.

**Методи досліджень.** У процесі виконання роботи використано комплекс загальнонаукових і спеціальних методів дослідження. Зокрема, застосовано методи аналізу та узагальнення наукових джерел для дослідження сучасних підходів до управління ІТ-проектами та функціонування ІС підтримки проєктної діяльності. Метод системного аналізу використано для визначення структури та функціональних вимог до модуля інформаційної системи. Методи структурного та об'єктно-орієнтованого проєктування застосовано під час розроблення архітектури програмного модуля та моделювання його основних компонентів. Для реалізації програмного забезпечення використано сучасні технології та засоби розроблення програмних систем. Перевірку працездатності розробленого модуля здійснено із застосуванням методів програмного тестування й аналізу результатів його функціонування.

**Структура роботи.** Кваліфікаційна робота складається із вступу, трьох розділів, висновків, списку використаних джерел. Обсяг основного тексту роботи складає 55 сторінок, 8 рисунків, 2 лістинги і 28 бібліографічних джерел. Загальний обсяг роботи – 67 сторінок.

## Розділ 1.

# ОКРЕМІ АСПЕКТИ АДМІНІСТРУВАННЯ ПРОЦЕСІВ УПРАВЛІННЯ ІТ-ПРОЄКТАМИ

### 1.1. Термінологія та фундаментальні поняття у галузі управління ІТ- проєктами

Управління ІТ-проєктами – це про перетворення ідеї (наприклад, створити сайт, мобільний застосунок або інформаційну систему) на реальний результат у визначені терміни, з певним бюджетом і з потрібною якістю. З боку це може здаватися просто «організацією роботи», але насправді тут є чимало важливих понять і термінів, які допомагають тримати процес під контролем. Ефективність управління ІТ-проєктами нині під час стрімкого розвитку інформаційних технологій є надзвичайно актуальним, так як ці технології займають провідне місце у розвитку практично усіх сфер людської діяльності.

Правильно організований процес керування розробленням проєкту у ІТ-галузі має надзвичайно важливе значення. Неефективне адміністрування здатне привести до затримання процесів упровадження проєкту, перевищення фінансування, а інколи і до відставання від розробок конкурентів.

На початку нашого дослідження обговоримо основні поняття та дефініції, які є ключовими для усвідомлення й упровадження теоретичних засад управління ІТ-проєктами.

Питанням застосування термінологічних понять у галузі управління ІТ-проєктами присвячено значну кількість дослідницьких робіт, наприклад, [10–21], однак існує необхідність провести узагальнення цих наукових пошуків, зосередивши основний акцент на визначенні суті кожного із розглядуваних понять та їх значення у досягненні ефективної діяльності у ІТ-галузі.

*IT-проект* – це тимчасова робота з чіткою метою (чітким завданням), яка пов’язана зі створенням або вдосконаленням цифрового продукту. Він має початок і кінець. Наприклад, розроблення вебсайту для закладу вищої освіти – це проект, бо є такі ознаки як завдання, команда (підрозділ), строки і результат. Сюди також слід додати бюджетування. А от постійна технічна підтримка сайту – це вже не проект, а операційна діяльність.

Ключова риса проекту у IT-галузі є *унікальність*. Кожен новий розроблювальний застосунок або інформаційна система має свої особливості, навіть якщо команда вже робила щось подібне.

Кожен IT-проект характеризується *метою і цілями*. *Мета* – це загальний бажаний результат, який необхідно отримати після завершення проекту. Наприклад, спроектувати онлайн-платформу для освітнього процесу закладу вищої освіти. *Цілі* – це конкретніші кроки, які сприяють досягненню мети: розробити особистий кабінет здобувача освіти, додати відео- та текстові лекції, забезпечити можливість проведення онлайн-тестування (дистанційного контролю знань).

Добре сформульовані цілі повинні бути зрозумілими, вимірюваними та реалістичними. Якщо під час розроблення проекту поставити ціль «зробити зручний сайт», то це нечітко. А от ціль «зменшити час реєстрації користувача до 2 хвилин» – це уже конкретніше.

*Стейкхолдери (або зацікавлені сторони)* – це усі, кого стосується проект. Сюди слід віднести замовників, користувачів, керівництво установи, команда проектувальників і навіть інвестори. У кожного із них власні очікування. Основна задача керівника проекту – максимально врахувати інтереси кожного із переліченої категорії і не допустити конфлікту між ними. Наприклад, замовник хоче швидко і дешево, команда – якісно і без поспіху, користувач – просто і зручно. Дотримувати баланс між цими очікуваннями і є частиною процесу управління.

*Менеджер проєкту* – це людина, яка координує всю роботу. Він не обов’язково пише код, але відповідає за строки, раціональне використання бюджету, комунікацію із стекхолдерами та за підсумковий результат. У випадку виникнення будь-яких проблем – приймає заходи щодо їх вирішення. Якщо існують ті чи інші ризики, то намагається їх передбачити і не допустити.

У ІТ-галузі менеджера проєкту часто називають продакт-менеджером (product manager (PM)). PM більше думає про те, яким має бути продукт і яку цінність він приносить користувачам. А менеджер проєкту зосереджується на процесі реалізації.

*Обсяг робіт* – за суттю це перелік того, що саме має бути зроблено у межах проєкту. Якщо обсяг чітко не визначити, може виникнути певна плутанина. Наприклад, замовник просить «додати ще одну маленьку функцію», потім ще одну, що у результаті призводить до виходу проєкту за межі бюджету та строків його виконання. Тому важливо з самого початку погодити, що входить у проєкт, а що – ні. Будь-які зміни в обсязі повинні обговорюватися окремо.

*Терміни (дедлайни)* – кінцева(-ві) дата(и) виконання роботи. У проєкті є загальний строк завершення і проміжні етапи. Часто роботу ділять на частини (етапи або спринти), щоб поступово перевіряти результат. Якщо дедлайни нереалістичні, команда працює у стресовій ситуації, зростає кількість помилок. Тому планування часу є однією із ключових частин менеджменту.

*Бюджет* – це фінансові ресурси, які виділені на проєкт. Він включає зарплати команди, оплату серверів, ліцензій, обладнання й інші витрати. Завдання менеджера проєкту не виходити за межі бюджетування або завчасно попередити, якщо витрати зростають.

*Ресурси проєкту* Сюди відносять усе, що потрібно для виконання проєкту: люди, час, технічне та програмне забезпечення, фінансові ресурси.

*Людські ресурси* – це команда: програмісти, тестувальники, дизайнери, аналітики. Тут РМ важливо правильно розподілити навантаження. Якщо на одного проєктувальника покласти занадто багато задач, це може вплинути на якість проєкту і строки його завершення.

*Життєвий цикл проєкту.* Будь-який проєкт проходить кілька етапів: ініціація (ідея і погодження), планування, виконання, контроль і завершення. На початку визначають потреби і можливості. Потім складають план. Далі – реалізація. Паралельно здійснюється перевірка і контроль виконання плану. І, зрештою, підбивають кінцеві підсумки.

*Методології управління.* У сфері ІТ популярні різні підходи до організації роботи. Класичний підхід часто називають «водоспадним» («Waterfall»). Спочатку все детально планується, потім послідовно виконується. Повернутися назад складно.

Гнучкі підходи («Agile») передбачають роботу невеликими частинами з постійним зворотним зв'язком. Серед них поширені «Scrum» і «Kanban». У «Scrum» робота поділяється на короткі цикли (спринти), зазвичай 2–4 тижні. У «Kanban» акцент робиться на візуалізації задач і рівномірному навантаженні. Гнучкі методи особливо зручні там, де вимоги можуть змінюватися.

*Завдання і беклог.* Під завданням розуміють конкретний обсяг робіт, який потрібно виконати. Усі заплановані задачі збирають у список, який отримав назву – беклог. У гнучких методологіях беклог постійно переглядають і пріоритезують. Під час пріоритизації списку визначають, які задачі є важливішими за інші. Спочатку виконують те, що має найбільшу цінність або критичність.

*Якість* – це відповідність результату очікуванням і вимогам. У ІТ-сфері важливо не тільки, щоб система працювала, а й щоб вона була стабільною, безпечною і зручною для користувача.

*Тестування* – окремий напрямок у проєкті. Тестувальники перевіряють, чи все працює правильно, чи немає помилок.

*Ризик* – потенційна проблема, яка може виникнути в майбутньому. Наприклад, ключовий розробник може захворіти або замовник може змінити вимоги. Управління ризиками полягає в тому, щоб заздалегідь продумати, що може піти не так, і підготувати план дій.

*Комунікація* – це обмін інформацією між усіма учасниками. Регулярні зустрічі, звіти, обговорення – усе це допомагає уникнути непорозумінь. Багато проблем у проєктах виникають саме через нестачу або неправильне інформування.

*Документація* – письмове описання вимог, рішень, інструкцій. Вона допомагає новим учасникам швидше включитися в роботу і зменшує залежність від окремих людей. У реаліях вимоги рідко залишаються незмінними. Управління змінами означає, що кожна нову ідею або правку потрібно оцінити: як вона вплине на строки, бюджетування і інші ресурси.

*Команда* – основа проєкту. Успіх залежить не лише від технологій, а й від взаємодії людей. Довіра, відповідальність і чіткий розподіл ролей є важливими аспектами під час створення проєкту.

*Інформаційна безпека*. Розроблювальний проєкт може включати відомості, що входять до комерційної таємниці установи чи відомості, які захищаються діючим законодавством. Для прикладу: фінансові або персональні дані. Такі аспекти обумовлюють труднощі, пов'язані з упровадження ІТ-проєкту.

У підсумку констатуємо, що управління ІТ-проєктами передбачає поєднання планування, організації, контролю й постійного спілкування. Тут важливо враховувати обсяг робіт, строки, бюджет, якість, ризики й інтереси різних сторін. Попри наявність різних методологій і термінів, суть залишається простою: потрібно організувати людей і процес так, щоб ідея стала реальним, працюючим продуктом. Головним у цій галузі є гнучкість, відповідальність і вміння бачити картину в цілому. Саме це дозволяє довести ІТ-проєкт від задуму до успішного завершення.

Головною особливістю, яка різнить ІТ-проекти від проектів у інших сферах, є те, що менеджмент у цій галузі стикається з нематеріальним результатом діяльності, який може існувати лише у віртуальному просторі. Ці проекти володіють унікальними властивостями, що значимо впливає на успіх реалізації.

## **1.2. Етапи та організаційні аспекти управління ІТ-проектами**

Управління ІТ-проектами на практиці – це вміння організувати час людські, фінансові і матеріально-технічні ресурси так, щоб ідея стала реальним продуктом. Незалежно від масштабу, у більшості ІТ-проектів є спільна логіка розвитку та схожі організаційні виклики. Нижче приведемо розгорнутий опис основних етапів і важливих організаційних аспектів, які притаманні даному виду діяльності.

Будь-який проект стартує з ідеї або потреби. Наприклад, компанія розуміє, що їй потрібна нова CRM-система, бо стара вже не відповідає сучасним вимогам. Або університет хоче створити онлайн-платформу для дистанційного навчання. На цьому етапі важливо відповісти на кілька базових питань: Навіщо це потрібно? Яку проблему ми вирішуємо? Хто буде користуватися результатом? Чи є ресурси для реалізації? Цей етап назвемо *ініціацією* – це ще не планування, а радше «перевірка реальності». Якщо ідея звучить красиво, але немає бюджету або команди, проект може так і не розпочатися. Далі, зазвичай, формується короткий опис проекту – щось на кшталт концепції або обґрунтування. На цьому етапі також визначають основних зацікавлених осіб: замовника, майбутніх користувачів, керівника проекту.

Наступний етап – *планування*. Планування — один із найвідповідальніших етапів. Саме тут формується «каркас» майбутньої роботи. Якщо його пропустити або зробити формально, проблеми виникнуть уже на стадії виконання. Під час планування визначають *обсяг робіт* – чітко

фіксується, що саме буде зроблено. Наприклад, якщо створюється сайт, чи входить у проєкт розроблення особистого кабінету? Чи буде створюватися мобільна версія? Чи передбачена інтеграція з іншими системами (сервісами)? Нечіткий обсяг робіт є однією із найпоширеніших причин конфліктів. Замовник очікує більше, команда планує менше, і в результаті усі незадоволені.

Кожен проєкт має загальний *дедлайн* і *проміжні* етапи. Часто роботу ділять на фази або короткі цикли (наприклад, по два тижні). Це дозволяє поступово контролювати результат і не чекати пів року, щоб зрозуміти, що щось пішло не так.

*Бюджетування.* Сюди входять зарплати команди, витрати на сервери, ліцензії, інструменти, можливо на консультації сторонніх фахівців. Навіть якщо проєкт внутрішній і гроші не передаються напряму, витрати все одно треба враховувати.

*Ресурси.* Мова йде не лише про гроші, а й про людей. Скільки розробників потрібно? Чи є тестувальник? Хто відповідатиме за дизайн? Якщо команда перевантажена іншими задачами, строки можуть «поплисти».

*Ризики.* На цьому етапі варто чесно подумати: що може піти не так? Наприклад, затримка з боку замовника, технічні складнощі, нестача фахівців. Краще передбачити проблему наперед, ніж реагувати на неї в паніці.

*Формування команди та розподіл ролей.* ІТ-проєкт – це завжди командна робота. Навіть невеликий продукт рідко створює одна людина. Зазвичай у команді є: керівник проєкту або координатор, розробники, тестувальники, дизайнер, аналітик, DevOps-фахівець (або адміністратор). Важливо не лише зібрати людей, а й чітко визначити, хто за що відповідає. Якщо відповідальність розмита, виникає ситуація, коли «це мав зробити хтось інший». Не менш важливий психологічний клімат. Конфлікти, непорозуміння, брак комунікації можуть звести нанівець навіть найкращий технічний план.

*Виконання проєкту.* Під час цього етапу команда переходить від планів до конкретних дій. Пишеться код, створюється дизайн, налаштовується інфраструктура. Тут важливо регулярно перевіряти прогрес, підтримувати зв'язок із замовником, вчасно реагувати на проблеми, не допускати хаотичних змін. Часто використовуються короткі робочі цикли, наприкінці яких демонструють проміжний результат. Це допомагає замовнику бачити розвиток проєкту, а команді – отримувати зворотний зв'язок.

*Контроль і моніторинг.* Контроль – це не недовіра, а спосіб не втратити напрям. Менеджер проєкту відстежує: чи вкладається команда у строки, чи не перевищено бюджет, чи відповідає результат вимогам, чи не з'явилися нові ризики. Якщо виявляються відхилення, потрібно коригувати план. Іноді це означає змінити пріоритети, додати ресурс або домовитися про перенесення дедлайну.

*Керування змінами.* У реаліях вимоги змінюються майже завжди. Замовник може побачити проміжний результат і зрозуміти, що хоче трохи інакше. З'являються нові технології або обмеження. Важливо, щоб зміни проходили не хаотично, а через обговорення й оцінку. Кожна зміна впливає на строки й бюджет. Якщо це не враховувати, проєкт може нескінченно розширюватися.

*Забезпечення якості проєкту.* Якість – це не тільки відсутність помилок. Це ще й зручність, швидкість роботи, безпека. У ІТ-проєктах тестування – окрема важлива складова. Тестування може бути функціональним (чи все працює як задумано); технічним (чи система витримує навантаження); користувацьким (чи зручно нею користуватися). Краще виявити помилку до запуску, ніж виправляти її вже після того, як продукт використовують користувачі.

*Комунікація.* Багато проблем виникає не через складність технологій, а через нестачу інформації. Якщо команда не знає про зміну вимог, вона

продовжує працювати за старим планом. Якщо замовник не бачить прогресу, з'являється недовіра. Регулярні зустрічі, короткі звіти, обговорення складних моментів є частиною організаційної культури проєкту.

*Документація.* Навіть у гнучких підходах повністю без документації не обійтись. У документації фіксуються вимоги, технічні рішення, досягнуті домовленості, інструкції для користувачів. Документація допомагає новим членам команди швидше включитися у роботу і зменшує залежність від окремих людей.

*Завершення проєкту.* На цьому етапі потрібно передати продукт замовнику, підготувати інструкції для використання продукту, провести фінальне тестування, зрештою, підбити підсумки проєкту. Корисним буде також проведення внутрішнього аналізу, під час якого встановлюють, що вдалося добре, а що можна покращити у наступних проєктах. Такий досвід є безцінний для розвитку команди.

Існують і інші важливі організаційні аспекти. Це:

- гнучкість – у ІТ усе швидко змінюється, тому надто жорсткі плани можуть заважати. Вміння адаптуватися – ключова риса сучасного управління;
- пріоритети – не все однаково важливе. Часто доводиться обирати: зробити більше функцій чи вкластися у строк, додати нову можливість чи зосередитися на стабільності;
- баланс між швидкістю і якістю – у поспіху зростає кількість помилок. Якщо занадто довго відшліфувати деталі, можна втратити актуальність продукту;
- мотивація команди – навіть найкращий план не спрацює, якщо люди не зацікавлені у результаті. Визнання, зрозумілі цілі, адекватне навантаження – усе це впливає на успіх.

Підсумовуючи можемо констатувати: управління ІТ-проєктами – це поєднання логіки, організації та людського фактору. Воно охоплює етапи від

появи ідеї до передачі готового продукту. Важливо чітко планувати, але й залишати місце для гнучкості. Потрібно контролювати строки й бюджет, але не забувати про якість і командну атмосферу. Центральне місце у проєкті займають люди: замовники, користувачі, команда. Технології можуть змінюватися, інструменти – оновлюватися, але основа успішного ІТ-проєкту завжди залишається однаковою: зрозуміла мета, налагоджена організація роботи та відповідальне ставлення до результату.

### **1.3. Значення управління ІТ-проєктами та роль менеджера проєкту в команді**

Управління ІТ-проєктом є важливою складовою створення будь-якого програмного продукту. За кожним застосунком, вебсайтом або інформаційною системою стоїть команда людей, які працюють над його створенням. Однак, сама наявність команди ще не гарантує успіху. Без чіткого менеджменту робота може стати неузгодженою, завдання виконуватимуться хаотично, а результат може суттєво відрізнятись від того, що очікував замовник. Саме тому управління ІТ-проєктом відіграє надзвичайно важливу роль у процесі проєктування.

Саме керування проєктом дозволяє впорядковувати процес створення будь-якого програмного продукту. Воно допомагає перетворити велику ідею на чіткий план дій. Коли усі учасники команди розуміють, що саме потрібно зробити, у які терміни та з якою метою, робота стає значно ефективнішою. Управління також допомагає уникнути ситуацій, коли декілька людей виконують одну й ту саму роботу або, навпаки, коли важливі завдання залишаються без відповідального виконавця.

Іще одним важливим аспектом управління ІТ-проєктом є контроль за ресурсами. У будь-якому проєкті існують обмеження – це час, бюджет і людські ресурси. Якщо ними не керувати, витрати можуть швидко збільшитися, а строки виконання постійно переноситимуться. Правильне

управління допомагає оптимально використовувати доступні ресурси та досягати поставленої мети без зайвих витрат.

Крім цього, управління проектом допомагає зменшити ризики. Під час розроблення ПЗ часто виникають непередбачувані ситуації. Наприклад, можуть з'явитися технічні труднощі, змінитися вимоги замовника або виникнути проблеми із інтеграцією різних частин системи. Якщо такі ризики не враховувати заздалегідь, вони можуть суттєво вплинути на результат. Управління проектом дозволяє передбачати подібні проблеми та знаходити шляхи їх вирішення ще до того, як вони стануть критичними.

Центральною фігурою в управлінні IT-проектом є проєктний менеджер. Його роль полягає у тому, щоб організувати роботу команди та забезпечити досягнення поставленої мети. У певному сенсі його можна порівняти з координатором, який допомагає різним спеціалістам працювати разом і рухатися в одному напрямку. Він не завжди бере безпосередню участь у технічній частині розробки, але саме він відповідає за те, щоб робота відбувалася організовано та ефективно.

Однією з основних задач проєктного менеджера є планування роботи. Перед початком розробки він разом із командою визначає, які етапи необхідно пройти, скільки часу вони можуть зайняти та які ресурси знадобляться для їх виконання. Це дозволяє створити чіткий план, за яким команда буде працювати протягом усього проєкту.

Також проєктний менеджер відіграє важливу роль у спілкуванні із замовником. Саме він часто виступає посередником між клієнтом і командою розробників. Менеджер з'ясовує вимоги до продукту, уточнює деталі, а потім передає цю інформацію команді у зрозумілому вигляді. Завдяки цьому розробники можуть краще зрозуміти очікування замовника та створити продукт, який максимально відповідає його потребам.

Під час роботи над проєктом менеджер також слідкує за тим, щоб команда дотримувалася встановлених строків. Якщо виникають затримки або

складнощі, він намагається знайти причину та допомогти її усунути. Іноді це може означати перерозподіл завдань між членами команди або зміну плану роботи. Головна мета в таких ситуаціях – не допустити того, щоб проблеми накопичувалися та впливали на весь проєкт.

Наступною важливою функцією проєктного менеджера є підтримка команди. У процесі розробки люди можуть стикатися з різними труднощами: складними технічними задачами, браком часу або навіть професійним вигоранням. Хороший менеджер намагається створити комфортні умови для роботи, допомагає вирішувати конфлікти та підтримує позитивну атмосферу у команді. Це важливо, адже ефективність роботи багато у чому залежить від того, наскільки добре взаємодіють між собою учасники проєкту.

Крім того, проєктний менеджер відповідає за контроль якості виконаної роботи. Він стежить за тим, щоб кожен етап розробки відповідав поставленим вимогам і щоб кінцевий продукт був надійним та зручним у використанні. Для цього менеджер співпрацює з тестувальниками та іншими спеціалістами, які перевіряють програмне забезпечення перед його запуском.

Успішний проєктний менеджер повинен мати не лише організаційні навички, а й добре розуміти принципи роботи команди та особливості процесу розроблення ПЗ. Він повинен уміти аналізувати ситуацію, приймати рішення та швидко реагувати на зміни. Водночас важливими є комунікаційні навички, адже значна частина його роботи пов'язана зі спілкуванням з людьми.

Зрештою, значення проєктного менеджера полягає у тому, що він допомагає перетворити ідею на реальний продукт. Без належного менеджменту навіть хороша ідея може залишитися нереалізованою або бути реалізованою неякісно. Натомість грамотна організація роботи дозволяє команді поступово рухатися від початкової концепції до готового рішення, яке можна використовувати на практиці.

Таким чином, управління ІТ-проєктом є важливим елементом сучасної розробки ПЗ. Воно забезпечує порядок у роботі команди, допомагає

контролювати ресурси, зменшує ризики та сприяє ефективній взаємодії між усіма учасниками процесу. Проектний менеджер у цьому процесі виконує роль координатора, який об'єднує зусилля команди та спрямовує їх на досягнення спільної мети. Саме завдяки його роботі складний процес розроблення ПЗ стає більш зрозумілим, організованим і результативним.

#### **1.4. Функціональні аспекти та інформаційні підсистеми управління ІТ-проєктами**

Однією з головних функцій управління ІТ-проєктом є планування. Перед початком роботи потрібно зрозуміти, який результат необхідно отримати, які етапи потрібно пройти та скільки часу це може зайняти. На цьому етапі формується загальний план роботи, визначаються основні завдання та приблизні строки їх виконання.

Не менш важливою є функція організації роботи. Після того як створено план необхідно правильно розподілити завдання між учасниками команди. Кожен спеціаліст повинен знати свою роль у проєкті та розуміти, за яку частину роботи він відповідає.

Ще однією важливою функцією є координація діяльності команди. Під час реалізації проєкту різні частини системи створюються паралельно, тому необхідно постійно узгоджувати роботу між різними спеціалістами. Наприклад, програмісти повинні взаємодіяти з дизайнерами, а результати їхньої роботи мають перевіряти тестувальники.

Важливою складовою управління є також контроль виконання роботи. Під час реалізації проєкту необхідно регулярно перевіряти, чи відповідає виконана робота запланованим вимогам і чи не виникають затримки. Якщо певні завдання виконуються повільніше, ніж очікувалося, потрібно вчасно виявити причину та знайти спосіб вирішення проблеми.

Окрім організаційних функцій, важливу роль у сучасному менеджменті ІТ-проєктами відіграють інформаційні підсистеми. Вони

являють собою програмні інструменти або спеціальні системи, які допомагають зберігати, опрацьовувати й передавати дані, пов'язані з проєктом. Завдяки таким системам команда може швидше обмінюватися відомостями, контролювати хід роботи й отримувати актуальну інформацію про стан проєкту.

Однією з основних інформаційних підсистем є система планування й управління завданнями. У таких системах зберігається інформація про усі завдання, які потрібно виконати у межах проєкту. Кожне завдання має опис, відповідального виконавця та визначений строк виконання. Завдяки цьому члени команди можуть бачити загальну картину роботи і розуміти, на якому етапі знаходиться проєкт.

Іншою важливою підсистемою є система комунікації між учасниками проєкту. Оскільки в IT-проєктах часто працюють люди з різних міст або навіть країн, важливо мати зручні інструменти для обміну інформацією. Такі системи дозволяють швидко обговорювати робочі питання, передавати файли, узгоджувати зміни та ділитися результатами роботи. Завдяки цьому команда може працювати злагодженіше навіть на відстані.

Також важливою є підсистема зберігання й управління документацією. У будь-якому IT-проєкті створюється велика кількість інформації: технічні описи, інструкції, вимоги до системи, результати тестування й інші матеріали. Якщо ці дані зберігати без системи, знайти потрібну інформацію буде складно. Інформаційні підсистеми дозволяють впорядкувати документацію та забезпечити доступ до неї для усіх членів команди.

Ще одним важливим елементом є підсистема контролю та звітності. Вона допомагає відстежувати прогрес роботи та оцінювати ефективність виконання завдань. За допомогою таких інструментів менеджери можуть бачити, які завдання вже виконані, які перебувають у процесі виконання, а які ще тільки заплановані. Це дає можливість швидше реагувати на зміни та приймати необхідні управлінські рішення.

Використання інформаційних підсистем значно полегшує процес управління ІТ-проектами. Вони допомагають автоматизувати багато процесів, які раніше виконувалися вручну, зменшують кількість помилок і дозволяють швидше отримувати потрібну інформацію. Завдяки цьому команда може більше часу приділяти безпосередньо проектуванню продукту, а не організаційним питанням.

Таким чином, основні функції менеджменту під час підготовки ІТ-проекту пов'язані з плануванням, організацією, координацією та контролем виконання завдань. Водночас важливу роль відіграють інформаційні підсистеми, які допомагають опрацьовувати відомості, підтримувати комунікацію між учасниками проекту та контролювати хід роботи. Поєднання цих елементів дозволяє створювати якісні ІТ-продукти та успішно реалізовувати навіть складні проекти.

### **1.5. Окремі аспекти колективної діяльності під час підготовки ІТ-проектів**

Колективна діяльність у сфері ІТ передбачає співпрацю різних фахівців. Зазвичай у команду входять програмісти, тестувальники, дизайнери, аналітики, менеджери проектів та інші спеціалісти. Кожен із них виконує свою частину роботи, але водночас усі працюють над спільною метою – створенням якісного програмного продукту. Саме тому важливо дотримуватися певних аспектів організації спільної діяльності.

Першим важливим аспектом є *чіткий розподіл ролей і обов'язків*. У колективі кожен учасник повинен добре розуміти, за що саме він відповідає. Наприклад, програмісти займаються написанням коду, дизайнери створюють зовнішній вигляд інтерфейсу, тестувальники перевіряють програму на помилки, а менеджер координує роботу усієї команди. Такий розподіл допомагає уникнути плутанини та дублювання роботи, а також дозволяє кожному спеціалісту зосередитися на своїй сфері.

Другим важливим принципом є *ефективна комунікація* між членами колективу. У процесі створення ІТ-проєкту постійно виникають питання, уточнення або нові ідеї. Якщо у колективі не відбувається обмін інформацією, це може призвести до помилок або непорозумінь. Тому важливо регулярно проводити зустрічі, обговорювати прогрес роботи, ділитися проблемами та спільно шукати рішення. У сучасних колективах для цього часто використовують різні інструменти: месенджери, системи управління проєктами, відеоконференції й інші онлайн-сервіси.

Ще одним важливим аспектом є *спільна відповідальність за результат*. Хоча кожен учасник має власні завдання, кінцевий результат залежить від роботи всього колективу. Якщо один елемент проєкту буде виконаний неякісно, це може вплинути на всю систему. Тому члени колективу мають підтримувати одне одного, допомагати у вирішенні складних завдань і разом працювати над покращенням продукту.

Важливу роль у колективній діяльності відіграє також *планування роботи*. Перед початком проєкту необхідно визначити основні етапи розробки, терміни виконання завдань та очікувані результати. Це сприяє покращенню організації праці й ефективному використанню часу.

Ще одним важливим принципом є *відкритість до ідей та конструктивна критика*. У командній роботі кожен учасник може запропонувати нові рішення або підходи. Обговорення різних варіантів часто допомагає знайти ефективніший спосіб реалізації проєкту. Водночас, важливо, щоб критика була конструктивною та спрямованою на покращення результату.

Крім того, важливим фактором успішної командної роботи є *взаємна повага та довіра між учасниками*. Коли члени команди поважають думку один одного і довіряють професійним навичкам колег, робота проходить продуктивніше та комфортніше. Це також допомагає швидше вирішувати конфлікти та знаходити компроміси у складних ситуаціях.

Не менш важливим аспектом є *постійне вдосконалення та навчання*. Сфера інформаційних технологій швидко змінюється, тому спеціалісти повинні постійно розвивати свої навички та вивчати нові технології. У командній роботі це особливо важливо, оскільки нові знання одного учасника можуть бути корисними для всієї команди.

Отже, колективна діяльність є ключовим елементом підготовки та реалізації ІТ-проектів. Ефективна співпраця між членами колективу, чіткий розподіл обов'язків, відкрита комунікація, спільна відповідальність та взаємна підтримка допомагають створювати якісні програмні продукти.

## Розділ 2.

# АНАЛІЗ МЕТОДОЛОГІЧНОГО ІНСТРУМЕНТАРІЮ УПРАВЛІННЯ ІТ-ПРОЄКТАМИ

## 2.1. Найпопулярніші методологічні інструменти управління ІТ- проєктами

Методологічні інструменти управління ІТ-проєктами передбачають сукупність принципів, процесів, а також практичних аспектів, які спрямовуються на ефективність адміністрування командної діяльності та досягнення мети проєкту. Такі підходи створюються із врахуванням характеристик галузі, технологічних аспектів і вимог колективу. Це надає можливість менеджменту підготувати структуру, усвідомити потенційні ризики й знайти потрібні рішення.

Зважаючи на те, що у різних галузях існує багато різноманітних проєктів, не є можливим розроблення єдиної методології, яка добре працюватиме для усіх сфер. Встановлено, що будь-яка методологія має свої позитивні особливості й обмеження, які можуть змінюватися в залежності від характеру завдань, процесу діяльності й кінцевої мети [22]. Для прикладу, загальноприйняті методи менеджменту, такі як «Waterfall», придатні для проєктів, які є лінійними. Однак, у динамічній ІТ-галузі вони здатні заважати розвитку й адаптуванню. З названої причини колективи проєктувальників мають створювати і використовувати гнучкіші підходи, наприклад, такий як «Agile», для того, щоб адаптуватися до нових вимог і швидшого виконання завдань.

З плином часу методи, що використовуються для проєктування ПЗ, почали використовувати в інших сферах суспільних відносин. Для прикладу, аспекти принципу «Lean», які на початку застосовувалися в автомобілебудуванні, ефективно інтегрувалися в управління ІТ-проєктами.

Це вказує на гнучкість й універсальність існуючих методологій, які можуть бути адаптовані до потреб відповідних проєктів.

Одним із найбільш поширених методів є метод «Waterfall». Він передбачає послідовне здійснення циклів проєкту: від планування до передачі проєкту. Цей метод придатний для проєктів, у яких вимоги чітко означені, а ризики мінімальні. Однак, жорсткі вимоги цієї структури моделі знижує її ефективність за умови виникнення подій, які важко передбачати.

Гнучкі технології спроектовані для роботи в умовах, що можуть швидко змінюватися. Разом з цим, вони націлені на такі характеристики, як інтерактивність, постійний зв'язок із замовниками й розподіл задач на дрібні етапи. Хоч такий підхід є гнучкішим, він вимагає постійної комунікації між проєктантами й замовником, що може спричинити складність у трудоемких проєктах.

«Scrum» – це вид «Agile», який орієнтовано на отримання конкретного результату упродовж короткого часового проміжку. Така технологія зосереджується на чіткому визначенню ролей у колективі, постійних нарадах і оглядах, які сприяють швидкому реагуванню на зміни.

Метод «Kanban» акцентує увагу на візуалізації процесу діяльності. «Дошки» «Kanban» забезпечують проєктантам розуміння задач, які визначені згідно статусу («Очікування», «Виконується», «Виконано»). Це забезпечує контроль робочого навантаження й ефективний розподіл ресурсів.

Інший інструментарій, такий як «Scrumban», «Lean», «Six Sigma», «Prince2», аналогічно застосовують відповідно з вимогами проєкту. Для прикладу, «Lean» акцентує увагу на підвищенні ефективності, а «Six Sigma» – на якість і уникнення помилок. Вибір методики керування проєктом визначається багатьма чинниками, у тому числі складністю завдань, розміром колективу, часом і фінансовими обмеженнями, а також характером діяльності стекхолдера. Жодна технологія не здатна бути універсальною і потребує адаптації до конкретного проєкту.

Вибір технології управління IT-проєктом забезпечує ефективність адміністрування діяльністю й дозволяє досягнути успішних результатів навіть у не простих умовах. Важливим є не лише обрання підходу, а й постійний аналіз його ефективності. За допомогою внесення потрібних змін можна досягнути максимальної продуктивності колективу.

### **2.1.1. Технологія «Водоспаду» («Waterfall»)**

Технологія «Waterfall» належить до однієї з найстарших і найтрадиційніших технологій керування проєктами. Вона була представлена В.Ройсом на початку 70-х років XX століття і була відповіддю на необхідність систематизації процесів розроблення ПЗ. Головними принципами цієї технології є наступне: «кожна фаза проєкту виконується строго після завершення попередньої фази, забезпечуючи чіткість і передбачуваність виконання завдань» [23].

Технологія «Waterfall» включає кілька головних циклів, які виконуються послідовно. На початку організовується збір та формалізація вимог до проєкту. Даний цикл надзвичайно важливий. Не чітке визначення вимог сприяє виникненню непорозумінь під час проєктування. Розроблення дизайну включає підготовку документації, планів, макету й моделі. Цей цикл завершується підготовкою відповідного плану, який є основою для впровадження.

Основні цикли у моделі «Waterfall» проводяться у суворо встановленому порядку. Будь-яку діяльність можна почати тільки після закінчення попереднього циклу. У концепції проєкту розроблення ПЗ ці цикли мають вигляд, який продемонстровано на рис. 2.1.

Під час завершення проєкту програмний код готується згідно з проєктною документацією. Тому, через детальне попереднє планування даний процес, як правило, проводиться швидко й без суттєвих правок. Після реалізації проєкту здійснюється перевірка, щоб пересвідчитись, що

спроектований продукт відповідає установленим вимогам та є безпомилковим.



Рис. 2.1. Цикли розроблення ПЗ у технології «Waterfall»

Останнім циклом є експлуатація і супровід, під час якого продукт надається користувачам, а розробники продовжують його підтримувати й оновлювати (за потреби).

Перевагою розглядуваної технології є простота та структурованість. Чіткість поділу на цикли сприяє розумінню завдань кожним учасником проєкту й дозволяє ефективно розподіляти робочий час. Лінійність технології сприяє високому ступеню контролю.

Кожен наступний цикл розпочинається лише після остаточного завершення попереднього циклу, що забезпечує стабільність розроблення

проєкту. Окрім цього, наявність розширеної документації, яка передбачена цією технологією, спрощує ознайомлення із проєктом нових учасників, що мінімізує залежність від унікальних навиків окремих фахівців.

Однак, цій технології притаманні і певні недоліки, головною з яких є її негнучкість. Зміни до вимог чи величини проєкту під час його розроблення створюють значні труднощі і призводять до додаткових витрат. Окрім цього, неможливість одержувати відгуки споживачів на проміжних циклах проєктування понижує адаптивність розробки до вимог стекхолдерів. Вказані характеристики визначають технологію «Waterfall» придатною для проєктів, у яких вимоги чіткі, а ризики – мінімальні.

У нинішніх умовах технологію «Waterfall», головно, використовують у сферах, у яких різні чинники легко передбачувати, наприклад у будівельній галузі, промисловості та розробленні проєктів із стислими строками. Не дивлячись на наявність новітніх технологій, «Waterfall» є прийнятним для проєктів, які потребують планування й дотримання структурованих підходів до керування процесами.

### **2.1.2. Технологія «Agile».**

Технологія «Agile» – це гнучкий й ітеративний спосіб керування проєктом, який спрямовано на ефективність адаптації до змін і дотримання планів швидкого виконання задач. Цю технологію офіційно представлено на початку 2001 року в публікації «Agile Manifesto», підготовленій фахівцями з інженерії програмного забезпечення. «Agile» винайдено як альтернативу традиційним технологіям, які є негнучкими у сучасних умовах розвитку менеджменту управління пореєктами. Головна суть «Agile» визначається тим, що будь-який проєкт подрібнюється на невеличкі, легко управляемі складові, які виконуються у короткі терміни, що забезпечує проєктну команду можливістю швидкого реагування на зміни та підтримання постійного контакту із стекхолдерами [24].

Головною особливістю «Agile» є акцентування уваги на результаті, який регулярно надається стекхолдеру і має для нього ключове значення. Проєкт у цій технології реалізується за допомогою короткострокових ітерацій (через «спринти»), що надає можливість проєктантам зосереджувати увагу на виконанні завдань й визначати результат після завершення певного етапу. Важливо, що дана технологія допускає зміну стратегій, включаючи останні цикли проєктування продукту. Це дозволяє бути їй ідеальною під час підготовки проєктів із значним ступенем невизначеності чи проєктів, вимоги до яких швидко змінюються. Наявність постійного зворотного зв'язку із стекхолдером забезпечує адаптацію продукту до актуальних потреб і забезпечує його високу якість.

У «Agile» є чотири головних принципи:

- 1) пріоритет розробників і їх взаємодії під час формалізованих процесів;
- 2) роботоздатність продукту замість перенавантаження інструкціями;
- 3) комунікація із стекхолдером замість неухильного дотримання умов співпраці;
- 4) сприйняття змін замість жорсткого дотримання початкового плану.

Такі принципи ґрунтуються на основних ідеях, які сформульовані в «Agile Manifesto» [25].

«Agile» притаманна низка позитивних особливостей, які сприяють її популяризації у галузі керування проєктами. Ця технологія дозволяє забезпечити швидкий та ефективний результат, позитивно сприймає інновації й підтримує гнучкість під час проєктування. Завдячуючи коротким ітераціям проєктанти здатні адаптовуватися до можливих змін, знизити ризик і підтримати значний ступінь взаєморозуміння між учасниками проєкту. Підтримання постійного зворотнього зв'язку сприяє зосередженню на дійсно

важливих для замовника аспектах і дозволяє проєктувати продукт, який відповідає його вимогам.

Однак, розглядувана технологія володіє і певними недоліками. Актуальний розрахунок фінансових затрат і термінів є складним у зв'язку із динамічністю процесу проєктування. Окрім цього, «Agile» потребує залучення потужного колективу програмістів, що є «тривожним» викликом під час підготовки великих ІТ-проєктів чи за умов співпраці з географічно розділеними колективами.

Зрештою, така характеристика технології «Agile», як гнучкість, надає їй універсальності, робить придатним інструментом для більшості сфер діяльності, у тому числі ІТ-галузі, маркетингу, фінанси тощо. Ця технологія особливо популярна у проєктах, які вимагають інноваційності, креативності й можливості швидкого реагування на зміни. «Agile» зосереджується на процесах удосконалення, що підвищує ефективність діяльності колективів і підготовці проєктів, які забезпечують сучасні технологічні ринкові вимоги.

### **2.1.3. Технологія «Scrum».**

«Scrum» – один із найпопулярніших ітеративних підходів у менеджменті проєктами, який забезпечує швидке реалізування завдань й адаптацію до можливих змін. Ця технологія орієнтована на короткотривалі ітерації, що отримали назву «спринти». Такі етапи, як правило, займають від одного до п'яти тижнів і вимагають виконання командою певного обсягу задач із зрозуміло означеною метою. Головною особливістю «Scrum» є те, що на початковому етапі задачі й пріоритети колективу розробників є постійними до його закінчення. Це забезпечує стабільну діяльність, сфокусовану на результат. У кінці певного спринту колектив розробників надає частково чи повністю підготовлений проєкт, який можна представити стейкхолдерам [26].

У технології «Scrum» головним аспектом є постійна ітерація, яка націлює колектив програмістів на постійний випуск результату, здійснення його оцінки й ефективності, на застосування потрібних корегувань. Така технологія знижує ризик, пов'язаний із довготривалими етапами роботи, який властивий для традиційних моделей менеджменту проєктами, і сприяє високій швидкості адаптування до змін. Використання технології «Scrum» забезпечує не лише швидку реалізацію проєкту, а й зниження фінансових затрат на його підготовку й впровадження. Цей аспект зумовив його популярність у колективах, які проєктують в умовах динамічних змін.

У цій технології не останню роль відведено регулярним нарадам, які забезпечують організацію робочого часу. Планування під час нарад сприяє доведенню до колективу завдань, які потрібно виконати, й провести їх розподіл між конкретними виконавцями. Постійні зустрічі дозволяють проводити обговорення результатів роботи, поточні робочі питання й можливі перепони. У кінці проєкту організовуються дві основні наради: представлення проєкту (колектив демонструє проєкт стекхолдерам) і ретроспектива, на якій проводиться аналіз сильних сторін і недоліків кінцевого етапу з метою підвищення ефективності діяльності у майбутньому.

«Scrum» характеризується наявністю унікальної структури ролей. Наприклад, Scrum-майстер – спеціаліст, що забезпечує координацію діяльності колективу, вирішення проблем і підтримання ефективності процесу проєктування. На противагу традиційному менеджеру проєкту, він не управляє командою, а сприяє максимальній продуктивності її діяльності. Наступним важливим аспектом є те, що замовник проєкту, яким визначено пріоритети задач, є відповідальним за підготовку беклогу проєкту – переліку функціональних вимог та задач.

«Scrum» притаманні п'ять основних аспектів: віданість, фокусування, відкритість, поважність, сміливість. Дані аспекти акцентують увагу на важливості взаємодії, довірі й прагненню отримати високий результат. Ця

технологія забезпечує розвиток креативного мислення й інноваційності, надаючи колективам розробників можливість застосувати різні підходи до вирішення проблем.

Як позитивний аспект слід зазначити, що технології «Scrum» притаманна простота реалізації, висока адаптованість до можливих змін у проєкті, у ній чітко означені процедури. Однак, дана технологія має й певні застереження щодо використання. Наприклад, вона не придатна для великих колективів чи проєктів, де встановлюються суворі вимоги до строків виконання, фінансування чи обсягу робіт. Самоврядування колективу може створити певну проблему у проєктах, де необхідний постійний контроль чи передбачуваність.

Технологію «Scrum» найліпше використовувати у невеличких, гнучких колективах, які розробляють проєкти в умовах динамічних змін і зі значним ступенем невизначеності. Ця технологія широко застосовується у галузі розроблення ПЗ, маркетингу, дизайнерських рішень тощо. Однак, у галузях із суворо визначеною вимогою до стабільності «Scrum», як правило, є не найліпшим вибором. За наявності таких випадків цю технологію використовують у сукупності з іншими методами управління. При цьому утворюються гібридні методи, у яких поєднуються переваги окремих технологій.

#### **2.1.4. Технології «Kanban» і «Lean».**

Технології «Kanban» і «Lean» є управлінськими підходами, які акцентуються на оптимізації технологічних процесів, підвищенні ефективності діяльності й забезпеченні високих цінностей для стейкхолдерів. Розглядувані технології розроблені під час проєктування технологічних процесів фірмою «Toyota», однак, закладені у їх основу принципи активно використовуються у різних галузях, у тому числі і в менеджменті ІТ-проєктами.

Головна ідея технології «Lean» полягає в економії фінансових затрат і акцентується на цінностях, важливих для стейкхолдерів. «Lean» сприяє постійному удосконаленню технологічних процесів за допомогою аналізування, експериментування та впровадження інноваційних підходів. Завдячуючи такому підходу розробники здатні одночасно досягти наступних важливих показників: зменшення фінансових затрат під час створення проєкту, удосконалення колективної діяльності, підвищення продуктивності.

«Kanban» своєю чергою акцентується на візуалізуванні технологічного процесу. «Kanban» (з японської «сигнальна карта») використовує «дошки» для онлайнного відображення стану виконання технологічного процесу. Сторінки на «дошці» відображають певні процеси, а їх перехід між стовпчиками вказує на прогрес реалізації. Також технологія використовує інші візуальні сигнали, наприклад, стежки для кластеризації задач, що сприяє координації діяльності.

Розглядані технології використовуються для оптимізації діяльності та ліквідації слабких ланок у технологічних процесах. Вони забезпечують швидку адаптацію до зміни технологічного процесу, що зменшує ризик його невиконання та поліпшує керування колективом. Технологія забезпечує можливість виявлення проблем під час робочого процесу на початкових стадіях і дозволяє вчасно вжити заходи для їх ліквідації.

До переваг «Kanban» і «Lean» відносять високу прозорість, ефективність і швидкість виконання задач. Ці технології надають можливість колективам ліпше організувати діяльність, знизити кількість невиконаних задач і забезпечити необхідну якість виконаної роботи. «Kanban», наприклад, зручний у використанні і придатний для організації діяльності колективів, які працюють над проєктами із циклами, які легко передбачити.

Однак, слід зазначити, що цим технологіям притаманні і певні недоліки. Для прикладу: невірне застосування візуальних «дошок» у «Kanban» здатне створити плутанину чи захаращеність інтерфейсу. «Lean», у

свою чергу, потребує глибокого усвідомлення технології роботи і готовності до можливих її змін.

Загальні особливості технологій «Kanban» і «Lean» описано в [27].

Обрання оптимальної технології управління IT-проектами є основним аспектом, що визначає успішність виконання задач і мети проєкту. Кожна із розглянутих технологій забезпечує унікальним набором інструментарію, який є ефективним за умов відповідності специфіки розроблювального проєкту. Через те, перед вибором певної технології слід проаналізувати певні фактори, які сприятимуть у адаптації підходів до вимог колективу, проєкту й очікуваного результату.

Важливо усвідомлювати, що обрана технологія управління інколи потребує деякої корекції під час управлінської діяльності. Потреба у такій корекції виникає внаслідок появи непередбачуваних ситуацій. Через те, керівникам необхідно бути готовими застосовувати інші, гібридні підходи й запроваджувати нові технології, які відповідатимуть потребі проєкту.

## **2.2. Програмний інструментарій IT-проектів**

До сучасних IT-проектів висувають багато вимог, головними з яких є висока ефективність, чіткість у плануванні й оперативному контролі за виконанням задач. З цією метою використовують спеціальний програмний інструментарій, який забезпечує організаційні заходи, контроль за прогресом й аналізування результату. Такий інструментарій є необхідним не тільки для офісних працівників, а й для тих, які виконують завдання віддалено, наприклад, за умов нинішнього переходу на онлайн-формат.

До одного із найважливіших інструментаріїв відносяться, так звані, таск-менеджери. Це відповідні програмні засоби, що забезпечують керівника проєкту можливістю ефективно здійснювати розподіл завдань, контроль їх виконання й відслідковувати загальний прогрес. Застосування такого інструменту надає менеджеру можливість не бути фізично присутнім біля

кожного члена колективу. Уся необхідна інформація по проєкту доступна на моніторі в онлайн форматі. Для онлайн-колективів таск-менеджер є надзвичайно важливим засобом, так як забезпечує створення єдиного простору для обміну даними.

Окрім таск-менеджерів, до програмного інструментарію керування проєктом можуть входити засоби, що охоплюють різносторонні чинники діяльності колективу. Зокрема, засоби планування й управління задачами сприяють формуванню діаграми Ганта, часових шкал чи візуальних «дошок» для візуалізації робочих процесів. Це забезпечує команду можливістю спостерігати за прогресом в онлайн-режимі й бачити, які задачі ще є невіршеними.

Одночасно слід відзначити важливу роль інструментарію для колективної роботи. Він здатен забезпечити єдиний простір для діяльності, обміну даними та дискусій. Цей інструментарій надзвичайно важливий під час роботи великих колективів чи проєктів, до яких можуть бути залучені позаштатні фахівці. У таких випадках він забезпечує підтримку прозорості діяльності та дозволяє зберігати усю інформацію в єдиному місці.

Інтегрування різного інструментарію також є надзвичайно важливим аспектом. Для прикладу, можливість інтеграції таск-менеджера, електронної пошти, програми-календаря, аналітичного сервісу спрощує технологічний процес і відкидає необхідність перемикання між різними сервісами. Цим досягається економія часу й підвищується ефективність діяльності колективу.

Також важливим аспектом ПЗ є функція контролю за часом. Такі засоби надають можливість проводити контроль і визначати терміни, які необхідні на виконання окремих задач, встановлювати можливі проблеми в адмініструванні діяльністю колективу. Це, зокрема, важливо не тільки для обчислення оплати праці, а й для підготовки наступних циклів проєкту.

Для менеджера проєкту важливим є можливість одержувати звіт в режимі онлайн. Завдячуючи цьому створюється можливість оперативного

реагування на затримку проєкту, оцінити поточну ситуацію й оперативно вирішити проблему. Окрім цього, інтегровані засоби економічного планування допомагають взаємодіяти із стейкхолдерами і контролювати фінанси.

Широко використовувані таск-менеджери, наприклад, «Jira», «Trello», «Asana» тощо, мають потужні функціональні можливості для керування проєктами. Обрання конкретного інструментарію визначається потребами колективу, величиною проєкту й наявністю фінансування. За їх допомогою можна не тільки вносити зміни у завдання, а й здійснювати аналіз продуктивності, покращувати робочий процес.

Отже, програмний інструментарій управління IT-проєктами є базовим аспектом ефективного адміністрування робочим процесом. Він надає можливість під час планування досягти гнучкості, прозорості та точності, що дозволяє колективам досягти потрібного результату навіть у непростих умовах. Запровадження цього інструментарію сприяє оптимізації технологічних процесів, заощадженню часу й ресурсів, а також продуктивнішій діяльності кожного учасника колективу.

Розглянемо деякі програмні інструменти управління IT-проєктами детальніше.

### **2.2.1. «Jira».**

«Jira» – це таск-менеджер, у якому інтегровано функціонал технологій «Kanban» і «Scrum». Такий підхід сприяв його популяризації у галузі управління проєктами. Завдячуючи вказаній інтеграції «Jira» забезпечує ефективну координацію діяльності великих колективів.

Одним із позитивних чинників цього інструменту є наявність можливості зворотного зв'язку під час адміністрування робочого процесу. Наприклад, коли проєктувальник особисто планує свої завдання, а менеджер, якого призначено відповідальним за контролем процесу, відслідковує його і

здійснює відповідне коригування. Окрім цього, у програмі є можливість здійснювати оцінку діяльності, що дозволяє відслідковувати ефективність роботи кожного члена команди.

Розглядуваний додаток доступний як у мобільній версії, так і для стаціонарного комп'ютерного обладнання, що дозволяє здійснювати контроль за діяльністю з будь-якого пристрою. «Jira» інтегрується із різними додатками і хмарними технологіями, що сприяє простоті синхронізації інформації між різним інструментарієм. Основою цього додатку є система тикетів, що призначаються для реєстрування і здійснення запитів. Така технологія сприяє популяризації цієї платформи у IT-галузі, де її застосовують для контролю за завданнями, які пов'язані із проектуванням ПЗ.

Головною із особливостей є значний ступінь кастомізації. Завдячуючи наявним можливостям API й значній бібліотеці плагінів, додаток надає можливість налагоджувати процес діяльності згідно з потребами колективу. Для прикладу, можна у автоматичному режимі відправляти нагадування у спільний чат чи приховати неактуальну інформацію щодо завдань. Однак, таке налагодження потребує відповідних технологічних навиків.

Слід відзначити, що даний інструмент добре адаптується для великих колективів, так як його можливості дозволяють створити структуру для декількох рівнів проекту й забезпечити необхідну взаємодію між підрозділами. Не дивлячись на наявні переваги, запровадження «Jira» може зайняти багато часу, оскільки існує потреба у тривалому навчанні, що є певною проблемою для працівників, які не мають практичних навиків у роботі з подібними сервісами.

До головних позитивних особливостей «Jira» відносять великий функціонал для кастомізації. Інтеграція функціоналу технологій «Kanban» і «Scrum» сприяє зручному плануванню спринтів. Головними недоліками додатку є складність в освоєнні, не високий рівень автоматизації, постійні оновлення інтерфейсу.

### 2.2.2. «Redmine».

«Redmine» – інструмент для управління ІТ-проектами із відкритим кодом, який дозволяє забезпечити гнучкість під час організації діяльності та відстежувати процеси проектування. Даний додаток побачив світ на початку ХХІ ст. і призначений для організації діяльності технічних команд, наприклад, підрозділів проектування, яким необхідна функція контролю за помилками.

Головною особливістю цього інструменту є гнучкість у розподілі ролей і обов'язків. У кожному проєкті чи завданні є можливість визначити кількісний склад колективу й підготувати для них відповідний функціонал в залежності від потреби. Такий підхід дозволяє організувати деревоподібну структуру, у якій кожному працівнику колективу доступні ті засоби, які необхідні йому для діяльності. У «Redmine» підтримується функція ведення кількох підпроектів в одній системі, що є особливо корисним під час діяльності великого колективу, який розробляє різні напрямки проєкту одночасно.

Дана система є багатофункціональною і включає: функцію виявлення помилок, контроль часу, календар, інструментарій сумісної діяльності із кодом. У додатку передбачені зручний функціонал фільтрування та впорядкування карточок, за допомогою яких спрощуються процеси навігації і пошуку даних. «Redmine» підтримує багатомовність, що обумовило його популярність у міжнародних проєктах. Наступним позитивним фактором є форум, який забезпечує ефективність під час обговорення задач.

Не дивлячись на безкоштовність сервісу, йому також притаманні і певні недоліки. Зокрема, застарілість інтерфейсу, окремий функціонал приховано. Окрім названого у базовій версії відсутній інструментарій для контролю активності проєктувальників і контролю за часом виконання завдань.

Отже, «Redmine» – це надійний та функціональний сервіс для організації діяльності колективів, яким необхідно забезпечити гнучкість у керуванні ІТ-проектами.

### 2.2.3. «Trello».

«Trello» – це один із найпопулярніших засобів керування ІТ-проєктами, який орієнтований на організацію діяльності невеликих колективів. У ньому закладені базові функції, які потрібні для організації діяльності й її координації. Основою додатку є технологія «Kanban», що забезпечує відстежування діяльності за допомогою інтерактивних «дошок». За їх допомогою організуються карточки, які переміщуються між стовпчиками, тим самим відзначаючи етапи виконаних завдань.

Даний сервіс відрізняється від розглянутих вище інструментів простотою у використанні і впровадженні. Наявність інтуїтивно зрозумілого інтерфейсу і можливість візуалізації організації задач надають йому зручності у використанні навіть для працівників, у яких відсутній досвід роботи із таким інструментарієм. Через це «Trello» є зручним сервісом для колективів, що виконують прості чи лінійні проєкти, у яких не передбачено складну аналітику чи жорсткий облік часу.

Додаток добре інтегрується із хмарними сервісами, наприклад, OneDrive, що забезпечує простоту під час обміну інформацією і синхронізації даних.

Сильною стороною «Trello» є зручність у роботі: усі базові процеси (створення, редагування, пересування карточок) здійснюються кількома кліками мишки. До завдань можна застосовувати фільтри (за кольором), що спрощує пошук необхідних даних. Базова версія доступна безплатно, що робить цей додаток популярним для використання у невеликих колективах із обмеженим фінансуванням. Однак, сервісу притаманне обмеження у кастомізації, відсутня підтримка аналітичних функцій та неможливість ведення обліку часу (у базовій версії). Також він не призначений для організації роботи у великих проєктах, так як інтерфейс здатен швидко перевантажитися, а задачі із різних проєктів не візуалізуються на одному моніторі.

Таким чином, під час аналізу інструментарію для підготовки ІТ-проектів встановлено, що вибір технології обумовлюється багатьма факторами: розміром колективу, складністю та специфікою завдань, потребою у досягненні відповідного ступеня адаптивності.

Проаналізовані технології «Agile» і «Scrum» ефективні під час виконання гнучких і динамічних завдань, а «Waterfall» ефективніша під час управління проектами з чіткими вимогами.

Аналіз інструментарію для управління ІТ-проектами показав, що сервісу «Trello» притаманна простота у використанні і застосуванні під час управління ІТ-проектами у невеликих колективах із нескладними завданнями. «Jira» має значний функціонал і краще підходить для комплексних завдань та технічних колективів, наприклад, для проєктувальників. «Redmine» володіє широким функціоналом, однак його використання обумовлюється необхідністю відповідного навчання й адаптації.

Проведений аналіз дозволив сформулювати чітке усвідомлення технологій й інструментарію управління ІТ-проектами, що є базовим аспектом під час їх обрання та використання на практиці.

### Розділ 3.

## РОЗРОБЛЕННЯ МОДУЛЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ ІТ-ПРОЄКТІВ

### 3.1. Вебдодаток як елемент ІС

ІС займають провідне місце в адмініструванні будь-якої діяльності, забезпечуючи збереження, опрацювання й обмін інформаційними ресурсами з метою прийняття обґрунтованих управлінських рішень. Особливе місце у цьому процесі займають вебзастосунки, які є у вільному доступі мережі інтернет і забезпечують додатковим функціоналом користувачів.

Такі вебзастосунки легко інтегруються із технологічними комплексами, що використовуються під час автоматизації процесів керування інформаційними ресурсами. Вони включають апаратне й програмне забезпечення, БД, мережеві ресурси та користувачів, що працюють із ІС. До вебзастосунків можна підключатися з будь-якого дивайса та одержати допомогу в управлінні за допомогою хмарних технологій.

До головних позитивних особливостей використання вебзастосунків відносять наявність інтуїтивно зрозумілого інтерфейсу, який легко може використовуватися навіть недосвідченими користувачами. Таким чином, спільна діяльність в режимі онлайн стає значно простішою. Це дуже важливо під час діяльності розподілених колективів. Централізація оновлень забезпечує усіх користувачів останньою версією без потреб ручного встановлення змін.

Застосовуючи такий інструмент, як вебзастосунок можна звести до мінімуму дублювання інформації, автоматизувати процеси її опрацювання й оптимізувати процеси управління. Завдячуючи гнучкості та доступності дані рішення набули важливого значення у сучасному менеджменті, дозволяючи вирішити проблеми підвищення ефективності діяльності компаній.

### 3.2. Основні переваги спроектованого модуля

Спроекований модуль створює якісно новий напрям в адмініструванні діяльністю колективів під час реалізації ІТ-проектів на противагу розглянутим у розділі 2 популярним інструментам, наприклад, таким як «Jira» та «Trello».

Головною позитивною особливістю розробленого вебдодатку є його висока гнучкість і кастомізація, що забезпечує адаптацію інструменту під визначені командою потреби власного проекту без застосування тих чи інших обмежень, які часто виникають під час використання традиційних рішень.

Також важливим позитивним чинником є можливість проектувати будь-яку кількість БД у визначеному командою форматі. Це забезпечує їх можливість самостійно визначитись із структурою керування діяльністю, обирати найзручніші параметри, поля, категорії. У згадуваних вище інструментах («Jira», «Trello») ці категорії задаються наперед визначеними шаблонами та структурами. Таким чином, спроектований модуль забезпечує реалізацію індивідуального підходу до адміністрування діяльністю.

Також вебдодаток надає широкий функціонал для інтегрування із зовнішніми елементами за допомогою API. Це надає користувачам можливість під'єднувати зовнішні служби, БД чи інший інструментарій. Таки чином можна об'єднати до однієї системи усі потрібні інструменти для створення проекту. Для прикладу, можна об'єднати аналітичні сервіси, інструмент контролю за часом, засоби для проектування дизайну.

Ще одним особливим позитивним аспектом є можливість організації підтримки різних медіаформатів. Модуль забезпечує можливість додавати рисунки, відео-, звукові або інші файли, які є важливими у діяльності команди. Окрім цього, документацію і файли можна зберігати у ІС із можливістю розсилання між учасниками проекту, що забезпечує спрощення процесу обміну інформаційними ресурсами та доступу до них.

Додаток також містить інструментарій колективної діяльності в режимі онлайн. Членам колективу надається можливість одночасної роботи з технічною документацією, БД й іншими додатками, оновлювати їх і здійснювати обмін даними без потреби використання сторонніх сервісів. Це забезпечує створення умов для ефективної діяльності, що є важливим аспектом у сучасних системах управління ІТ-проєктами.

Також перевагою є застосування спрощеного підходу до менеджменту проєктами. Значна кількість сучасного інструментарію має складний інтерфейс і потребує тривалого терміну для адаптування, а пропонований вебдодаток спроектовано за принципом максимальної простоти. Це забезпечує користувачів можливістю швидкого освоєння функціоналу і дозволяє сконцентруватися на вирішенні завдань, а не на вивченні вебдодатка.

З огляду на кастомізацію розроблений модуль забезпечує необмеженими можливостями персоналізації процесів діяльності. Учасники команди можуть адаптовувати під себе кожен цикл діяльності у системі. Через те додаток можна використати як під час роботи невеликих колективів, що вирішують локальні задачі, так під час діяльності великих фірм, що мають багаторівневу структуру менеджменту.

Таким чином, спроектований вебдодаток інтегрує у собі гнучкі інтеграційні можливості й зручність у використанні, що відсутнє у багатьох традиційних рішеннях. Універсальність, наявність значного функціоналу і акцентування на адаптації до потреб окремого користувача роблять ІС необхідним засобом в управлінні ІТ-проєктами. Такий підхід відповідає нинішній ситуації в організації ІТ-діяльності.

### **3.3. Проєктування БД для модуля ІС управління ІТ-проєктами**

Проєктування зручної БД є одним з головних етапів під час розроблення модуля ІС управління ІТ-проєктами. У даному підрозділі

опишемо процеси розроблення й використання БД, яка забезпечить надійність збереження та швидкий доступ до документації, завдань і проєктів.

На початковому етапі нашої діяльності слід означити головний функціонал бази даних. Згідно задуму він має здійснювати підтримку зберігання відомостей про членів команди з іменами, електронними адресами та хешованими паролями з метою забезпечення безпеки персональних даних відповідно до існуючого законодавства (Закону України «Про захист персональних даних» [28]). Окрім цього, у БД має розміщуватися як текстова, так і мультимедійна інформація, яку можна впорядковувати за видами з метою зручності пошуку та категоріювання. Особливе значення надано керуванню задачами та проєктами. З цією метою запроваджено структурування контролю термінів виконання, статусу й зв'язку між задачами і проєктами. БД також має підтримувати функцію співпраці, забезпечуючи користувачів доступом до проєктів, гнучкість контролю над правами редагування. При цьому забезпечується історія змін, яка надає можливість за необхідності відстежити зміни й повернутися до попередньо збережених даних.

Для розроблення БД обрано реляційну СУБД «PostgreSQL». Вона забезпечує вимоги щодо продуктивності, надійності й підтримки масштабованості. Ця СУБД забезпечує також ефективне опрацювання складних запитів і транзакцій. Це є важливим функціоналом під час роботи зі значними обсягами інформаційних ресурсів. Одночасно вона підтримує можливість використання різноманітних розширень, що сприяє формуванню потрібного функціоналу.

Проєктування БД передбачає підготовку базових таблиць із чітким зазначенням сутностей з метою забезпечення інтегрованої структури для організації та управління інформаційними ресурсами. Даний підхід дозволяє забезпечити надійність ІС, отримати необхідний функціонал для ефективного

керування ІТ-проектом. На рис. 3.1 наведено організаційну структуру спроектованої БД.

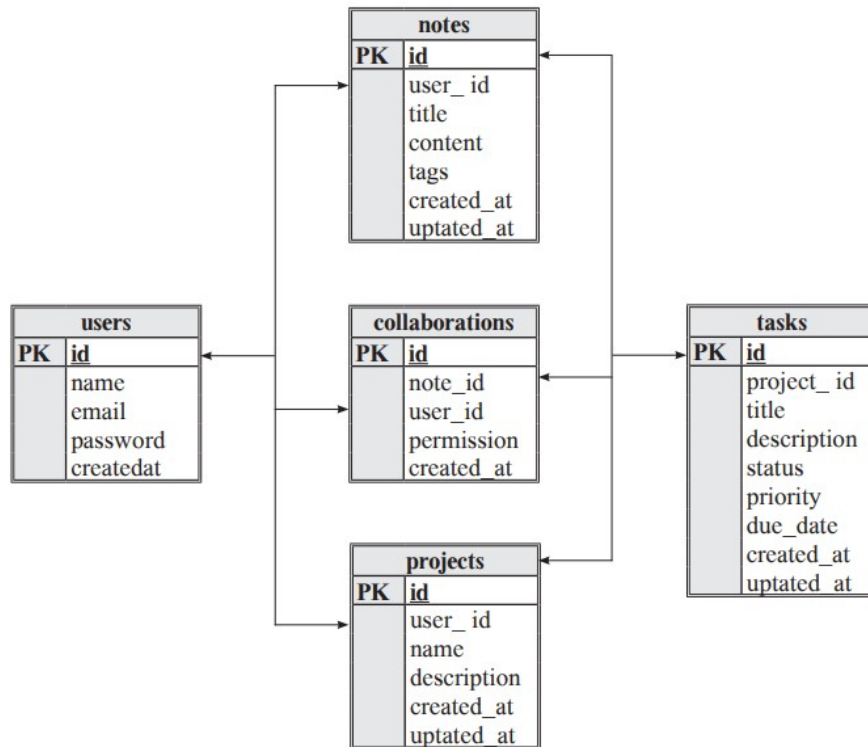


Рис. 3.1 Структура БД для модуля управління ІТ-проектами

Структурно БД має зв'язки між головними таблицями, що відображає логічність взаємозв'язку користувачів, проектів, завдань і нотатків. Наприклад, користувач може мати багато нотаток, однак, кожна з них належатиме тільки одному користувачу. Такий зв'язок реалізовано за допомогою зв'язку один–до багатьох між таблицями «users» та «notes».

Звідси, кожен користувач має можливість підготувати значну кількість проектів, однак кожен з них пов'язується із визначеним членом команди, що також представляється зв'язком один–до багатьох між таблицями «users» і «projects».

Для управління задачами створюється зв'язок між таблицями «projects» і «tasks». Проект може мати багато задач, однак кожна з них відноситься до одного проекту. Тут також застосовано зв'язок один–до багатьох.

Стосовно нотаток і взаємостосунків додаток забезпечує гнучкість і ефективні засоби редагування. Певну нотатку можна передивлятися чи редагувати кілька разів. Усі користувачі можуть мати доступ до визначеної кількості нотаток. Це реалізується зв'язком багато–до багатьох між «users» та «notes» через сутності «collaborations».

Визначившись із структурою БД потрібно спроектувати необхідні таблиці та сутності. Під час цього циклу використовуються SQL-запити для встановлення сутностей, які дозволяють досягти ефективності роботи у системі. Лістинг 3.1 демонструє приклад SQL-запиту для проектування таблиці.

Лістинг 3.1.

```
CREATE TABLE Users (
  id SERIAL PRIMARY KEY,
  name VARCHAR(100),
  email VARCHAR(100) UNIQUE,
  password VARCHAR(255),
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);|
```

Проектування БД для додатку управління IT-проектами є важливим циклом, завдяки якому досягається надійність та ефективність роботи з інформаційними ресурсами.

### 3.4. Моделювання додатку ІС управління IT-проектами

Процес моделювання ІС для вебзастосунку керування IT-проектами потребує проведення детального опису архітектурного рішення і означення головного функціоналу. Даний етап включає огляд сценаріїв застосування додатку й аналізування взаємозв'язку між компонентами ІС з метою досягнення необхідної ефективності й простоти застосування.

Архітектурно ІС розділена на два компоненти: фронтенд і серверна складова. Вони активно взаємодіють із БД. Популярним фреймворком «Next.js», досягається підтримка візуалізації зі сторони сервера, тому його

застосовуємо для проектування інтерфейсу. Дана методологія забезпечує підвищення швидкодії ІС і сприяє покращенню пошукової оптимізації. Для досягнення надійності збереження інформаційних ресурсів й ефективності опрацювання великих масивів інформації як сховище даних оберемо реляційну БД «PostgreSQL».

Модель даної БД включає кілька головних таблиць: «users», «notes», «projects», «tasks» і «collaborations». Кожна із цих таблиць забезпечує досягнення унікального функціоналу у структурі ІС. «Users» є головним компонентом, за допомогою якого забезпечується доступність до інших елементів, а «notes» є засобом для збереження текстових даних про проєкт. «Projects» застосовується для організації «tasks» і «notes». «Tasks» визначає певну дію чи мету, яку слід досягти у рамках проєкту. За допомогою «collaborations» забезпечується доступ до інформаційних ресурсів і співпраця над проєктом.

Між сутностями в ІС створюються відповідні зв'язки, які встановлюють їх взаємодію. Проектувальники можуть створити й управляти різними завданнями одночасно. Кожен проєкт, як правило, містить значну кількість завдань, що забезпечує деталізацію робочого процесу. Декілька розробників ІТ-проєкту можуть проводити редагування нотаток одночасно з використанням модулів спільної діяльності, які забезпечують обмін відомостями та синхронізують їх.

Розроблена модель організовує гнучкість та практичність структури ІС, яка просто адаптується до вимог адміністратора ІТ-проєкту, що сприяє ефективній організації роботи над проєктом. Для кращого розуміння функціональності ІС проведемо розгляд деяких сценаріїв її практичного використання.

За допомогою реєстрації створюється обліковий запис, у якому користувач вводить власне ім'я, e-mail і пароль. Для діючих користувачів необхідно лише ввести ці дані для входу. Відомості, уведені у відповідному

вікні інтерфейсу, надходять за допомогою API до сервера. За умов успішної автентифікації працівник одержує доступ до ІС.

Робота з нотатками полягає у їх створенні та редагуванні. З цією метою вводяться заголовки і зміст нотатки. Вони через API надсилаються до серверу, опрацьовуються й зберігаються у БД. Змінені відомості направляються через інтерфейс, забезпечуючи його оновлення.

Новий проєкт створюється за допомогою введення назви й опису та додається до ІС. Керування проєктом передбачає включення до нього задач, зміни їх статусів і пріоритетів. Члени команди взаємодіють між собою через інтерфейс, а серверною частиною опрацьовуються запити та оновлюються дані у БД з метою їх синхронізації з інтерфейсом.

Використання функціоналу спільної діяльності забезпечує можливість обмінюватися нотатками через надання прав перегляду або редагування будь-яким членом команди. Відомості пересилаються до сервера, де організовується запис у таблиці співпраці. За допомогою «WebSockets» чи іншими засобами здійснюється синхронізація в режимі онлайн.

Наведені сценарії показують зручність використання ІС щодо підтримання базових операцій і складних завдань, які стосуються спільної діяльності. Обрання «PostgreSQL» та «Next.js» сприяє ефективній і надійній роботі ІС. Реалізування функцій реєстрації, формування проєкту і взаємодії з нотатками формує універсальний інструментарій для організації відповідних завдань.

Графічну модель ІС для управління ІТ-проєктами показано на рис. 3.2.

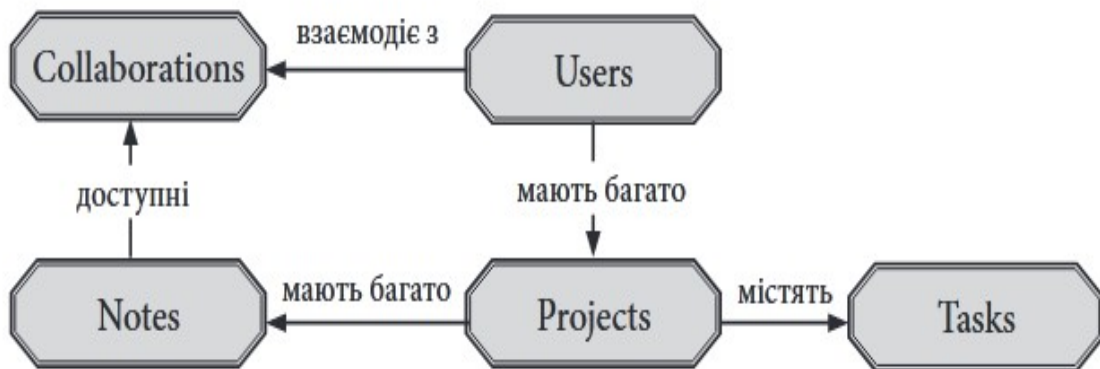


Рис 3.2. Моделювання ІС управління ІТ-проєктами

Дана схема відображає загальні сутності й їх взаємозв'язок. За її допомогою легко зрозуміти структуру БД і взаємозв'язок між різними елементами цієї системи.

### 3.5 Архітектура модуля ІС управління ІТ-проєктами

Інструмент «Next.js» є популярним фреймворком у системі «React» і фактично став стандартом під час підготовки програм «React». Його популярність обумовлюється наявністю широкого функціоналу, у тому числі інтеграцією «TypeScript», підтриманням додатків CSS (Cascading Style Sheets), оптимізацією візуалізації тощо. Даний функціонал дозволяє проєктувати сучасне ПЗ.

Першим етапом під час проєктування у програмі «Next.js» є її активування з використанням команди, яка забезпечує основу підтримки «TypeScript»:

```
npx create-next-app --typescript TrackIT
```

«TypeScript» забезпечує безпеку типів і легкість розроблення масштабованого ПЗ. Для запуску проєкту необхідно виконати команду сервера на «localhost:3000»:

```
npm run start
```

Під час проектування слід пересвідчитися, що версія «Node.js» і пакетний менеджер узгоджено. Це забезпечується файлом `.nvmrc`, яким визначається версія «Node.js». Користувачі можуть уникнути проблем задавши потрібні версії «Node.js» та `npm` файлом `package.json`.

Після базового конфігурування рекомендовано вказати ІС за контролем версій, наприклад, `Git`. Також необхідно додати файл `.gitignore` для запобігання додавання конфіденційних даних до БД.

Для упорядкування коду у програмі «Next.js» створюється структура папок, що містять такі каталоги, як `App`, `Components`, `Utils`, `Hooks`, у яких визначаються, відповідно: маршрутизація, елементи «React», бібліотеки й утиліти, хуки.

Така побудова забезпечує збереження коду й сприяє його масштабуванню під час зростання розроблювального ПЗ. Структурно програму відображено у лістингу 3.2.

Лістинг 3.2.

```
src >
  > app
  > components
  > utils
  > hooks
```

«Next.js»-додаток містить наступні папки «`app`», «`components`», «`utils`», «`hooks`». Така структура допомагає організувати кодування й сприяє масштабуванню ІТ-проєкту. За допомогою такого підходу забезпечується використання серверних і клієнтських компонентів. Серверні – забезпечуються сервером, що впливає на ефективність і швидкодію. Клієнтські – забезпечують інтерактивні функції та вказуються за допомогою директиви «`use client`» на початку файла. У каталозі «`app`» додаються файли «`layout.tsx`», якими досягається загальне макетування усіх сторінок у визначеній директорії чи підпорядкованих папка, що забезпечує єдине структурування вебдодатку.

Представлена структура забезпечує зручне масштабування вебдодатку. Додавання нових сторінок, компонентів, маршрутів дає можливість спростити підтримку, що покращує продуктивність модуля ІС. Структурно каталоги «app»-роутингу показано на рис. 3.3.

Вебдодаток включає функціональні елементи, які формують загальні властивості і роботу модуля. У директорії «components» розташовуються елементи, які застосовуються багаторазово.

Директорія «hooks» містить хуки, тобто функціонал, який можна використати під час різних циклів розроблення програм.

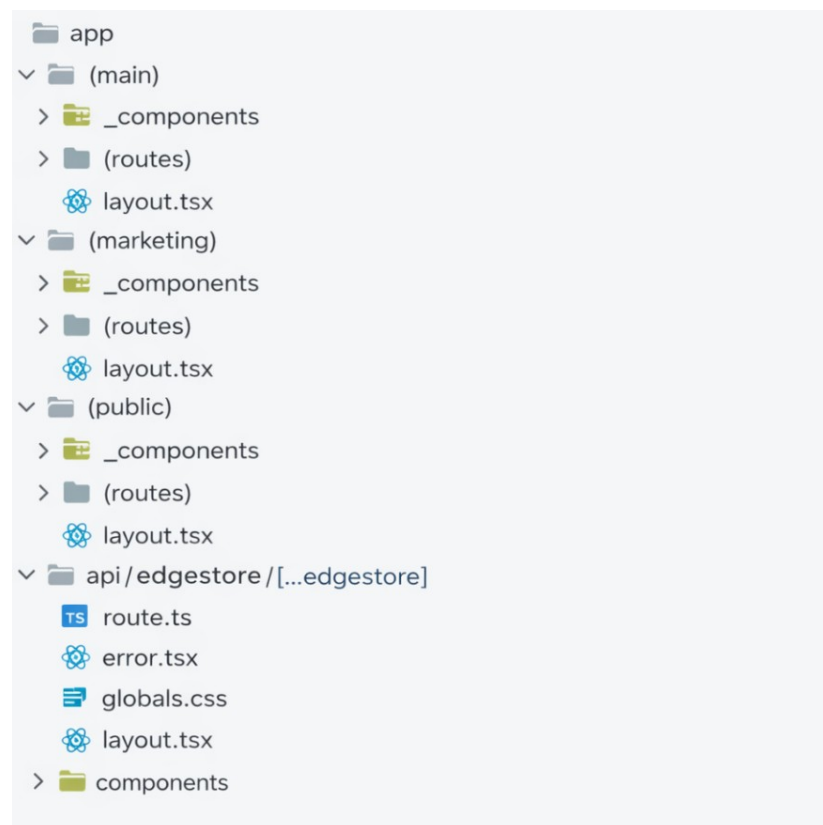


Рис. 3.3. Структурування маршрутизації

У директорії «utils» розміщують додатковий функціонал, який використовується для перетворення інформаційних ресурсів, що дозволяє покращити та спростити роботу із опрацювання інформації у вебдодатку.

Наведені рекомендації спрощують фундаментальні засади для розроблення вебдодатків.

### 3.6. Розроблення модуля управління ІТ-проектами

Початковим елементом будь-якого вебресурсу є його маркетингова сторінка. За її допомогою забезпечуються суто рекламні заходи. Її головною метою є надання зрозумілого опису функціоналу платформи та запросити користувачів до реєстрування чи авторизування. Інтерфейс маркетингової вебсторінки спроектованого додатку показано на рис. 3.4.

На даній вебсторінці розміщено засоби для реєстрування нових користувачів чи входження до додатку. Здійснити вхід можна також використовуючи акаунт Google (рис. 3.5).

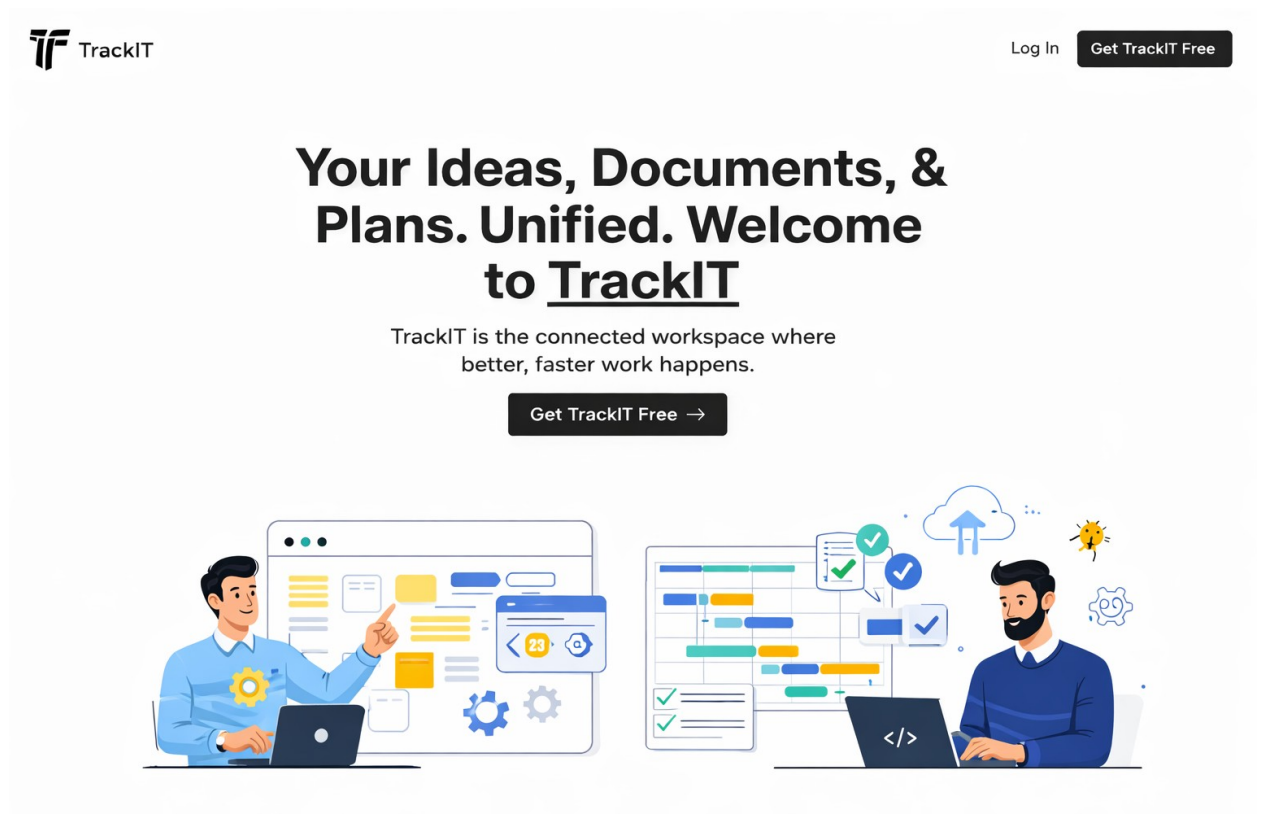
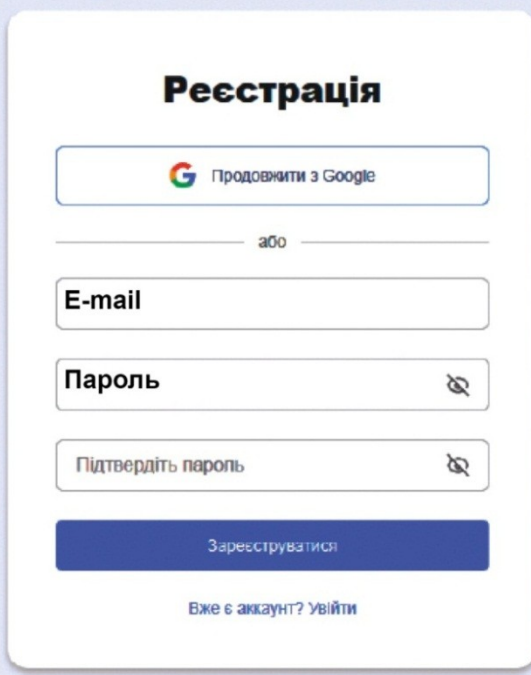


Рис. 3.4. Маркетингова сторінка вебдодатку



The image shows a registration form titled "Реєстрація" (Registration). At the top, there is a button labeled "Продовжити з Google" (Continue with Google) with the Google logo. Below this, there is a separator line with the word "або" (or) in the center. The form contains three input fields: "E-mail", "Пароль" (Password) with a visibility toggle icon, and "Підтвердіть пароль" (Confirm password) also with a visibility toggle icon. At the bottom of the form is a blue button labeled "Зареєструватися" (Register). Below the button, there is a link that says "Вже є акаунт? увійти" (Already have an account? Log in).

Рис. 3.5. Сторінка авторизації

За умови вдалого входу чи реєстрації в ІС у автоматичному режимі здійснюється перенаправлення на головну вебсторінку. Інтерфейс цієї вебсторінки показано на рис. 3.6.

Представлена вебсторінка є основою адміністрування ІТ-проєктами. У ній доступний функціонал, який необхідний для проєктування нових завдань. Із аналізу рис. 3.6. бачимо, що її графічний інтерфейс є достатньо зрозумілим. Будь-який проєкт має свою назву, ярлик й обкладинку. Такі іконки надають можливість просто обирати потрібний об'єкт із списку.

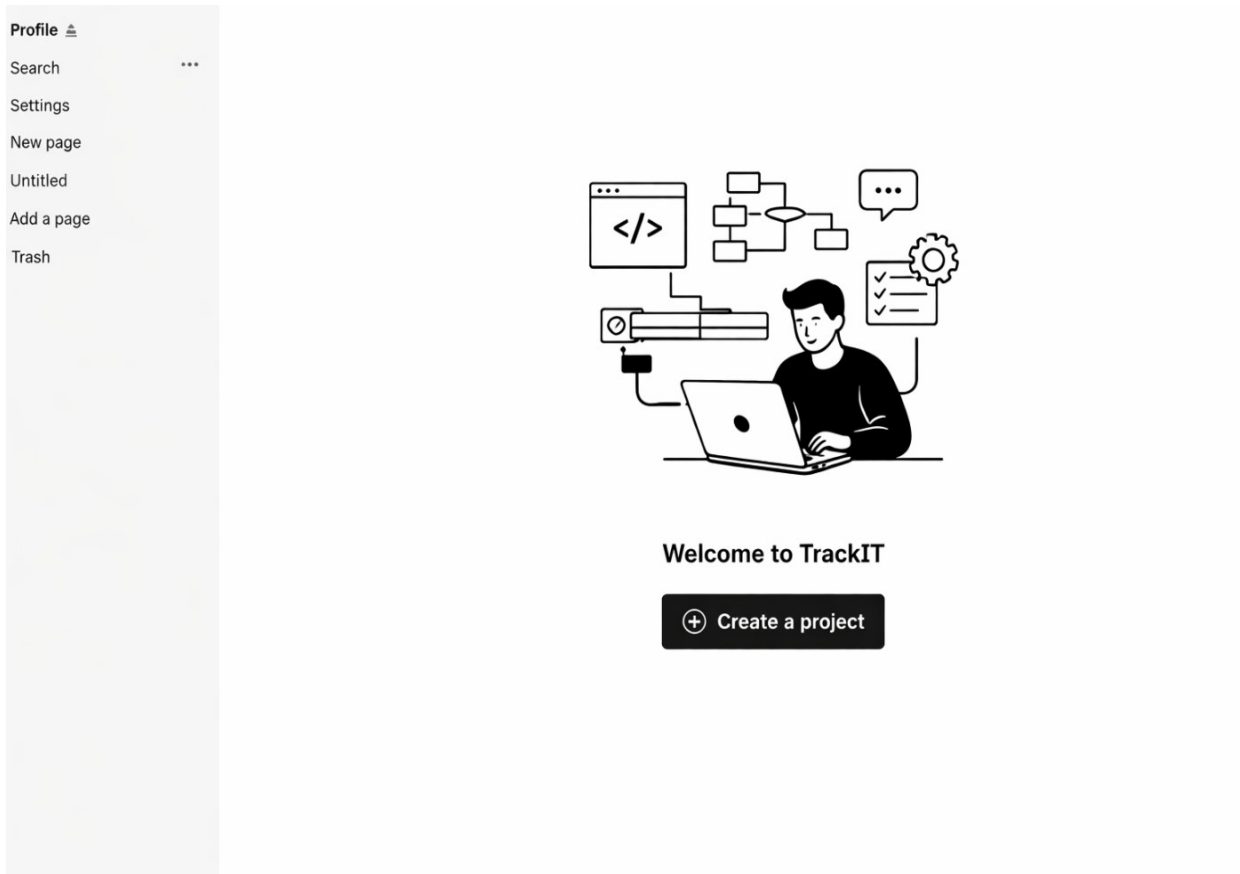


Рис. 3.6. Головна вебсторінка

Проектування нового завдання здійснюється кількома кліками мишки. Додавши основні елементи можна розпочинати діяльність із новим проектом. Інструментарій управління ІТ-проектами досить простий у застосуванні та охоплює як вбудований функціонал, так і можливість інтегрування інших вебсервісів, що забезпечує гнучкість додатку та можливість його переналаштування під потреби проекту.

Головна вебсторінка – це інтерфейс, у якому здійснюється управління ІТ-проектом. Її організацію максимально спрощено з метою ефективного та комфортного використання. Основним функціоналом на вебсторінці є:

- встановлення налаштувань проекту;
- менеджмент проектом з використанням команд;
- інтегрування із зовнішніми ресурсами;
- надання доступу для спільної діяльності.

Інтерфейс вебсторінки ІТ-проекту показано на рис. 3.7. У ньому є бокова стрічка, яка дозволяє здійснювати спрощений доступ до головного функціоналу. Дана стрічка має наступні команди:

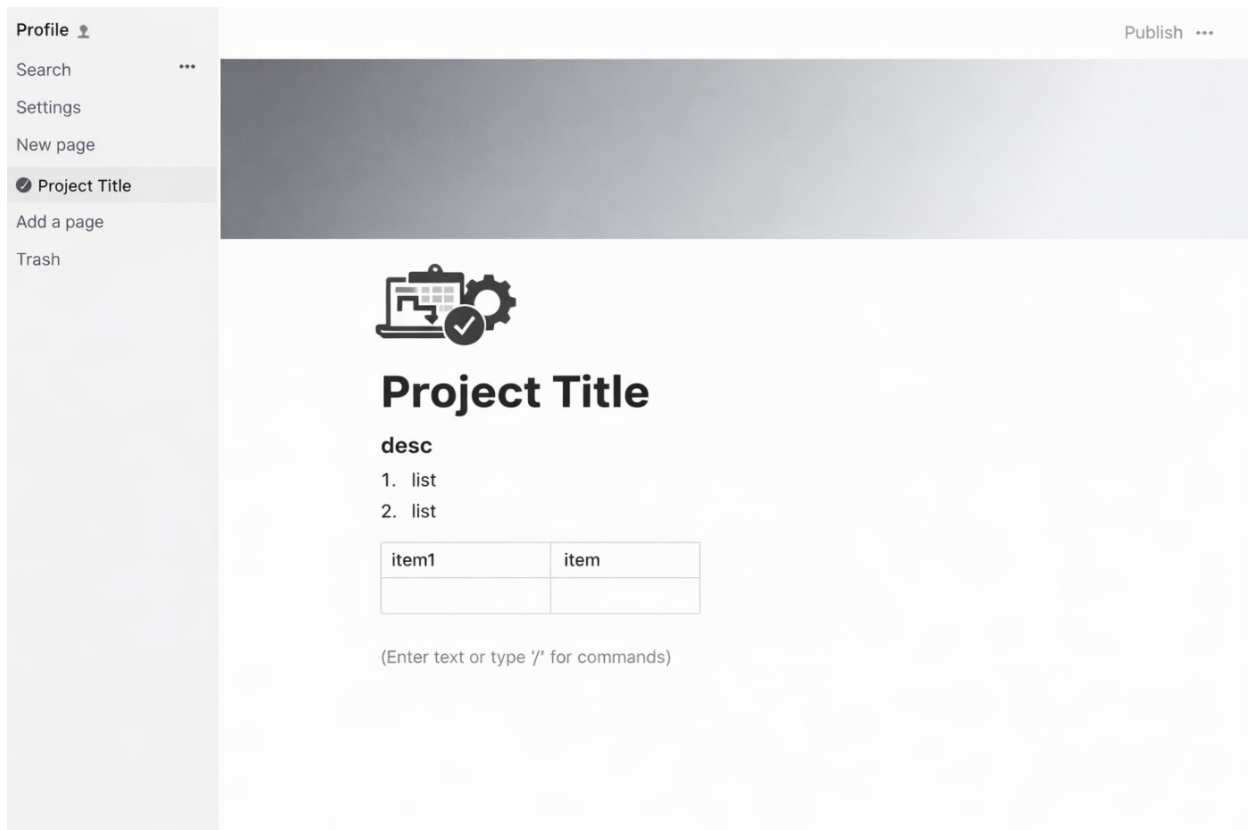


Рис. 3.7. Вебсторінка проекту

- профіль користувача – огляд облікових даних про користувача й вихід з додатку;
- пошук – функціонал, який забезпечує пошук за усіма проектами;
- налагодження – слугує для зміни вигляду інтерфейсу;
- новий проект – створення проекту;
- перелік проектів – візуалізація усіх створених завдань;
- смітник – видалення завдань.

Інтерфейс вебдодатку орієнтовано на мінімалізм. Уся дизайнерська структура вебсторінки продумана так, щоб не відволікати користувача зайвими відомостями. Таким чином, структурно розроблений модуль забезпечує користувачу доступ до усього необхідного інструментарію для

зручного керування ІТ-проєктом. Зручність та доступність головного функціоналу й простота у налаштуванні роблять розроблений модуль необхідним інструментом для адміністрування ІТ-проєктів.

Отже, у даному розділі представлено розроблення модуля ІС управління ІТ-проєктами: від обрання інструментарію до проєктування архітектурного рішення й БД. Наведені результати є надійною базою для подальшого інтегрування нового функціоналу й удосконалення інформаційної системи.

## ВИСНОВКИ

У кваліфікаційній роботі спроектовано та реалізовано модуль ІС, який призначений для підтримання процесів управління ІТ-проектами, що забезпечує ефективну організацію роботи колективу, управління завданнями та контроль виконання проектних робіт.

За результатами розробки виконані такі завдання:

- проаналізовано особливості управління ІТ-проектами та визначено основні вимоги до ІС підтримки таких процесів;
- досліджено існуючі програмні рішення та підходи до автоматизації управління ІТ-проектами та проведено їх порівняння;
- розроблено структурну й функціональну модель модуля ІС;
- реалізовано програмний модуль підтримки ІТ-проектів із використанням сучасних засобів та технологій розроблення ПЗ;
- проведено тестування й оцінку ефективності роботи розробленого модуля.

У результаті виконання роботи розроблені підходи до проектування модуля ІС підтримки ІТ-проектів, який забезпечує інтегроване управління завданнями, учасниками проекту та процесами контролю виконання робіт у межах єдиного інформаційного середовища. У рамках дослідження уточнено функціональні вимоги до програмного модуля підтримки ІТ-проектів з урахуванням особливостей організації командної роботи під час розроблення програмного забезпечення. Запропоновано структуру програмного модуля та модель організації взаємодії між його основними компонентами, що забезпечує ефективніше опрацювання інформації про завдання, ролі користувачів і стан виконання проекту.

Отримані результати розширюють підходи до створення модульних ІС управління ІТ-проектами та можуть бути використані під час проектування програмних засобів підтримки проектної діяльності.

Розроблений модуль дозволяє автоматизувати основні процеси організації проєктної діяльності, зокрема формування структури проєкту, управління завданнями, розподіл ролей між учасниками команди та контроль виконання робіт.

Використання такого програмного рішення сприяє підвищенню ефективності роботи команд розробників ПЗ, покращенню координації між учасниками проєкту та забезпеченню оперативного доступу до актуальної інформації про стан виконання завдань. Результати дослідження можуть бути використані під час розроблення та впровадження інформаційних систем підтримки ІТ-проєктів у навчальних, наукових або виробничих середовищах.

За результатами виконання кваліфікаційної роботи отримано нові знання і практичні навички з управління ІТ-проєктами.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Rebuglio, M., Zenezini, G., Ottaviani, F. M., & De Marco, A. (2025). Project management information systems: A systematic review. *Procedia Computer Science*, 256, 1739–1747. <https://doi.org/10.1016/j.procs.2025.02.313>
2. Micale, R., La Fata, C. M., Lombardo, A., & La Scalia, G. (2021). Project management information systems (PMISs): A statistical-based analysis for the evaluation of software packages features. *Applied Sciences*, 11(23), 11233. <https://doi.org/10.3390/app112311233>
3. Rath, S. P., Jain, N. K., Tomer, G., & Singh, A. K. (2025). A systematic literature review of agile software development projects. *Information and Software Technology*, 182, 107727. <https://doi.org/10.1016/j.infsof.2025.107727>
4. Chen, M., Martins, T. S., Zhang, L., & Dong, H. (2025). Digital transformation in project management: A systematic review and research agenda. *Systems*, 13(8), 625. <https://doi.org/10.3390/systems13080625>
5. Nenni, M. E., De Felice, F., De Luca, C., & Forcina, A. (2025). How artificial intelligence will transform project management in the age of digitization: A systematic literature review. *Management Review Quarterly*, 75, 1669–1716. <https://doi.org/10.1007/s11301-024-00418-z>
6. Almalki, S. S. (2025). AI-driven decision support systems in agile software project management: Enhancing risk mitigation and resource allocation. *Systems*, 13(3), 208. <https://doi.org/10.3390/systems13030208>
7. Loureiro, E., Gomes, B., Varajão, J., & Silva, C. (2024). Information systems project success surveys: Insights from the last 30 years. *Heliyon*, 10(23), e40619. <https://doi.org/10.1016/j.heliyon.2024.e40619>
8. Храпкін, О., Кіндрат, О., & Чопей, Р. (2023). Управління проєктами в ІТ-галузі: методика, інструменти та керування ризиками. *Економіка та суспільство*, (55). <https://doi.org/10.32782/2524-0072/2023-55-110>

9. Onysko, Y., & Farmaga, I. (2024). Review and analysis of project management systems. *Computer Systems and Networks*, 6(1), 209-215. <https://doi.org/10.23939/cds2024.01.209>
10. Project Management Institute. (2022). *PMI lexicon of project management terms*. Project Management Institute. <https://www.pmi.org/pmbok-guide-standards/lexicon>
11. Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The definitive guide to Scrum: The rules of the game*. Scrum.org. <https://scrumguides.org>
12. Masood, Z., Hoda, R., & Blincoe, K. (2020). Real world Scrum: A grounded theory of variations in practice. *Empirical Software Engineering*, 26(4), Article 60. <https://doi.org/10.1109/TSE.2020.3025317>
13. Kerzner, H. (2022). *Project management: A systems approach to planning, scheduling, and controlling* (13th ed.). Wiley. 880 p.
14. Koi-Akrofi, G. Y., Koi-Akrofi, J., & Matey, H. A. (2019). Understanding the characteristics, benefits and challenges of agile IT project management: A literature-based perspective. *International Journal of Computer Applications*, 178(22), 1–7. <https://doi.org/10.5121/ijsea.2019.10502>
15. Leech, B., & Hanslo, R. (2025). The evolution of agile and hybrid project management methodologies: A systematic literature review. *arXiv*. <https://doi.org/10.48550/arXiv.2511.02859>
16. Насад, Н. В., & Крискун, І. М. (2023). Засади управління проєктами ІТ-компаній: методика та інструменти. *Економіка. Менеджмент. Бізнес*, (4), 85–92. <https://doi.org/10.31673/2415-8089.2025.015762>
17. Шабардіна, Ю. М., & Хоменко, О. В. (2024). Особливості сучасних підходів до управління проєктами в розподілених командах ІТ-сфери. *Науковий вісник Полісся*, (1), 112–120. [https://doi.org/10.25140/2410-9576-2025-1\(30\)-343-356](https://doi.org/10.25140/2410-9576-2025-1(30)-343-356)
18. Янишин, О. Б., Романенко, Н. В., & Мандзюк, О. І. (2022). Термінологія проєктно-командного менеджменту Scrum: перекладацький аспект.

- Translation Studies Journal*, (15), 74–82. <https://doi.org/10.32999/ksu2663-3426/2025-1-10>
- 19.Юрчук, Н. П., & Кіпоренко, С. С. (2023). Методології управління ІТ-проєктами: критерії вибору. *Вісник Вінницького політехнічного інституту*, (2), 96–103. <https://doi.org/10.32702/2307-2105.2024.11.76>
- 20.Мельник, Г. В., Калашник, В. Ю., & Стаценко, Д. В. (2021). Еволюція технологій управління ІТ-проєктами. *Вісник Київського національного університету технологій та дизайну*, (3), 45–53. Взято з <https://er.knutd.edu.ua/handle/123456789/31223>
- 21.Крамської, Д. Ю., Глізнуца, М. Ю., Соболев, Р. Г., & Крамської, О. Ю. (2022). Інформаційні системи як інструмент ефективного управління проєктами. *Вісник НТУ «ХПІ»*, (1), 58–66. <https://doi.org/10.20998/2519-4461.2025.3.54>
- 22.Основи управління ІТ проєктами [Електронний ресурс]: навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки»/ КПІ ім. Ігоря Сікорського ; уклад.: В. О. Кузьмініх, Р. А. Тараненко. Київ : КПІ ім. Ігоря Сікорського, 2019. – 75 с. Взято з <https://ela.kpi.ua/server/api/core/bitstreams/7c313e5c-5477-4be2-9806-d32e9eace0c3/content>
- 23.Elleh, F. (2023). *Project Management Waterfall-Agile-It-Data Science*, Authorhouse, 2023, 402 p.
- 24.Todaro, D. (2019). *The Epic Guide to Agile: More Business Value on a Predictable Schedule with Scrum*, R9 Publishing LLC, 518 p.
- 25.McCarthy, R. (2020). *Agile and Scrum: unlock the power of Agile project management, Lean Thinking, the Kanban process, and Scrum*, Author's Republic, 212 p.
26. Zainab Masood, Rashina Hoda, Kelly Blincoe (2020). Real World Scrum A Grounded Theory of Variations in Practice. *IEEE Transactions on Software Engineering*. 1–12, <https://doi.org/10.1109/TSE.2020.3025317>

27. Blyznyukova, I., Semko, I., & Kiyko, S. (2020). Огляд сучасних методологій управління командами ІТ-проектів. *Управління розвитком складних систем*, (43), 60–66. <https://doi.org/10.32347/2412-9933.2020.43.60-66>
28. Про захист персональних даних: Закон України від 01 червня 2010 року № 2297-VI. Взято з <https://zakon.rada.gov.ua/laws/show/2297-17#Text>