

**ПОЛТАВСЬКИЙ ДЕРЖАВНИЙ АГРАРНИЙ УНІВЕРСИТЕТ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕКОНОМІКИ, УПРАВЛІННЯ,  
ПРАВА ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ**

**Пояснювальна записка**

до кваліфікаційної роботи на здобуття ступеня вищої освіти магістр

на тему: **«Методології та програмні засоби управління проектами  
впровадження інформаційних систем»**

Виконав: здобувач вищої освіти  
за освітньо-професійною програмою  
Інформаційні управляючі системи та  
технології спеціальності  
126 Інформаційні системи та технології  
ступеня вищої освіти магістр  
групи 126ІСТмд\_21  
Чубенко В.М.  
Керівник: Уткін Ю. В.  
Рецензент: Петраш Р.В.

**Полтава – 2023 року**

## ВСТУП

Впровадження інформаційних систем у більшість аспектів діяльності в сучасному суспільстві потребує великої кількості все нових видів програмного забезпечення. Сфера ІТ висуває перелік нових знань та способів організації праці, суттєво змінює ринок професій. Сьогодні програміст має бути і маркетологом власного продукту, а компанії з розробки програмного забезпечення шукають нові засоби та інструменти підвищення ефективності діяльності команд. Розробка програмного забезпечення вимагає високого рівня професіоналізму, креативності та аналітичних навичок від розробників. Значну роль при цьому відводиться новітнім методам складання та реалізації проєктів розробки і впровадження інформаційних систем. Науковий напрямок теорії і практики управління проєктами інформаційних систем розвивається так само інтенсивно і використовує оригінальні технології та програмні продукти, методології, підходи.

*Актуальність* теми кваліфікаційної роботи розглядається пов'язана із необхідністю дослідження особливостей методів планування та реалізації проєктів програмного забезпечення, так і з вибором спеціальних інформаційних систем для здійснення і підтримки управління такими проєктами. Розумна автоматизація проєктної діяльності, оптимальний план і послідовність постановки і контролю виконання завдань, управління ресурсами і командою проєктів дозволяють здійснити розробку програмного забезпечення на потребу підприємств або бізнесу, масштабну автоматизацію або реінжиніринг виробничих, управлінських процесів з дотриманням кращих термінів та оптимальних затрат. Проєкти розробки і впровадження інформаційних систем вважаються одними з найактуальніших напрямків у сфері інформаційних технологій, тому потребують сучасних і ефективних методів та інструментів управління.

*Зв'язок роботи з науковими програмами, темами.* Кваліфікаційна робота виконана у відповідності до науково-дослідної ініціативної теми

«Організаційно-методологічні аспекти впровадження інформаційно-комунікаційних систем і технологій в управлінні діяльністю сучасних організацій та підприємств за умов переходу до цифрової економіки» ДРН 0117U003099, яка реалізується на кафедрі інформаційних систем та технологій Полтавського державного аграрного університету (ПДАУ).

*Мета* кваліфікаційної роботи – дослідження сучасних методологій та спеціальних програмних засобів підтримки управління проєктами розробки і впровадження інформаційних систем.

*Завдання* кваліфікаційної роботи:

- використання наукових підходів щодо систематизації теоретичних основ проєктів розробки і впровадження інформаційних систем як предметної області;
- аналіз сучасних методологій та підходів щодо створення і реалізації проєктів розробки і впровадження програмного забезпечення;
- здійснення порівняльних характеристик спеціальних систем управління проєктами та обґрунтування вибору для практичного використання;
- розробка прикладів типових проєктів галузі ІТ у середовищах різних спеціалізованих систем управління проєктами з висвітленням їхніх сильних і слабких сторін;
- оцінювання ефективності та доцільності застосування програмних засобів управління ІТ-проєктами.

*Об'єкт дослідження* – ключові процеси розробки і впровадження проєктів інформаційних систем з використанням спеціальних програм і засобів на різних етапах життєвого циклу.

*Предмет* дослідження – методології розробки й реалізації проєктів інформаційних систем, функціональні характеристики систем управління проєктами.

*Методи наукових досліджень*: аналітико-синтетичний, інформаційно-пошуковий, порівняльного аналізу, емпіричний, кейс-метод, графічний, методи оцінювання економічної ефективності тощо.

*Інформаційна база* роботи сформована на основі наукових статей, основоположних праць і міжнародних стандартів з управління проєктами програмного забезпечення, офіційних сайтів розробників систем управління проєктами, рейтингових звітів міжнародних аналітичних компаній.

*Елементи наукової новизни роботи* включають сформовані висновки щодо спроможності різних систем автоматизувати ключові операції в проєктах інформаційних систем, авторське узагальнення результатів досвіду практиків та досліджень науковців в області ІТ-проєктів.

*Практична значущість* роботи полягає в систематизації різних методологій і моделей розробки проєктів інформаційних систем, розробленні методичних підходів та рекомендацій щодо вибору й використання спеціальних інформаційних систем управління проєктами.

*Апробація результатів* дослідження відбувалася шляхом оприлюднення доповідей на наукових конференціях. За результатами дослідження опубліковано тези: «Determining the Project Model for Implementation ERP Class Information System for the Resurses and Activities Management in Territorial Communities», матеріали XII Міжнародної наукової конференції «Інформаційні технології в енергетиці та агропромисловому комплексі», 04-06 жовтня 2023 р., м. Львів; «Місце ERP-систем у фреймворку технологій Індустрії 4.0», Матеріали I Міжнародної науково-практичної конференції «Стратегічний менеджмент агропродовольчої сфери в умовах глобалізації економіки: безпека, інновації, лідерство», 28 вересня 2023 р., м. Полтава; «Особливості сучасних методологій проєктів розробки програмного забезпечення», матеріали XX щорічного міждисциплінарного семінару «Студентські роботи за науковою тематикою кафедри інформаційних систем та технологій ННІ ЕУПТ ПДАУ», 29 листопада 2023 р., м. Полтава.

Структура і обсяг кваліфікаційної роботи: пояснювальна записка викладена на 71 сторінці і складається зі змісту, вступу, трьох розділів, списку використаних джерел та додатків. Робота містить 7 таблиць і 41 рисунок.

# РОЗДІЛ 1

## ТЕОРЕТИЧНІ ТА МЕТОДОЛОГІЧНІ АСПЕКТИ УПРАВЛІННЯ ПРОЄКТАМИ ІНФОРМАЦІЙНИХ СИСТЕМ

### 1.1 Теоретичні основи управління проєктами

Теоретичні основи управління проєктами - це галузь знань, що вивчає принципи, методи, моделі та інструменти для планування, організації, реалізації та контролю проєктів. Загалом, проєкт – це унікальний комплекс завдань, що має чітко визначену мету, обмежені ресурси, терміни виконання та очікувані результати. Управління проєктами – це застосування знань, навичок, методів та технологій для досягнення цілей проєкту з урахуванням обмежень та ризиків [1].

Історія теорії управління проєктами сягає давніх часів, коли люди виконували складні завдання, такі як будівництво пірамід, великих храмів або воєнних кампаній. Проте сучасна теорія управління проєктами сформувалася лише у ХХ столітті, завдяки розвитку науки, технологій і промисловості, коли терміни проєктів потребували чітких рамок, так само як і обліку ресурсів та видів робіт. Основними етапами розвитку теорії управління проєктами були: створення методу критичного шляху (CPM) і методу аналізу програм і бюджетування (PERT) у 1950-х роках; запровадження професійних стандартів і сертифікації управління проєктами у 1960-1970-х роках; поширення комп'ютерних програм і інформаційних систем для планування і контролю проєктів у 1980-1990-х роках; розширення сфери застосування управління проєктами на різноманітні галузі діяльності, такі як освіта, охорона здоров'я, державне управління тощо, у 2000-2010-х роках. Сьогодні теорія управління проєктами є важливою складовою успішної реалізації будь-якої ідеї або інновації. Поступово сформувався своєрідний «світ управління проєктами», який об'єднав фахівців різних континентів і країн, напрямків і сфер діяльності, національностей і культур; утворилися світові організації, наприклад Інститут управління проєктами (Project Management Institute, PMI, США) [1], Міжнародна

асоціація управління проєктами (Європа), які відграють керівну роль у формуванні методології, правил, принципів та інших напрямків управління проєктами. По суті виникла нова світова культура, яка об'єднала мільйони людей навколо вдосконалення оточуючого світу, технологій, знань.

Під патронатом РМІ була видана перша версія колективної роботи – зведення знань із управління проєктами (Guide to Program Management Body of Knowledge, PMBOK) [2], яка має вже 7 видань, оновлюється у відповідності до потреб сучасності та виступає інтегратором і провідником у світі проєктного менеджменту. Все це відіграло суттєву роль у розвитку управління проєктами взагалі, зокрема й ІТ-проєктами.

Особливості створення і впровадження програмного забезпечення, яке є специфічним продуктом, поклали початок розвитку сучасних технологій управління ІТ-проєктами. Однак, за принципами планування і загальними структурними елементами всі проєкти мають подібні риси, елементи і правила. Визначень проєктів існує безліч, але основне визначення проєкту є загальним і спільним без деталізації.

Проєкт (від лат. *projectus*) це «...задум, ідея, образ, втілений у форму опису, обґрунтування, розрахунків, креслень, які розкривають суть задуму і можливість його практичного застосування» [3]. За своєю сутністю поняття проєкт трактується як план, задум; прообраз (прототип) об'єкта, явища чи процесу. Іншими словами, проєкт – це сукупність певних дій, обмежених у часі і спрямованих на вирішення проблеми або досягнення конкретної мети. До найбільш загальних ознак будь-якого проєкту належать:

- регламентований час виконання із визначеними датами початку і завершення;
- визначений перелік та кількість ресурсів, які обмежені;
- передбачає керування змінами;
- унікальний і неповторюваний [4].

Класифікація проєктів – це процес розподілу проєктів за певними критеріями, такими як обсяг, складність, терміни, ризики, бюджет тощо.

Класифікація проектів за сферою діяльності, типом діяльності, за типом продукту, масштабом може бути представлена схематично (рис. 1.1).



Рисунок 1.1 –Класифікація проектів за найбільш вагомими критеріями

Класифікація проектів допомагає визначити оптимальну стратегію управління проектами, вибрати найефективніші методи та інструменти, а також забезпечити належний рівень контролю та моніторингу. Коли проєкт завершено, в світі щось змінюється: скажімо, з'являється новий будинок, нова комп'ютерна система або якийсь інший продукт. Зрозуміло, що один проєкт може мати різні ознаки. Наприклад, технічний проєкт може бути локальним, інформаційним (див. рис. 1.1). Серед основних етапів проєктної діяльності, як правило, є такі:

- розробка концепції проєкту;
- планування проєкту
- складання бюджету;
- оцінка життєздатності проєкту;
- захист проєкту;
- попередній контроль;
- етап реалізації проєкту;
- корекція проєкту за підсумками;
- завершення проєкту і його ліквідація.

Розробка концепції проєкту є визначальним етапом, який впливає на успіх всього проєкту. Концепція проєкту – це основні положення, що представлені у певній системі і поєднують основні його характеристики. Загальна концепція проєкту, як правило, подається у формі короткого опису (табл. 1.1).

Таблиця 1.1 – Складові концепції проєкту та їхній зміст

Елемент концепції	Значення елемента в описі концепції проєкту
Актуальність	Коротке формулювання ситуації, яка потребує змін, економічні, соціальні наслідки.
Постановка проблеми	Проблема розглядається як протиріччя між тим, що є і тим, як має бути. Постановка проблеми виявляє тих, кого це стосується: чия це проблема (люди, організації тощо); які масштаби проблеми; чи підлягає проблема розв'язанню взагалі; визначаються характеристики проблеми.
Мета проєкту	Бажаний кінцевий результат. Проєкти мають чітку, заздалегідь визначену мету, яку потрібно досягти в обумовлені терміни і в рамках бюджету. Досягнення результату намагаються чітко спланувати, щоб знати, в які терміни завершити проєкт і скільки коштів на це можна витратити
Завдання проєкту	Дія, направлена на досягнення мети. Як правило, задачі ставляться щось створити, об'єднати, змінити, покращити і т. ін. Задачі вказують на конкретний результат
Методи проєкту	Спосіб виконання дії

Особливу увагу при ініціації проєктів приділяють формулюванню цілей. Ціль дозволяє чітко відповісти на питання: «Що буде досягнуто при реалізації проєкту?». При сучасному формулюванні цілей дотримуються правила SMART [5], яке поєднує п'ять характеристик, сутність яких можна трактувати наступним чином:

- (S, Specific) Специфічність: потребує, щоб сформульована мета давала чітке якісне уявлення про унікальні та інноваційні властивості майбутнього продукту проєкту порівняно з іншими альтернативами;

- (M, Measurable) Вимірюваність: потребує, щоб сформульована мета давала чітке якісне уявлення про унікальні та інноваційні властивості майбутнього продукту проєкту порівняно з іншими альтернативами;

- (A, Attainable) Досягненність: передбачає встановлення вимірюваних показників вартості, тривалості;

– (R, Realistic) Реалістичність: показує, що мета є такою, що її можна досягти з урахуванням реально доступних можливостей і обмежень (людських, фінансових тощо);

– (T, Time-related) Обмеженість у часі: Зумовлює необхідність прив'язки мети проєкту до певних обґрунтованих термінів.

Правильно деталізовані за допомогою SMART-методу цілі дозволяють сформулювати загальне уявлення про проєкт, дають змогу зрозуміти масштаби і особливості проєкту. Отже, проєкт – це є засіб інноваційного розвитку. Мета – формулювання того, що бажано досягти. Стратегія – вибір засобів досягнення цілі. Проєкти є перетворенням стратегії в конкретні дії, реалізацією цілей. Успішне управління проєктом залежить від чіткого підходу, здатності контролювати процес виконання робіт, уваги до деталей і вправного керівництва командою проєкту.

## **1.2 Моделі життєвого циклу проєкту і програмного продукту**

Життєвий цикл проєкту – це послідовність етапів, через які проходять проєкти від ініціації до завершення, незалежно від їх специфіки. Відрізок часу між цими подіями називають життєвим циклом проєкту [6]. Життєвий цикл (ЖЦ) проєкту є базовим, вихідним поняттям для дослідження проблем реалізації проєкту, фінансування робіт, прийняття рішень про доцільність капіталовкладень та деталізації проєкту. Життєвий цикл проєкту зазвичай складається з чотирьох основних фаз: ініціювання, планування, виконання та закриття. Кожна фаза має свої цілі, завдання, ресурси, ризики та результати. Життєвий цикл проєкту допомагає керувати проєктом ефективно та контролювати його якість, обсяг, терміни та вартість.

Фаза проєкту (*Project Phase*) – набір логічно взаємопов'язаних робіт проєкту, в процесі завершення яких досягається один з основних результатів проєкту [2]. Фази проєкту (або просто фази), як правило, виконуються

послідовно, але в окремих ситуаціях можуть перекриватися. На сьогодні відсутній загальний, єдиний підхід, який би чітко регламентував кількість фаз і етапів, їхній зміст, термінологічні означення тощо. Програмний продукт це сукупність програм та технічної документації по їх встановленню, налаштуванню, використанню і доопрацюванню. Відповідно до стандарту життєвий цикл програми, інформаційної системи, програмного продукту складається з розробки, розгортання, підтримки та супроводу.

Найпоширенішою можна назвати чотирьохфазну модель життєвого циклу (рис. 1.2). Вона містить послідовні фази ініціалізації, розроблення, реалізації і завершення проєкту [7].

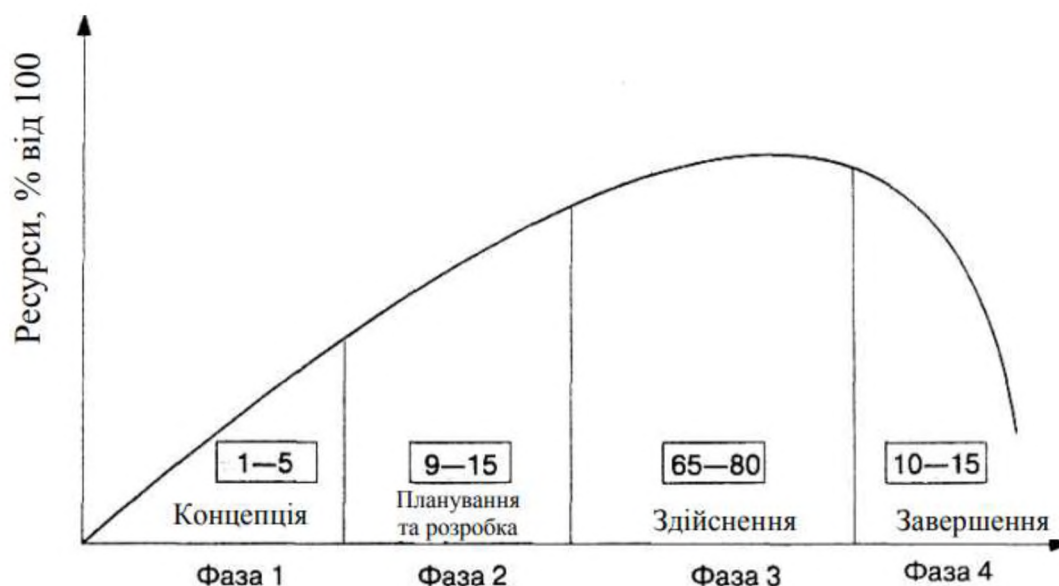


Рисунок 1.2 – Чотирьохфазна модель життєвого циклу проєкту (за [7])

Фази можна розділити на підфази, етапи, стадії. У випадку, якщо проєкт або його частини поділені на фази, така ієрархія буде представлена в структурі ієрархічної роботи. Фази є нерівномірними за часом та ресурсами. Так, на фазі ініціалізації відбувається розроблення концепції задуманого проєкту, проєктних документів, які дадуть можливість замовнику та інвестору ухвалити рішення щодо інвестування. Ця фаза займає до 5% загального бюджету проєкту. Під час другої фази – розроблення, відбувається розроблення основних компонентів проєкту (кінцевого продукту) та підготування до реалізації, планування.

Програмний продукт це сукупність програм і супровідної документації по їх встановленню, налаштуванню, використанню і доопрацюванню. Відповідно до стандарту [8] ЖЦ програми, програмної системи, програмного продукту включає в себе розробку, розгортання, підтримку та супровід. Найбільш популярною і зваженою є каскадна модель життєвого циклу проекту розробки програмного забезпечення (рис. 1.3).

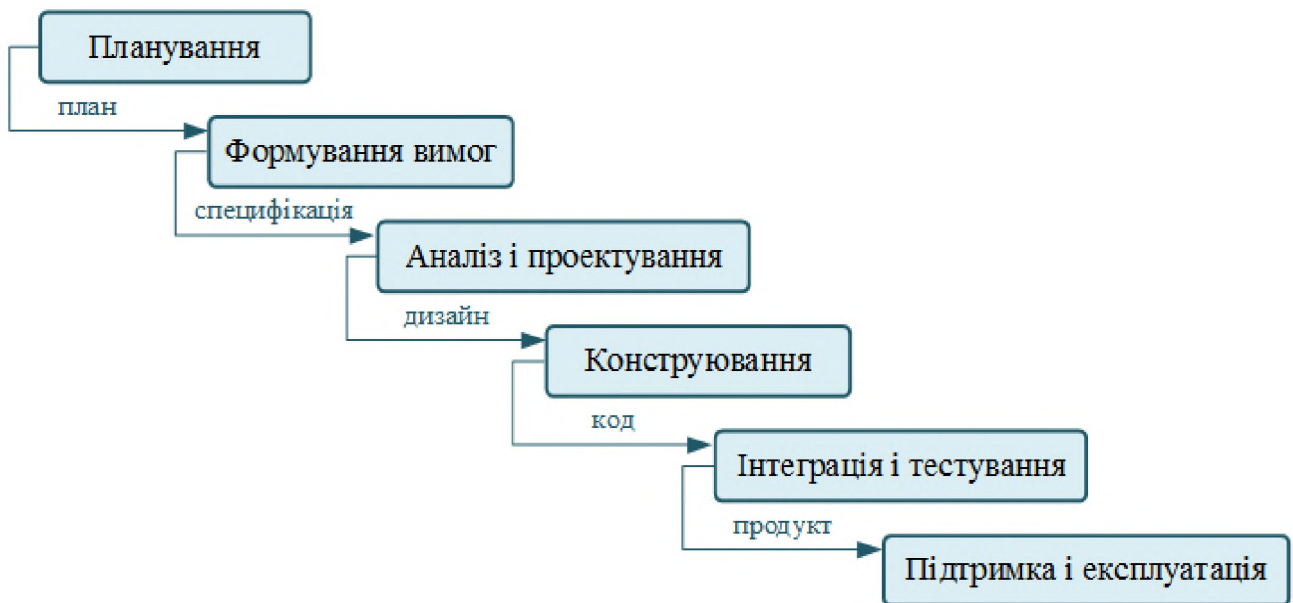


Рисунок 1.3 – Каскадна модель життєвого циклу програмного забезпечення

Якщо програмний продукт виготовлений не в коробковому форматі, а є досить складним, то його розгортання у клієнтів потребує, як правило, самостійного проекту впровадження. Супровід передбачає усунення критичних несправностей в системі і часто реалізується не як проект, а як процесна діяльність. Підтримка полягає в розробці нової функціональності, переробці вже існуючої функціональності у зв'язку зі зміною вимог, і поліпшенням продукту, а також усунення некритичних зауважень до ПЗ, виявлених при його експлуатації. ЖЦ програмного продукту завершується виведенням продукту з експлуатації і зняттям його з підтримки і супроводу. Процес розробки ПЗ – сукупність процесів, що забезпечують створення і розвиток ПЗ.

Специфічна діяльність з розробки ПЗ дала поштовх новим проектним моделям. Автором Баррі Боемом була запропонована удосконалена модель

життєвого циклу, яка стала прикладом застосування еволюційної стратегії конструювання – спіральна модель [9]. Порівняно з каскадною в новій моделі додається новий елемент – аналіз ризиків (рис. 1.4).

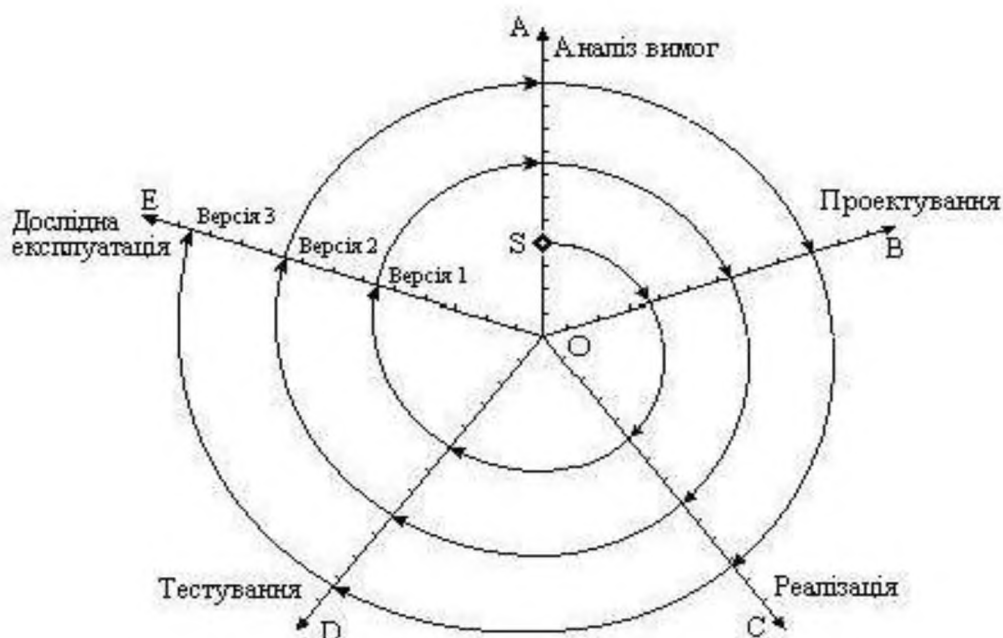


Рисунок 1.4 – Спіральна модель Боема проєкту створення ПЗ (за [9])

Основними перевагами спіральної моделі (див. рис. 1.4) є найбільш реальне відображення процесу розробки ПЗ, можливість явно враховувати ризик на кожному кроці еволюції. Серед недоліків – ускладнення контролю над розробкою.

Надзвичайно важливо скласти детальний та вичерпний план. На етапі реалізації відбуваються заплановані роботи, здійснюється фінансування, витрачаються ресурси. Ця фаза найбільш ємнісна за витратами часу та ресурсів (див. рис.1.2). Фаза закриття проєкту передбачає досягнення кінцевої мети – передача замовнику в користування створений продукт проєкту. На цій фазі підбивають підсумки, здійснюють архівацію, визначають якість проєктних робіт.

У міжнародній практиці прийнято інакше називати основні фази проєктного циклу: передінвестиційна (розробка проєкту як документа); інвестиційна (формування активів проєкту «під ключ»); експлуатаційна (запуск та господарська експлуатація ресурсів та регулярне отримання прибутку,

повернення вкладених коштів); ліквідаційна [10]. Кожна з цих фаз, у свою чергу, поділяється на стадії. Згідно з довідником UNIDO виділяються чотири такі стадії передінвестиційної фази: пошук інвестиційних концепцій (opportunity studies); попередня підготовка проєкту (pre-feasibility studies); остаточна підготовка проєкту і оцінка його техніко-економічної і фінансової прийнятності (feasibility studies); стадія фінального розгляду і ухвалення рішення по проєкту (final evaluation) [11].

Програмою промислового розвитку ООН (UNIDO) було запроваджене представлення проєкту у вигляді послідовності трьох основних фаз (рис. 1.5): передінвестиційної, інвестиційної та експлуатаційної [12].

Життєвий цикл проєкту		
Передінвестиційна (Pre-Investment) фаза	Інвестиційна Investment, Implementation, Execution) фаза	Експлуатаційна (Execution, Operation) фаза

Рисунок 1.5 – Трифазна модель життєвого циклу згідно UNIDO [11]

Інвестиційна фаза є частиною фази впровадження, під час якої виконується план проєкту. Основні дії, що відбуваються під час інвестиційної фази, можна деталізувати наступним поясненням [10].

Закупівля: команда проєкту починає закупівлю необхідних ресурсів, включаючи матеріали, обладнання та людські ресурси, необхідні для виконання проєкту.

Розподіл ресурсів: ресурси розподіляються між різними діями проєкту, і команда проєкту переконується, що ресурси використовуються ефективно та ефективно.

Побудова потужностей: проводяться дії з побудови потужностей для підвищення навичок та знань команди проєкту та інших зацікавлених сторін, що беруть участь у проєкті.

Моніторинг проєкту: команда проєкту контролює хід проєкту, щоб переконатися, що він знаходиться на правильному шляху, і що будь-які проблеми або виклики вирішуються вчасно.

Комунікація та звітність: команда проєкту спілкується з зацікавленими сторонами та надає регулярні звіти про хід роботи, щоб тримати їх в курсі стану проєкту.

Управління ризиками: команда проєкту виявляє потенційні ризики та розробляє плани управління ризиками для їх зменшення.

Контроль якості: команда проєкту переконується, що результати проєкту відповідають вимогам до якості, і що вони поставлені вчасно та в межах бюджету.

Загалом, інвестиційна фаза є критичною для успішного впровадження плану проєкту. Вона вимагає ефективного управління проєктом, розподілу ресурсів, управління ризиками та контролю якості, щоб забезпечити завершення проєкту вчасно, в межах бюджету та з бажаними результатами

Процес реалізації кожної фази проєкту протікає в певних часових межах (має початок і закінчення). У кожний період часу проєкт характеризується певною інтенсивністю інвестицій. Залежність «час-інтенсивність інвестицій» характеризує динаміку процесу розвитку проєкту по фазах життєвого циклу.

Трифазна модель життєвого циклу проєкту – це підхід, який ділить проєкт на три основні фази: планування, виконання та закриття. Кожна фаза має свої цілі, завдання, ресурси та результати. Трифазна модель допомагає керувати проєктом ефективно та систематично, а також забезпечує контроль якості та ризиків на всіх етапах. Якщо порівняти схеми на рис. 1.3-1.5, то можна помітити, що в трифазній моделі передінвестиційна фаза включає перші дві з чотирифазної моделі, тобто більш укрупнена. Це іще раз підтверджує досить умовний поділ проєкту на фази, тому варто зосередитися на конкретному змісті його етапів.

Управління проєктами – це процес планування, організації, розподілу ресурсів та контролю за виконанням завдань, пов'язаних з досягненням конкретних цілей. Управління проєктами вимагає від керівника та його команди

застосування знань, навичок, методів та інструментів для забезпечення якості, обсягу, часу та бюджету проєкту. Управління проєктами також включає інтереси зацікавлених сторін, управління ризиками, змінами, комунікаціями та якістю.

### **1.3 Сучасні методології та методи управління проєктами розробки програмного забезпечення**

#### **1.3.1 Класичні моделі розробки програмного забезпечення. SWEBOOK**

Проєкти створення програмного забезпечення (ПЗ), які стали основою так званих ІТ-проєктів, мають низку особливостей, пов'язаних із особливостями та новизною самого продукту. Термін software (програмне забезпечення) увів у 1958 р. всесвітньо відомий статистик Джон Тьюкей (John Tukey) [12]. Термін software engineering (програмна інженерія) вперше з'явився у назві конференції НАТО, що відбулася в Німеччині в 1968 р. і була присвячена відомій кризі програмного забезпечення. За період 1990-1995 рр. велася робота зі створення міжнародного стандарту, який повинен був дати єдине уявлення про процеси розробки ПЗ. У результаті був випущений стандарт ISO/IEC 12207 [8]. У 2004 р. опублікована основоположна праця «Керівництво до зводу знань з програмної інженерії» (SWEBOOK) [13], в якій були зібрані основні теоретичні і практичні знання, накопичені в цій галузі.

Згідно SWEBOOK 2004, програмна інженерія включає в себе 10 основних та 7 додаткових галузей знань, на яких базуються процеси розробки ПЗ. До основних галузей знань відносяться області, що наведені далі.

- 1) Software requirements – вимоги до програми.
- 2) Software design – дизайн (архітектура).
- 3) Software construction – конструювання програмного забезпечення.
- 4) Software testing – тестування програмного забезпечення.
- 5) Software maintenance – експлуатація (підтримка) ПЗ.
- 6) Software configuration management – управління конфігурацією.

- 7) Software engineering management – управління у програмній інженерії.
- 8) Software engineering process – процеси програмної інженерії.
- 9) Software engineering tools and methods – інструменти і методи.
- 10) Software quality – якість програмного забезпечення.

Враховуючи складність, мінливість, високий рівень абстракції програмування, утворилося чимало моделей розробки ПЗ. Всі вони мають свої переваги і недоліки. Методології процесів розробки ПЗ прийнято класифікувати за «вагою» – наявності кількості формалізованих процесів (більшість процесів або тільки основні) і детальності їх регламентації. Чим більше процесів документовано, ніж більш детально вони описані, тим більше «вага» моделі.

Розробка моделі SW-CMM почалася в середині 80-х років минулого століття Міністерством оборони США при вирішенні проблеми про вибір розробників ПЗ для реалізації великих програмних проєктів. Інститут програмної інженерії, що входить до складу Університету Карнегі-Меллона, розробив SW-CMM, Capability Maturity Model for Software [14] в якості еталонної моделі організації розробки програмного забезпечення. Дана модель визначає п'ять рівнів зрілості процесу розробки ПЗ.

1) Початковий. На початковому рівні процеси є неорганізованими, випадковими і навіть хаотичними. Успіх, ймовірно, залежить від індивідуальних зусиль і не вважається повторним. Це пов'язано з тим, що процеси недостатньо визначені та задокументовані, щоб їх можна було тиражувати.

2) Повторюваний. На повторюваному рівні встановлюються, визначаються та документуються необхідні процеси. В результаті напрацьовуються базові методи управління проєктами, і успіхи в ключових областях процесу можуть бути повторені.

3) Визначений. На визначеному рівні організація розробляє власний стандартний процес розробки програмного забезпечення. Ці визначені процеси дозволяють більше уваги приділяти документації, стандартизації та інтеграції.

4) Керований. На керованому рівні організація відстежує та контролює власні процеси за допомогою збору та аналізу даних.

5) Оптимізація. На рівні оптимізації процеси постійно вдосконалюються шляхом моніторингу зворотного зв'язку від процесів і впровадження інноваційних процесів і функціональності.

Уніфікований процес RUP (Rational Unified Process, RUP) [15] був створений компанією Rational Software у 1990-х роках і з тих пір став однією з найбільш широко використовуваних методологій розробки програмного забезпечення. Три ключові елементи, які визначають RUP:

1) Інструкції з розробки програмного забезпечення, які є основою для успіху. RUP було розроблено навколо цих концепцій.

2) Фреймворк, який включає багаторазово використовувані будівельні блоки для вмісту методів і процесів, з яких можна створювати індивідуальні процедури та налаштування методів.

3) Мова, яка описує методи та процеси.

RUP – це систематичний спосіб розподілу завдань і обов'язків у команді розробників, який пропонує найкращі практики та вказівки для ефективної розробки програмного забезпечення. Завдяки цьому вона може виробляти високоякісне програмне забезпечення вчасно та в рамках бюджету, задовольняючи при цьому вимоги своїх клієнтів. У RUP є чотири основні фази: початок, розробка, конструкція та перехід. Кожна фаза має власні цілі, заходи та результати.

Модель Microsoft Solutions Framework (MSF) [16] – досить гнучка і легка модель, побудована на основі ітеративної розробки. Привабливою особливістю MSF є велика увага до створення ефективної гнучкої проєктної команди. Для досягнення цієї мети MSF пропонує досить нестандартні підходи до організаційної структури, розподілу відповідальності та принципами взаємодії всередині команди.

PSP/TSP Одна з останніх розробок Інституту програмної інженерії Personal Software Process / Team Software Process [17,18]. Ця модель визначає вимоги до компетенцій розробника. Згідно цієї моделі кожен програміст повинен вміти:

- враховувати час, витрачений на роботу над проєктом;

- враховувати знайдені дефекти;
  - класифікувати типи дефектів;
  - оцінювати розмір задачі;
  - здійснювати систематичний підхід до опису результатів тестування;
  - планувати програмні завдання;
  - розподіляти їх по часу і складати графік роботи.
  - виконувати індивідуальну перевірку проєкту та архітектури; •
- здійснювати індивідуальну перевірку коду;
- виконувати регресійне тестування. Team Software Process робить ставку на самокеровані команди чисельністю від 3 до 20 розробників.

### **1.3.2 Методологія Agile**

Agile означає гнучкі технології програмних проєктів. Методологія Agile в управлінні проєктами – це гнучкий підхід до розробки та впровадження програмного забезпечення, який базується на ітеративному та інкрементному процесі. Вона дозволяє командам швидко адаптуватися до змінних вимог клієнтів та ринку, а також постійно покращувати якість продукту. Методологія Agile включає в себе ряд цінностей та принципів, які спрямовані на співпрацю, гнучкість, самоорганізацію, неперервне навчання та залучення користувачів до процесу розробки.

Цінності Agile в управлінні проєктами - це набір принципів, які спрямовані на підвищення гнучкості, адаптивності та співпраці в командах, що розробляють складні продукти. Ці цінності базуються на Маніфесті Agile, який був створений в 2001 р. групою експертів з розробки програмного забезпечення [19]. За Маніфестом Agile, команди повинні надавати перевагу:

- людям і взаємодії над процесами та інструментами;
- функціонуючому продукту над детальною документацією;
- співпраці з замовником над дотриманням договору;
- реагуванню на зміни над слідуванням плану.

Ці цінності підкреслюють важливість людського фактору, адаптивності, постійного вдосконалення та співпраці з клієнтами в процесі розробки

програмного забезпечення. Вони слугують основою для 12 принципів Agile [20], які разом формують основу Agile управління проєктами (табл. 1.2).

Таблиця 1.3 – Формулювання 12 принципів Agile

№ з/п	Формулювання (зміст) кожного з принципів Agile
1	Нашим головним пріоритетом є задоволення клієнта шляхом ранньої та постійної поставки цінного програмного забезпечення.
2	Зміни вимог приймаються навіть на пізніх стадіях розробки.
3	Постачання робочого програмного забезпечення повинно відбуватися через короткі проміжки часу.
4	Бізнес-люди та розробники повинні працювати разом щодня протягом всього проєкту.
5	Будуйте проєкти навколо мотивованих осіб. Надайте їм необхідне середовище та ресурси, і довірте їм виконання роботи.
6	Найефективніший спосіб передачі інформації до та в межах команди розробників - це безпосередні розмови.
7	Робоче програмне забезпечення є основним критерієм прогресу.
8	Agile-процеси сприяють сталому розвитку.
9	Постійна увага до технічної відмінності та гарного дизайну підвищує гнучкість.
10	Простота - мистецтво мінімізації необхідної роботи - важлива.
11	Найкращі архітектурні рішення, вимоги та дизайни виникають з самоорганізованих команд.
12	Команда повинна регулярно аналізувати свою роботу та коригувати її, щоб підвищити ефективність.

Ці принципи підкреслюють важливість людського фактору, адаптивності, постійного вдосконалення та співпраці з клієнтами в процесі розробки програмного забезпечення.

Основна ідея всіх гнучких моделей полягає в тому, що застосовуваний у розробці процес повинен бути адаптивним. Вони декларують своєю вищою цінністю орієнтованість на людей і їх взаємодію, а не на процеси і засоби. По суті, так звані, гнучкі методології це не методології, а набір практик, які можуть дозволити (але не завжди) домагатися ефективної розробки ПЗ, ґрунтуючись на ітеративності, інкрементальності, самоврядності команди і адаптивності процесу.

Один з авторів «Agile Manifesto» (Маніфесту гнучкої розробки ПР) [21] проаналізував дуже різні програмні проєкти, які виконувалися по різних моделям від абсолютно полегшених і «гнучких» до важких (СММ-5) за останні

20 років [22]. Він не виявив кореляції між успіхом чи провалом проєктів і моделями процесу розробки, які застосовувалися в проєктах. Звідси він зробив висновок про те, що ефективність розробки ПЗ не залежить від моделі процесу, а також про те, що: у кожного проєкту повинна бути своя модель процесу розробки; у кожної моделі – свій час [23, 24]. Це означає, що в кожному новому проєкті процес повинен визначатися щоразу заново, в залежності від проєкту, продукту та персоналу, у відповідності з «Законом 4-х П» (рис. 1.6). Абсолютно різні процеси повинні застосовуватися в проєктах, в яких беруть участь 5 осіб, і в проєктах, в яких беруть участь 500 чоловік.



Рисунок 1.6 – Елементи правила 4-х «П» визначення процесу проєкту

Команда, яка починала проєкт, не залишається незмінною, вона проходить певні стадії формування і, як правило, кількісно зростає в міру розвитку проєкту. Тому процес повинен постійно адаптуватися до цих змін. Головний принцип: не люди повинні будуватися під обрану модель процесу, а модель процесу повинна підлаштовуватися під конкретну команду, щоб забезпечити її найвищу ефективність. Стів Макконнелл у своїй книзі [25] призводить тест програмного проєкту на виживання. Цей чек-лист з 33-х пунктів. Керівник програмного проєкту повинен його періодично використовувати для внутрішнього аудиту своїх процесів.

В управлінні проєктами розробки програмного забезпечення можна виділити два ключових напрямки: сучасні методи і технології розробки

програмного забезпечення як продукту проєкту, а також методи управління такими проєктами. Щодо першої складової, то напрямок робіт має тренд до використання різних інтегрованих програмних середовищ (IDE) при розробці ПЗ, використання хмарних репозитаріїв готових кодів. В ІТ-проєктах при опису використовується спеціальний термін DevOps.

### 1.3.3 Методологія розробки програмного забезпечення DevOps

DevOps означає набір практик, які поєднують розробку (Development, Dev) та ІТ-операції (Operations, Ops) [26]. DevOps – це методологія розробки та підтримки програмного забезпечення, яка поєднує процеси розробки, тестування, випуску, експлуатації та моніторингу. Метою DevOps є забезпечити швидку, надійну та безперервну доставку програмних продуктів та послуг. DevOps спирається на культуру співпраці, автоматизацію, неперервну інтеграцію та неперервну доставку. Методологія передбачає активну взаємодію фахівців з розробки з фахівцями з інформаційно-технологічного обслуговування та взаємну інтеграцію їх технологічних процесів один в одного для забезпечення високої якості програмного продукту. Призначена для ефективної організації створення та оновлення програмних продуктів та послуг. Заснована на ідеї тісної взаємозалежності створення продукту та експлуатації програмного забезпечення, яка прищеплюється команді як культура створення продукту. Схема управління розробкою ПЗ за методологією DevOps [27] представлена на рис. 1.7.

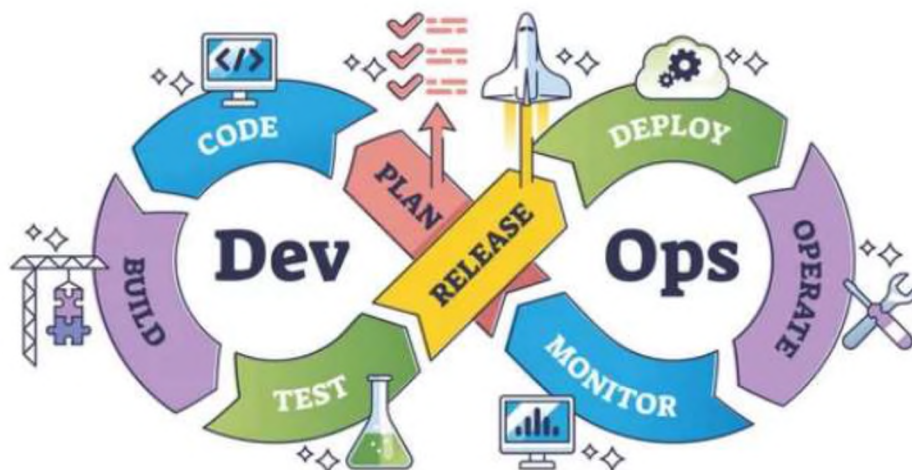


Рисунок 1.7 – Схематичне зображення концепцій DevOps-методології

Ця схема (див. рис. 1.7) показує, як різні компоненти DevOps, такі як неперервна інтеграція (CI), неперервна доставка (CD) та неперервний моніторинг, взаємодіють між собою в рамках одного процесу. В роботі [27] зазначено, що «DevOps – це спільна та багатопрофільна робота в рамках організації з автоматизації безперервної доставки нового програмного забезпечення». Моделі DevOps є специфічними та сучасними моделями розробки ПЗ. Зовсім недавно все більше компаній, що займаються програмним забезпеченням, перейняли DevOps, щоб адаптуватися до бізнес-середовища, яке постійно змінюється [28].

DevOps – це спільна та міждисциплінарна організаційна робота з автоматизації безперервної доставки нових оновлень програмного забезпечення, гарантуючи при цьому їх правильність і надійність.

Дослідження, в якому було опитано майже 2200 ІТ-керівників і професіоналів галузі, виявило, що серед керівних посад лише 17 % визначили DevOps як ключову методологію. Це набагато нижче від програмного забезпечення як послуги (42 %), великих даних (41 %) і публічної хмарної інфраструктури як послуги. (39 %). «Незважаючи на те, що вони традиційно пов'язані зі збільшенням швидкості виходу на ринок, ключовими рушійними факторами використання фреймворків і наборів інструментів, пов'язаних із DevOps, залишаються масштабованість і скорочення операційних витрат», — зазначається у звіті. Незважаючи на те, що DevOps і методології автоматизації все частіше використовуються для доставки додатків, лише кожен п'ятий респондент опитування [29] відмітив, що DevOps має стратегічний вплив на їхню організацію.

## **Висновки до розділу 1**

Управління проєктами інформаційних систем та розробки програмного забезпечення мають спільні керівні засади, які базуються на теорії управління проєктами як порівняно новому науковому напрямку. Теоретичні основи

управління проєктами ведуть відлік від 70-х років минулого століття, коли виникли такі міжнародні організації, як Інститут управління проєктами (PMI), європейська Міжнародна асоціація управління проєктами, а також вийшла перша версія колективної роботи зведення знань із управління проєктами PMBOK. Загалом, проєкт – це сукупність певних дій, обмежених у часі і спрямованих на вирішення проблеми або досягнення конкретної мети. Проєкти класифікують за різними критеріями та ознаками. Проєкти інформаційних систем є одним із видів технічних інформаційних проєктів. Як і всі інші, проєкти інформаційних систем мають сформовану концепцію, мету, завдання, вирішують певне коло проблем, мають методологію досягнення цілей.

Життєвий цикл проєкту є базовим, вихідним поняттям для дослідження проблем реалізації проєкту, фінансування робіт, прийняття рішень про доцільність капіталовкладень та деталізації проєкту. Найчастіше використовують трьох- або чотирьохфазну модель життєвого циклу проєктів. Каскадна (водоспадна) модель життєвого циклу є найбільш апробованою моделлю представлення життєвого циклу проєкту. Більш сучасними є спіральна модель Боема та інші адаптивні моделі.

Існують різні методології реалізації проєктів програмного забезпечення від класичних до новітніх, як Agile або DevOps технології. Останні поки що не мають визначеного пріоритету, з приводу їхньої ефективності в науковому світі ведуться дискусії та обговорення.

Завданнями наступних розділів даної роботи є аналіз функціоналу спеціального програмного забезпечення для підтримки проєктної діяльності та розробка порівняльних прикладів реалізації тестових проєктів інформаційних систем.

## РОЗДІЛ 2

### АНАЛІЗ ФУНКЦІОНАЛУ ТА ХАРАКТЕРИСТИКИ ІНФОРМАЦІЙНИХ СИСТЕМ ПІДТРИМКИ УПРАВЛІННЯ ПРОЄКТАМИ

#### **2.1 Порівняльні характеристики спеціальних інформаційних систем управління проєктами за функціональними можливостями**

Важливість управління проєктами в бізнес-компаніях, організаціях важко переоцінити. Коли управління ведеться правильно, це допомагає кожній частині бізнесу працювати більш якісно і команді зосередитися на важливій роботі, не відволікаючись від завдань, які збиваються з плану, або бюджетів, які виходять з-під контролю. За визначенням IPM «...управління проєктами – це мистецтво керівництва й координації людських і матеріальних ресурсів протягом життєвого циклу проєкту шляхом застосування системи сучасних методів і техніки управління для досягнення визначених в проєкті результатів за складом і обсягом робіт, вартістю, часом, якістю та задоволеністю учасників проєкту» [2].

Управління проєктами є синтетичним видом діяльності, в якому проглядаються принаймні три групи різних галузей знань. Першу групу відносять до загальноуправлінських. Вони стосуються планування, прогнозування, статистики, логістики, контролю. Другу групу становлять спеціалізовані знання, які стосуються прикладної сфери проєкту, його середовища. Але більшість знань становлять третю групу – унікальні знання (метод критичного шляху, ієрархічна структура робіт, PERT-метод та ін.) [30]. Для успішної реалізації всіх видів діяльності в проєктах призначені і широко використовуються сьогодні, зокрема в галузі ІТ, спеціальні системи управління проєктами (СУП). Це спеціалізовані інформаційні системи, які дозволяють планувати, контролювати та відстежувати прогрес виконання завдань, планування ресурсів та бюджету проєкту, забезпечують ефективну комунікацію між учасниками проєкту, дозволяють реалізувати проєкт у визначені терміни та в межах бюджету.

Завдання СУП полягає в тому, щоб забезпечити ефективне планування, виконання та контроль проєктів різного рівня складності та обсягу. Системи управління проєктами допомагають керівникам проєктів та командам виконавців координувати свої дії, відстежувати прогрес, вирішувати проблеми та досягати поставлених цілей. Вони можуть мати різну структуру, функціональність та інтерфейс, але їх основними компонентами є: модуль планування, модуль виконання, модуль контролю, модуль зв'язку та модуль документування [31]. Отже, спеціальні системи управління проєктами - це програмні продукти, які дозволяють планувати, контролювати та аналізувати роботу над проєктами різного рівня складності та обсягу. Їх характеристиками є:

- адаптація до специфіки проєкту, галузі або організації;
- інтеграція з іншими системами, такими як ERP, CRM, BI тощо;
- підтримка різних методологій управління проєктами, таких як Agile, Waterfall, PRINCE2;
- можливість створення та використання шаблонів, стандартів та кращих практик;
- наявність функцій співпраці, комунікації та зворотного зв'язку між учасниками проєкту;
- забезпечення високого рівня безпеки, надійності та доступності даних.

Розвиток спеціальних систем підтримки проєктної діяльності відбувався паралельно із становленням теорії проєктів (див. розділ 1) з 60-х рр. XX століття. Однією з перших був Microsoft Project. Поява співпала з періодом, коли з'явилися нові найважливіші методології та інструменти управління проєктами, що дозволили краще планувати, контролювати та координувати проєкти.

PERT (англ. Program Evaluation and Review Technique) - техніка оцінки та перегляду програми, яку можна розуміти як інструмент для обробки програм і проєктів, які включають серію заходів. Він підтримує аналіз цих дій і їх послідовне розташування разом із визначенням пропонованої тривалості, необхідної для виконання кожного з цих завдань за допомогою складання критичного шляху, що називається аналізом PERT.

CPM (англ. critical path method) - це метод визначення завдань, необхідних для виконання проекту, та гнучкого планування термінів його реалізації. Визначивши всі дії на побудованому критичному шляху, можна адекватно оцінити майбутні витрати та скласти прогноз термінів проекту. Критичний шлях в управлінні проектами є найдовшою послідовністю робіт, які потрібно виконати, щоб завершити проект у встановлені терміни

WBS (англ. work breakdown structure ) – ієрархічна структура робіт, тобто розбиття проекту на конкретні результати, які мають бути досягнуті для досягнення цілей проекту [32]. На сьогодні існує кілька сотень систем, які мають функції виконання й візуалізації названих методів. У табл. 2.1 наведено приклади використання різних видів ПЗ при управлінні проектною діяльністю.

Таблиця 2.1 – Характеристика методів управління проектом та види ПЗ

Елемент моделі управління проектом	Зміст і характеристика методів в управлінні проектом	Вид ПЗ для виконання робіт
Цілі, ініціація проекту Інструмент – контракт, технічне завдання (ТЗ)	Визначаються вимоги до проекту з огляду на обсяги, витрати, час, якісь, а також визначається, який із них домінує	MS Office, CorelDraw, Draw.oi, Lucidchart, SmartDraw
Що робити? Інструмент - WBS	Визначаються робочі процеси через розробку ієрархічної структури проекту за допомогою блок-схем / діаграм	CorelDraw, Draw.oi, Lucidchart, SmartDraw, Creately, Visio, Cacao, Apple iWork, SureTrak
Хто виконує? (команда) Інструмент - OBS	Призначається керівник і формується команда проекту за допомогою організаційної структури (OBS)	Jira, Trello, Smartsheet, Asana
Хто Що Робить? (призначення) Інструмент – матриця відповідальності	Створюється матриця відповідальності, в якій роботи закріплюються за виконавцями із визначенням міри відповідальності	Asana, Jira, Trello MS Project, Project 365, Project Manager, Wrike i Teamwork
Як: інструменти керування проектами– діаграми Ганта, ресурсні діаграми	Узгоджуються плани виконання проекту щодо встановлених цілей і взаємовідношень робочих елементів	Trello, Jira, MS Project, Asana, Project 365, SureTrak Project Manager, Smartsheet
Час, Витрати (контроль) Інструмент - аналітичні та інформаційні звіти, метод скоригованого бюджету	Визначаються документи, які містять інформацію для контролю щодо термінів, обсягів, бюджету шляхом визначення відхилень від плану	Asana, Jira, Trello Jira, MS Project, Project 365, SureTrak Project Manager

Як видно з табл. 2.1, для виконання головних етапів проєкту можна задіяти один або декілька сучасних видів програмного забезпечення, які будуть використані: для оформлення документів проєкту (MS Office); візуалізації опису і підготовки презентаційних матеріалів, технічного завдання (CorelDraw, Draw.oi, Figma) та інші графічні системи; для формування команди проєкту, постановки і контролю виконання завдань (Asana, Jira, Trello і т. п.)

Згідно аналітичних досліджень, найпопулярнішими системами управління проєктами на початок 2023 р. є Jira і Microsoft Project. Менш популярними є: Asana, Trello, Smartsheet (рис. 2.1).

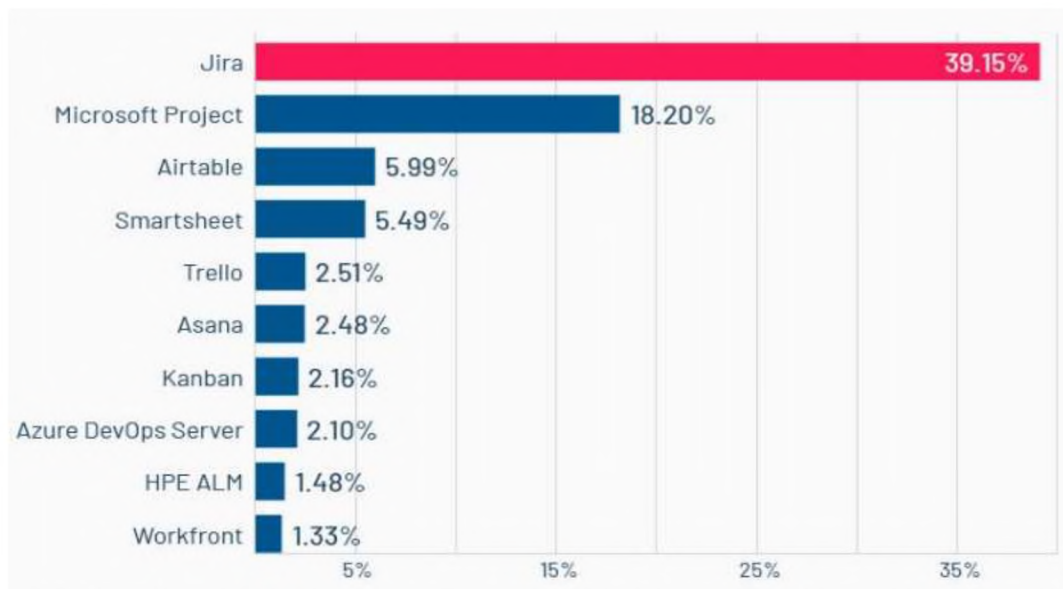


Рисунок 2.1 – Статистика використання систем управління проєктами станом на початок 2023 р. (згідно [33])

Названі системи (див. рис. 2.1) відрізняються за функціоналом, простотою використання, вартістю та відгуками користувачів. При виборі системи управління проєктами важливо враховувати розмір команди, тривалість проєкту, складність завдань і тип методології [34]. Кожна система має свої особливості, тому важливо знайти ту, яка найкраще підходить для вашої команди та ваших проєктів. Кожна СУП за замовченням надає користувачам можливість створювати та редагувати проєкт, календар, а також його наповнення (задачі, розподіл обов'язків тощо). Окрім цього, майже в кожній СУП наявна діаграма

Ганта. Але окрім цих базових функцій, кожна система може виділятися особливим функціоналом, якого немає у конкурентів. Також, СУП відрізняються між собою за складністю та зручністю інтерфейсу для користувачів. Для визначення сильних і слабких сторін та критеріїв вибору таких систем необхідно провести детальний аналіз кожної з них (табл. 2.2) [34, 35].

Таблиця 2.2 – Порівняльний аналіз функціональних можливостей, переваг та слабких сторін популярних систем управління проектами за рейтингом

Назва СУП	Особливості функціональних можливостей, сильні сторони	Недоліки (слабкі сторони)
Jira (Atlassian)	Хмарний сервіс управління проектами з великою кількістю інструментів: проекти (задачі, помилки, запити), робочий процес (задачі, контроль). Має додатково: створення backlog, розробка roadmap, розподіл і коментування задач, Канбан, аналіз продуктивності співробітників, легко інтегрується з іншими, підтримує методологію Agile	Жіга доволі складна і заплутана через велику кількість вбудованих функцій. Ціна (стандарт) \$7,50 за користувача на місяць
Microsoft Project (Microsoft)	Створення та управління зв'язками між задачами, управління ресурсами, можливість встановлення віх та критичних задач, управління вартістю та бюджетом, формування звітів, діаграма Ганта, налаштування різних календарів і варіантів представлення проектів. Це найбільш комплексна СУП із представлених. Найкраще підходить для складних проектів	MS Project може бути дещо складним через велику кількість функцій. Мало підходить для гнучкої методології, не хмарний. Цін за 1 встановлення \$51,2
Asana	Таск-менеджер, виділяється системою структурування проекту шляхом розбиття на папки та розділи (конструктор процесу), має діаграму Ганта, Канбан-дошки і відстежування прогресу в проекті, календар, інтеграцію додатків (Dropbox, Google Drive, Adobe Creative Cloud). Asana веб-версію, мобільний додаток	Складний інтерфейс, обмежений експорт лише у форматах JSON або CSV, планування ресурсів, Ціна \$10,99 за користувача на місяць
Trello (Atlassian)	Простий хмарний сервіс для невеликих проектів і команд. Створення Канбан-дошок для проектів та розподілення карток за списками (lists) відповідно до етапів виконання, створення карток з описом завдань та відповідальними за їх виконання користувачами Trello має дружній до користувача інтерфейс, змінювати дизайн. (необхідно встановлювати плагін)	Відсутня діаграма Ганта за замовчування, обмежене хмарне сховище 10 Мб, відсутні функції управління завданнями, ресурсами, фінансами. Ціна \$5 за користувача на місяць
Worksection (Україна)	Хмарний сервіс для систематизації проектних даних, сховище даних; систематизація робочих процесів; контроль термінів; комунікація; тайм-трекінг; діаграма Ганта; побудова звітів; дошка Канбан, підтримка гнучких методологій. Наявна мобільна версія	Відсутній спільний чат, слабке планування ресурсів, без мобільного додатку. Ціна \$49 на місяць за команду

Відповідно до даних табл. 2.2 можемо зробити висновок, що з розглянутих систем управління проектами кожна має свої переваги та недоліки. Jira та MS Project є досить потужними та функціональними системами, які дозволяють детально планувати та контролювати проекти, але вони можуть бути складними для оволодіння та вимагати більше часу для їх використання. Висока популярність є цілком обґрунтованою. Найбільша різниця – в методологіях проектів. MS Project ідеально працює з каскадною моделлю в тривалих проектах. Jira підійде для складних проектів за гнучкою моделлю, в якій зазвичай важко вдається візуалізувати стан роботи команди, спринти [36]. Але Jira спеціально передбачає саме такі елементи інтерфейсу та інструменти (рис. 2.2).

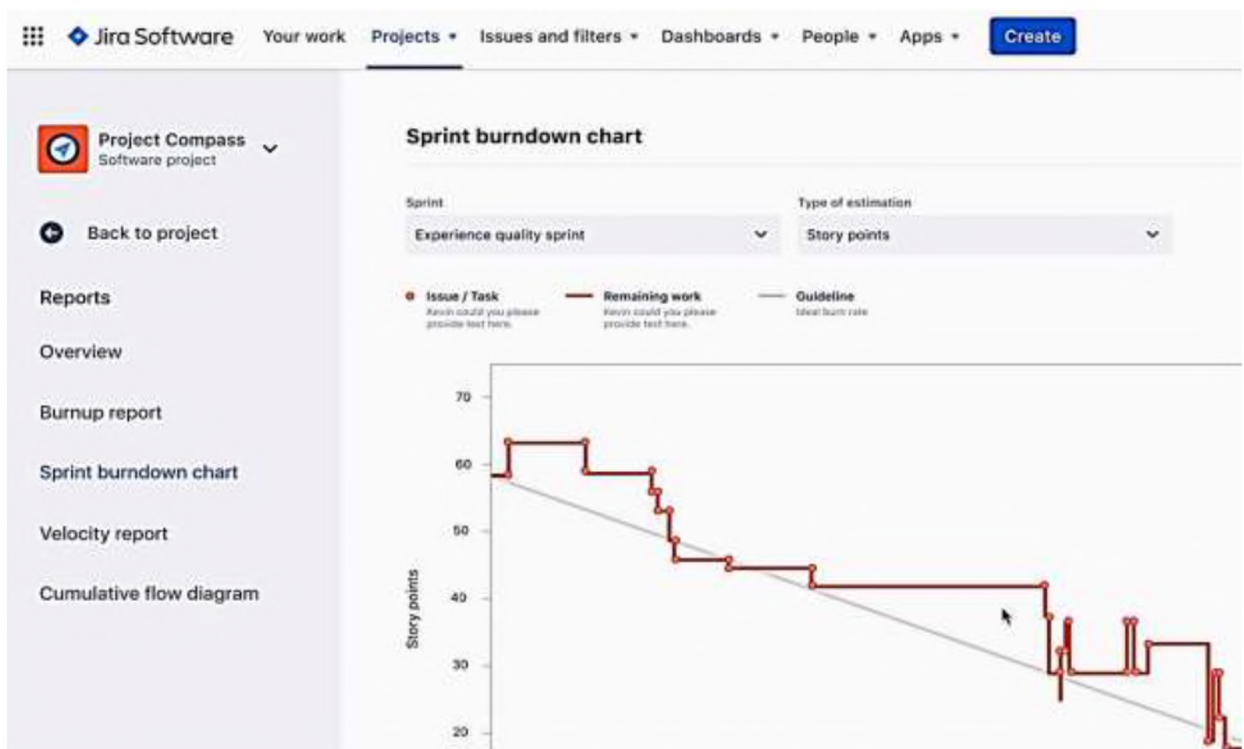


Рисунок 2.2 – Фрагмент інтерфейсу Jira при роботі зі спринтом

Jira найчастіше використовується для управління життєвим циклом додатків, управління тестами та проектами. Конкретні завдання часто включають відстеження помилок і проблем, а також створення історій користувача. У Jira можна керувати процесом розробки від ідеї до запуску готового продукту. Окрім IT-команд, її використовують маркетологи, аналітики, тестувальники та інші спеціалісти.

Основні напрями робіт, для яких може знадобитися Jira, є такі:

– Управління вимогами. Вимоги – це вступні дані для роботи над проектом. Їх пишуть в окремому документі разом із замовником, щоб не виникало розбіжностей у процесі роботи, а розробники могли на них орієнтуватися.

– Управління продуктами. Команди складають у Jira дорожні карти – покрокові плани масштабних проектів. Такі карти допомагають налагодити взаємодію між відділами. Наприклад, при грамотному управлінні маркетингологи можуть планувати промокампанію паралельно з розробкою, а не чекати на готовий продукт. У дорожніх картах розставляються цілі, пріоритети та позначаються залежності роботи одного відділу від роботи іншого.

– Управління проектами. Jira налаштовується під проекти, в ній можна візуально відстежити шлях кожного завдання від створення до результату. Величезна кількість команд по всьому світу використовують розгортання Jira Cloud для управління проектами, розробки ПЗ, DevOps і не тільки.

Asana, Worksection та Trello мають менше функціональності (відсутні деякі базові функції), але вони базуються на дошках та картках, що дозволяє організувати проекти у вигляді зручних списків завдань (рис. 2.3), є зручними для управління за методологією Agile, для невеликих команд [35].

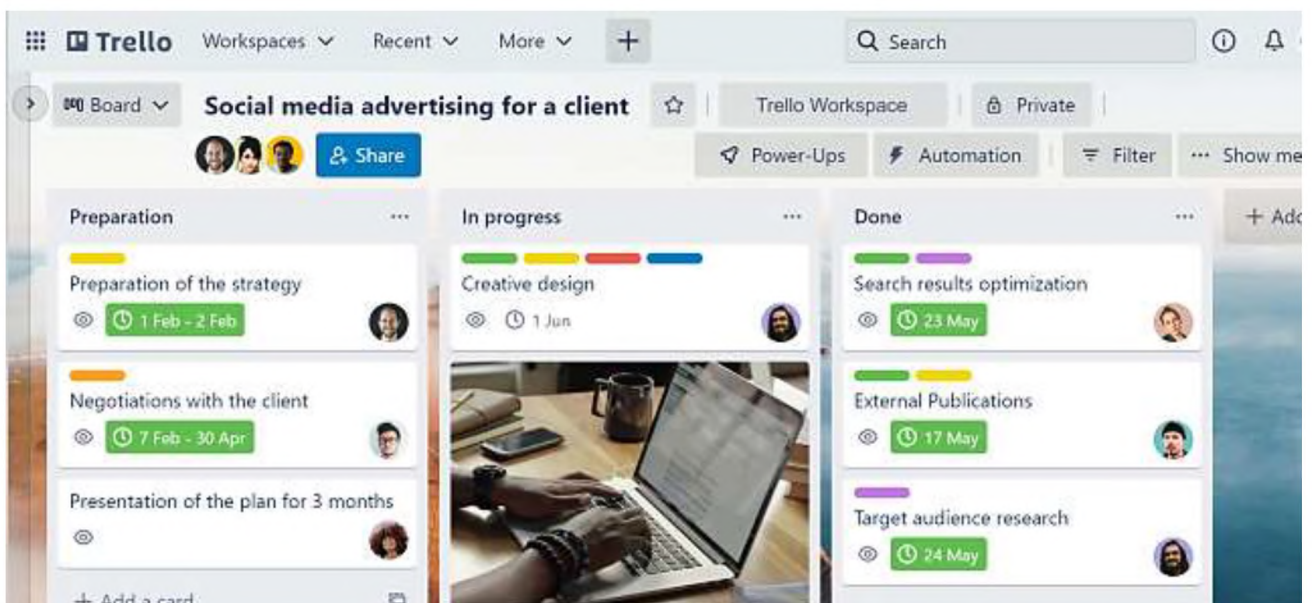


Рисунок 2.3 – Фрагмент інтерфейсу Канбан-дошки Trello

MS Project є, можливо, найпопулярніший серед численних інструментів управління великими проектами з використанням діаграм Ганта та інших методів, як то управління і візуалізація ресурсів, оцінювання робіт за графіками та формами, календарем тощо [37]. Цей продукт є частиною офісної екосистеми Microsoft і посідає ключове місце у виборі систем управління проектами. Але має і суттєві обмеження: відсутність можливості розподіленої роботи з коробки. Тобто, будучи десктопним standalone-додатком, MS Project розрахований лише на роботу одного користувача в один момент часу. Щоб організувати паралельну роботу, доведеться купувати ліцензію Microsoft Project Server, Project for the web або Microsoft Planner. Найчастіше використовується для великих проектів

Серед альтернатив із open source можна розглянути ProjectLibre, GanttProject та TaskJuggler (безкоштовні десктопні програми для індивідуального користування, без можливості розподіленої роботи); Projector та Redmine (вебдодатки для розподіленої командної роботи) та Project-open (вебдодаток для управління проектами й процесами компанії в цілому). Ці інструменти обрані не випадково, вони відповідають вимогам, які можна сформулювати так [37]:

- наявність редактора діаграм Ганта;
- кросплатформність - Windows, Linux і macOS;
- масштабованість;
- функції імпорту/експорту;
- генерація звітів.

Організувати управління проекту можна через Jira. Дана ІС входить до сімейства продуктів (Jira Software), розроблених з метою спростити управління робочим процесом для різних команд. Сьогодні Jira – це потужний інструмент управління роботою, що підходить для різних випадків, від управління вимогами і сценаріями тестування до agile-розробки програмного забезпечення

Отже, існує широкий спектр програмного забезпечення, які розроблене спеціально для фахівців області управління ІТ-проектами з метою підвищити ефективність команд і досягнення найоптимальнішим способом поставлених

цілей проєктів. Оскільки вибір системи управління проєктами залежить від потреб конкретного проєкту та команди, важливо попередньо ретельно проаналізувати можливості та вимоги, щоб визначити, яка система буде найбільш ефективною для його управління.

## **2.2 Розробка концепції та плану реалізації проєкту впровадження інформаційної системи на тестовому підприємстві**

В роботі для прикладу використання програмної підтримки для проєктів ІС розглянута поетапна реалізація проєкту впровадження автоматизованої інформаційної системи на умовному підприємстві – тестовий проєкт впровадження системи класу ERP. Для конкретизації завдань та окремих технічних умов розглядається програмний комплекс (ПК) «Універсал 9», який був обраний за результатами порівнянь із іншими системами такого ж класу та вивчення потреб споживачів. ПК «Універсал 9» - потужне багаторівневе рішення класу ERP для автоматизації бізнес-процесів середнього та великого бізнесу, підприємств, організацій [38].

При адаптації модулів системи для конкретних підприємств відбувається не лише створення моделі бази даних підприємства, але й доопрацювання на рівні програмних кодів. Тому, впровадження такої системи розглядається як повномасштабний проєкт, тривалість якого складає більше року. Такий проєкт має значний бюджет, таким чином, і СУП потрібно обрати у відповідності до вимог. Окрім того, необхідно протестувати й забезпечення командної роботи, можливість швидких комунікацій із підприємством-замовником. Початком, або ініціацією ІТ-проєкту на підприємстві можна вважати рішення про впровадження автоматизованої інформаційної системи і обрання, наприклад системи «Універсал 9» класу ERP. Таке рішення, як правило, ґрунтується на аналізі власних потреб і можливостей, прагненні інновацій і розвитку і

підкріплюється ознайомленням із різними видами подібних систем, побудові систем переваг, оцінці фінансової спроможності тощо.

Перехід на нову ERP-систему може виконуватися з використанням проектної або стандартної технології впровадження. В даній роботі надалі розглядається варіант проектного впровадження на умовному підприємстві, яке передбачає сумісне із виконавцем передпроектне вивчення стану управління інформаційним процесами, існуючих баз даних, наявного технічного забезпечення, розробку технічного завдання, тим самим зменшуючи ризики впровадження та отримання прогнозованого і задуманого результату. Для успішної реалізації проекту впровадження ERP готується його опис із зазначенням цілей, завдань, очікуваних результатів та необхідного ресурсного й фінансового забезпечення. Опис проекту, коротка версія якого представлена в додатку А, зазвичай є результатом сумісної роботи групи у складі провідних фахівців організації-замовника, представників компанії ТОВ «СофтПро», консультантів [39].

Вибору проектного рішення із досвіду компанії-розробника сприяють особливості підприємства (організації), при яких рекомендовано саме створення проекту. Серед факторів сприяння проектному рішенню називають:

1) складність і масштаб діяльності підприємства та завдання автоматизації, що охоплює значну кількість контурів управлінського та бухгалтерського обліків (виробництво, торгівля, склад, закупівлі, бюджетування, персонал і т. ін.);

2) необхідність контролю за термінами впровадження і розмір виділеного бюджету;

3) передбачається автоматизація повного набору бізнес-процесів організації;

4) на початок проекту існує бачення, як має працювати інформаційна система, однак відсутні чіткі вимоги до елементів системи;

5) на момент впровадження немає іншого галузевого рішення;

6) при реалізації проекту потрібно відстежувати хід виконання завдань;

- 7) необхідно відстежувати освоєння бюджету на етапах проєкту;
- 8) потреба мінімізувати втрати при переході на нову систему;
- 9) необхідність ресурсного планування (фахівці, час, матеріальне забезпечення, комунікації);
- 10) розташування об'єктів автоматизації територіально розосереджене;
- 11) автоматизації підлягають кілька юридичних осіб із різними видами рахунків;
- 12) планується в подальшому інтегрованість з іншими системами (електронний документообіг для місцевого самоврядування, CRM або інші).

Для планування та реалізації проєкту впровадження ІС класу ERP ПК «Універсал» пропонується (рекомендовано розробниками [40]) обрати чотирьохфазну модель ЖЦ, яка включає етапи, зміст яких наведено на рис. 2.4.



Рисунок 2.4– Етапи життєвого циклу проєкту впровадження ПК «Універсал»

У будь-якій системі управління повинен бути формалізований її об'єкт і побудовані алгоритми самого управління. На першому етапі (див. рис. 2.4) це відбувається як алгоритмічний опис пов'язаних із цими процесами потоків інформації, її взаємозв'язків, бізнес-правил і критеріїв їх оцінки. Проєктується не тільки документообіг, але й ергономіка системи, інтерфейс користувача. Разом із представниками замовника вирішується питання «як має бути» і створюються технічне завдання (ТЗ) проєкту. Важливу увагу приділяють також побудові архітектури системи на основі базової клієнт-серверної архітектури.

На другому етапі відбувається виявлення недоліків, які допущені у формуванні ТЗ, а також проводиться моделювання працездатності системи в різних виробничих ситуаціях. Це дозволяє підготуватися до запуску і використання.

На третьому етапі відбувається перенесення інформації, яка вже існує на підприємстві, є певним чином формалізована, використовувалась у попередніх системах. Максимальна увага приділяється збереженню даних. Надалі вводиться нова оперативна інформація безпосередньо на платформу «Універсал». Цей етап досить напружений, але не надто тривалий за оцінками фахівців.

На четвертому етапі участь компанії-розробника (або виконавця проєкту) зменшується, увага зосереджується на технічній підтримці, оновленні системи в майбутньому, навчанні персоналу. Виконання проєкту протягом усіх етапів супроводжується використанням відповідного програмного забезпечення, яке призначене для підтримки ефективності проєктної діяльності – СУП.

Побудову концепції проєкту впровадження автоматизованої інформаційної системи на підприємстві доцільно розглянути на основі практичного кейсу, який має всі суттєві ознаки реальної ситуації та моделює дії компанії, що займається реалізацією таких проєктів.

Пріоритетами таких проєктів є фінансова цінність, стратегічна цінність, рівень ризиків. Висока фінансова цінність передбачає окупність проєкту до 1 року, очікувані доходи перевищують витрати не менше, ніж у 1,5 рази.

### **2.3 Особливості формування WBS проєкту в різних системах управління проєктами**

WBS – не єдиний, але найефективніший спосіб наочно відобразити весь обсяг проєкту. WBS фокусує увагу не на процесі, а на очікуваному результаті. В ідеалі у розробці WBS бере участь замовник або його представник і вся команда виконавця.

За допомогою WBS можна наочно обґрунтувати потреби у фінансах чи людських ресурсах, запобігти ризикам і змінам або принаймні значно (дуже значно!) знизити їх ймовірність і вплив, оскільки саме тут спливають багато неочевидних раніше речей і «а ми хотіли зовсім інше» (і так і має бути, для цього інструмент і призначений).

На рівні WBS вже можна визначити та узгодити контрольні точки проекту (як для рішень щодо продовження проекту після чергового етапу, так і для контролю витрат людських та фінансових ресурсів).

Маючи уявлення про зміст етапів життєвого циклу проекту (див. рис. 2.4) і розроблену концепцію, можна сформувавши перелік основних видів робіт (позначено літерами) у проектах такого типу, до якого, зазвичай, включають:

А. Обстеження об'єкту автоматизації.

Цей етап включає в себе вивчення технічного стану об'єкту, аналіз функцій та процесів, які повинні бути автоматизовані.

В. Розробка технічного завдання на створення автоматизованої системи.

Після завершення обстеження об'єкту, необхідно розробити технічне завдання (ТЗ) на створення автоматизованої системи. У цьому етапі визначаються технічні та функціональні вимоги до системи, які повинні бути відповідним чином документовані.

С. Узгодження з керівництвом компанії-замовника змісту ТЗ і можливе внесення коректив.

На цьому етапі замовник може внести свої вимоги та побажання щодо системи. Результатом цього етапу є остаточне затвердження ТЗ.

Д. Розробка за узгодженим техзавданням детального плану робіт.

Після того, як технічне завдання затверджене, розпочинається розробка детального плану робіт, який включає визначення термінів виконання робіт, виділення необхідних ресурсів та визначення відповідальних за їх виконання.

Е. Розробка специфікації апаратної (технічної) частини ERP.

У цій фазі команда розробників визначає апаратні засоби, які будуть використовуватися для створення автоматизованої інформаційної системи

(AIC). Крім того, в цій фазі розробляється технічна документація, яка описує технічні характеристики та вимоги до обладнання, яке буде використовуватися для створення AIC. Команда розробників повинна врахувати вимоги до безпеки і ефективності роботи системи, щоб забезпечити високу якість проєкту.

#### F. Поставка і встановлення контурів AIC.

На цьому етапі розпочинається поставка необхідного обладнання та програмного забезпечення для встановлення AIC. Після того, як обладнання буде доставлено на місце, фахівці займаються встановленням необхідних компонентів та деталей AIC.

#### G. Тестова експлуатація системи (тестування).

Після встановлення системи ERP проводиться її тестова експлуатація з метою перевірки функціонування та відповідності вимогам, які були визначені в технічному завданні. На цьому етапі проводяться різні види тестів, включаючи модульні, інтеграційні та системні тести, щоб перевірити правильність роботи ERP в різних умовах та в різних сценаріях.

#### H. Навчання персоналу для роботи з новою системою.

Навчання персоналу, який буде працювати з AIC (ERP), проводиться з метою надання їм достатньої кваліфікації та знань, необхідних для успішної роботи з новою системою. Навчання може проводитись в режимі онлайн або офлайн, в залежності від потреб клієнта та специфіки системи.

#### I. Розробка і оформлення технічної документації:

Команда розробників готує технічну документацію, яка містить детальний опис функцій і вимог до ERP. Ця документація включає у себе технічні креслення, опис функцій та інструкції зі зборки і налаштування системи. Розробляється інструкція з експлуатації, про використання системи.

#### J. Підписання акту по введенню в експлуатацію ERP.

Після завершення робіт зі створення ERP, компанія із замовником підписує акт, який підтверджує введення системи в експлуатацію.

Наступними етапами роботи є створення послідовності робіт в програмі MS Project або її аналогах. Для подальшого опису та управління планом проєкту

серед проаналізованих систем обрана система MS Project, яка має необхідний функціонал та відповідає потребам роботи з тривалими та складними проєктами за класичними каскадною (інкрементальною) моделями.

Програма MS Project передбачає підключення автоматизованого або ручного календарного планування операцій, визначення та коригування періодів робіт, визначення критичного шляху термінів виконання проєкту (СРМ-метод) від самого початку внесення первинних даних. Для цього в середовищі програми необхідно ввести найменування, тривалість кожної з робіт, розраховуючи від дати початку проєкту. Дату початку проєкту встановлюють першою в розділі про проєкти: можна обрати від моменту ініціалізації проєкту і першого обговорення із замовником. Ця дата не є поточною і надалі незмінна.

При уведенні переліку задач по проєкту вказується дата початку і тривалість, надалі планування здійснюється автоматично, використовується календар (рис. 2.5).

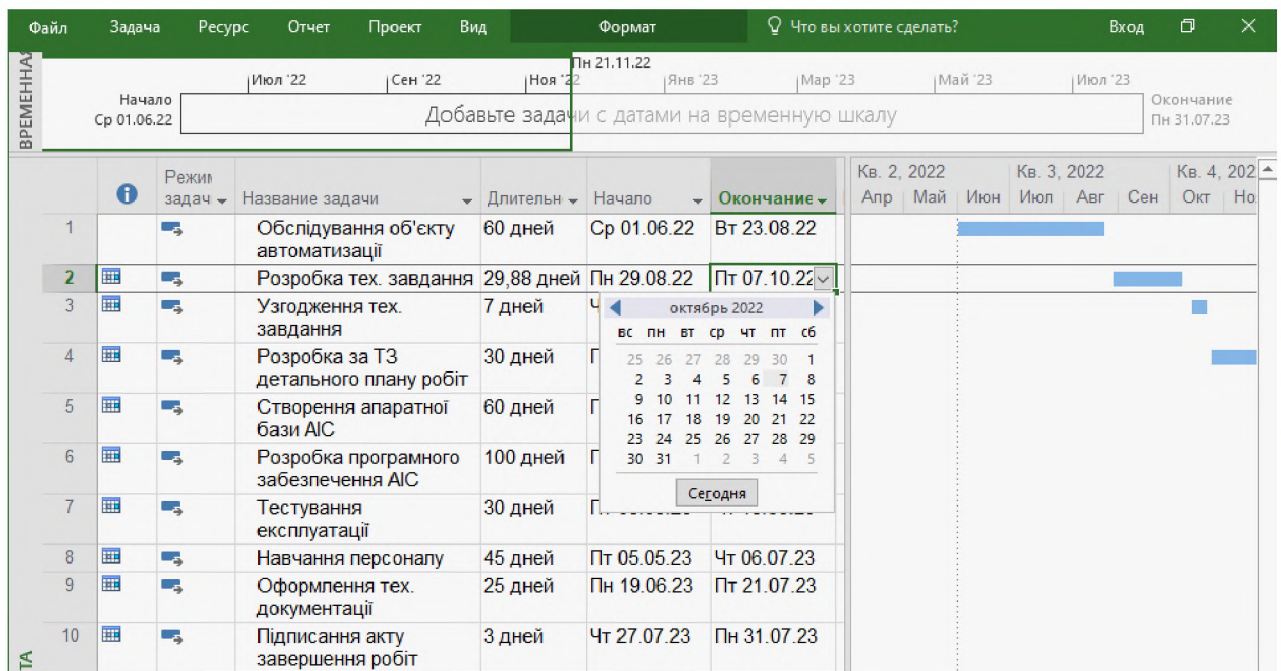


Рисунок 2.5 – Внесення переліку виконання завдань по проєкту

Спеціальні форми відомостей про кожну задачу заповнюються окремо, де встановлюється тип і попередники, а також додатковий резерв часу (рис. 2.6).

№	Режи задач	Название задачи	Длительность	Начало	Окончание
1		Обслуговування об'єкту автоматизації	60 дней	Ср 01.06.22	Вт 23.08.22
2		Розробка тех. завдання	30 дней	Пн 29.08.22	Пт 07.10.22
3		Узгодження тех. завдання	7 дней	Чт 13.10.22	Пт 21.10.22
4		Розробка за ТЗ детального плану робіт	30 дней	Пн 24.10.22	Пт 02.12.22
5		Створення апаратної бази АІС	60 дней	Пн 12.12.22	Пт 03.03.23

Сведения о задаче

Общие | Предшественники | Ресурсы | Дополнительно | Заметки | Настраиваемые поля

Название: Створення апаратної бази АІС Длительность: 60 дней  Предв. оценка

Процент завершения: 0%

Режим планирования:  Вручную  Автоматически

Приоритет: 500  Неактивные

Даты

Начало: Пн 12.12.22 Окончание: Пт 03.03.23

Показать на временной шкале

Скрыть отрезок

Рисунок 2.6 – Приклад заповнення форми про поточну задачу і її попередника

Важливо чітко розуміти види робіт, які можуть виконуватись одночасно (паралельно) і врахувати це в проєкті. Форма має вкладку (рис. 2.7), на якій автоматично заповнюються найменування задачі-попередника, вид попередника (ОН – Кінець-Початок або НН-Початок-Початок), а також резерв часу у колонці. Для задач, які розпочинаються одночасно, вводимо тип НН. Вони розпочинатимуться в одну дату і мають бути завершені до початку наступної задачі. Результат опису задач представлено на рис. 2.7.

№	Ид.	Название задачи	Длительность	Начало	Окончание	Тип	Запаздывание
6		Розробка програмного забезпечення АІС	100 дней	Ср 14.12.22	Вт 02.05.23	4ОН+5 дней	
7		Тестування експлуатації	30 дней	Пн 08.05.23	Пт 16.06.23	6ОН+3 дней	
8		Навчання персоналу	45 дней	Пт 05.05.23	Чт 06.07.23		

Сведения о задаче

Общие | Предшественники | Ресурсы | Дополнительно | Заметки | Настраиваемые поля

Название: Навчання персоналу Длительность: 45 дней  Предв. оценка

Предшественники:

Ид.	Название задачи	Тип	Запаздывание
5	Створення апаратної бази АІС	Начало-начало (НН)	3д

Тип:

Окончание-начало (ОН)

Начало-начало (НН)

Окончание-окончание (ОО)

Начало-окончание (НО)

(Нет)

Рисунок 2.7 – Способи налаштування типів і порядку виконання задач

Головним результатом підтримки програми MS Project є можливість фіксації паралельних робіт і відображення їх на діаграмі Ганта, можливість коригування тривалості робіт і часу запізнення. Календар дозволяє розрахувати критичний шлях виконання проєкту від раннього початку до пізнього завершення, побудувати первинний звіт за проєктом.

У переважній більшості випадків операції в проєктах не є рівноправними. Найчастіше можна виділити складені операції, які розділяються на більш прості роботи. При цьому така вкладеність має декілька рівнів глибини. Можливості по задаванню ієрархічної структури операцій, що входять в проєкт, реалізовані в MS Project шляхом додавання рядків із новими задачами (рис. 2.8) та надання їм рівня вкладеності.

	Режи задач	Назва задачи	Длительность	Начало	Окончание	Предшественники
1		Обслідування об'єкту автоматизації	60 днів	Ср 01.06.22	Вт 23.08.22	
2		Розробка тех. завдання	30 днів	Пн 29.08.22	Пт 07.10.22	1ОН+3 днів
3		Узгодження тех. завдання	7 днів	Чт 13.10.22	Пт 21.10.22	2ОН+3 днів
4		Розробка за ТЗ детального плану робіт	30 днів	Ср 26.10.22	Вт 06.12.22	3ОН+2 днів
5		Створення апаратної бази АІС	60 днів	Ср 14.12.22	Вт 07.03.23	4ОН+5 днів
6		Розробка програмного забезпечення АІС	100 днів	Ср 14.12.22	Вт 02.05.23	4ОН+5 днів
7		Тестування експлуатації			Пт 16.06.23	6ОН+3 днів
8		<Новая задача>				
9		<Новая задача>				
10		<Новая задача>				
11		Навчання персоналу			т 06.07.23	5ОН+3 днів
12		Оформлення тех. документації			т 21.07.23	6ОН+2 днів
13		Підписання акту завершення робіт			н 31.07.23	6ОН+2 днів

Рисунок 2.8 – Спосіб додавання нових полів для задач через контекстне меню

Зазвичай, у проєктах не рекомендується робити більше 3-5 рівнів. Наприклад, операцію «Тестова експлуатація системи (тестування)», на пропозицію фахівців компанії краще розділити на декілька складових (підетапів) (рис. 2.9):

	Режи задач	Название задачи	Длитель	Начало	Окончание	Предшественн
6		Розробка програмного забезпечення АІС	100 дней	Ср 14.12.22	Вт 02.05.23	4ОН+5 дней
7		Тестування експлуатації	30 дней	Пн 08.05.23	Пт 16.06.23	6ОН+3 дней
8		Тестування апаратного забезпечення	10 дней	Вт 09.05.23	Пн 22.05.23	
9		тестування обчислювальної техніки	4 дней	Вт 09.05.23	Пт 12.05.23	
10		тестування комунікаційного обладнання	6 дней	Вт 09.05.23	Вт 16.05.23	
11		Роздільне тестування підсистеми програмного забезпечення	10 дней	Пт 19.05.23	Чт 01.06.23	
12		Комплексне тестування автоматизованої системи	10 дней	Чт 01.06.23	Ср 14.06.23	
13		Навчання персоналу	45 дней	Пт 05.05.23	Чт 06.07.23	5НН+3 дней
14		Оформлення тех. документації	25 дней	Пн 19.06.23	Пт 21.07.23	6ОН+2 дней
15		Підписання акту	3 дней	Чт 27.07.23	Пн 31.07.23	6ОН+2 дней

Рисунок 2.9 - Створення підпорядкованих задач та задавання їх рівня

На місце доданих нових задач вводяться назви підпорядкованих операцій і понижується їхній рівень (рис. 2.10). При цьому у вищій по ієрархії задачі з'являється характерна стрілка, яка може приховувати нижчий рівень.

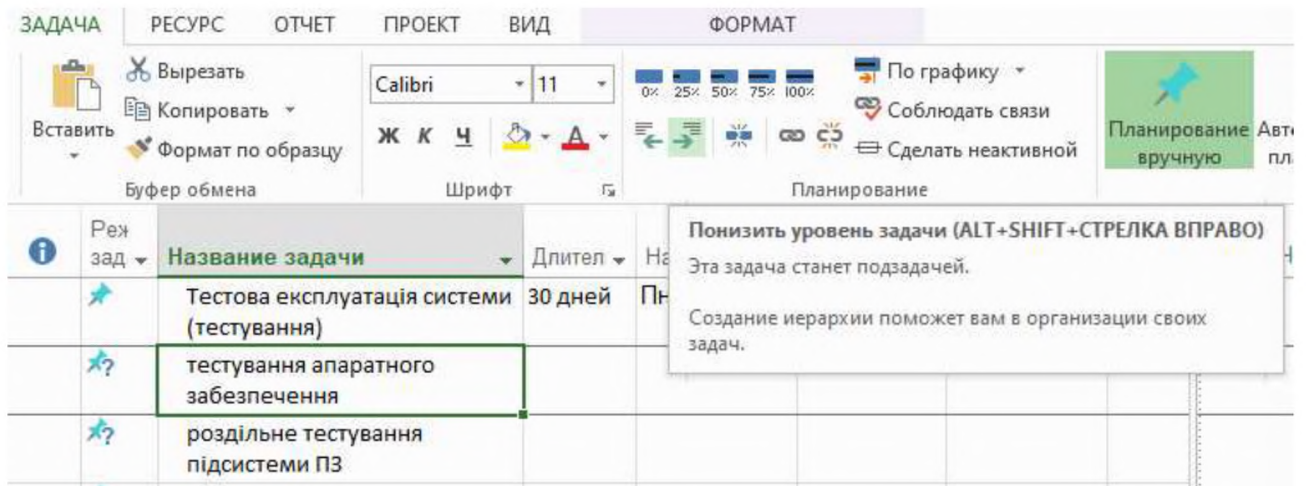


Рисунок 2.10 – Команда задавання ієрархії підпорядкованих задач

Всі підпорядковані операції виконуються послідовно, при цьому попередників можна вказати за номерами, а деталі автоматично з'являються у формі опису задач, що значно полегшує внесення даних по проекту.

Окрім задавання логічних зв'язків між операціями, окремі, ключові операції, можна прив'язати в MS Project до конкретних моментів часу. Для того щоб здійснити прив'язку поточної операції, необхідно просто клацнути мишею по потрібній задачі в переліку, у вікні діалогу вибрати вкладку «Дополнительно», змінити значення полів, що входять в групу «Ограничение задачи»: «Тип ограничения», «Крайний срок» (до якого виробляється прив'язка) (рис. 2.11).

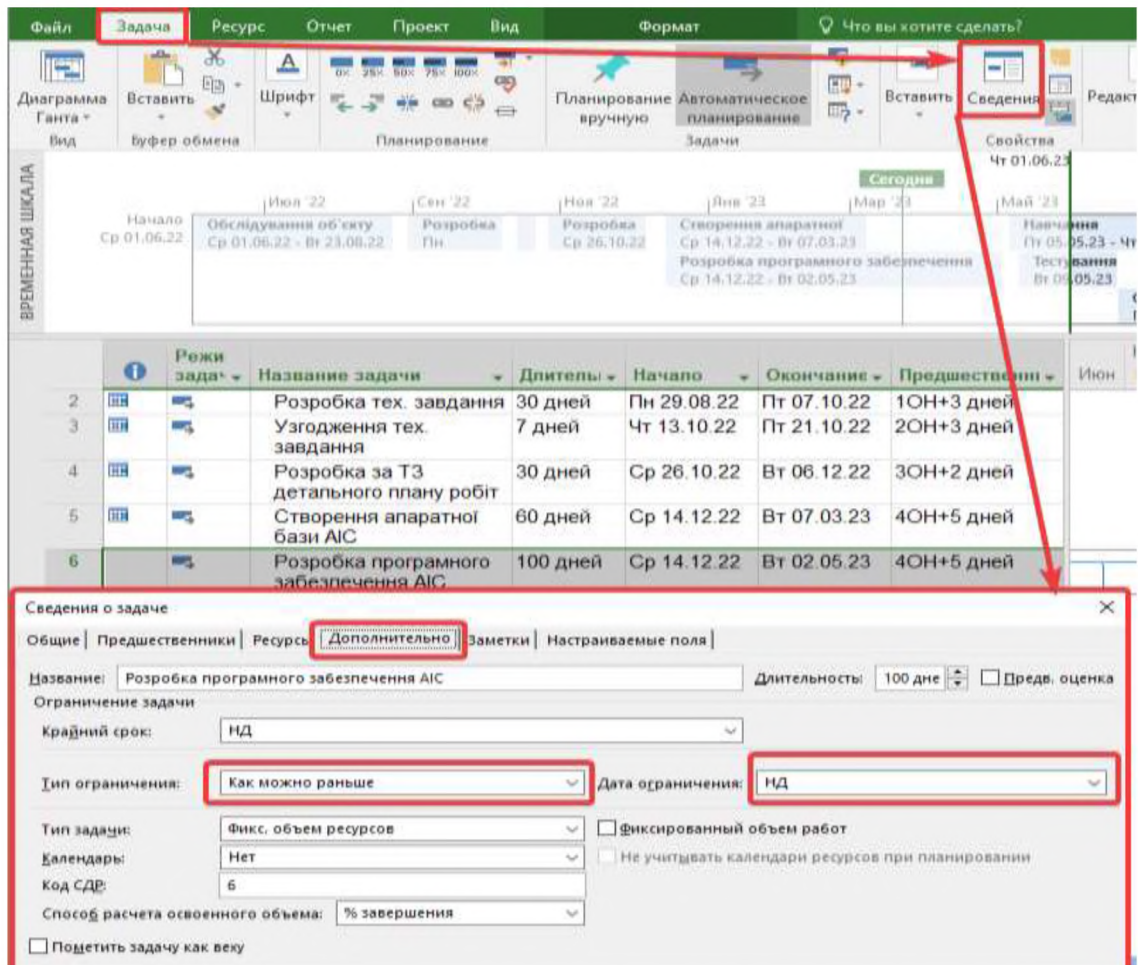


Рисунок 2.11 - Вигляд вікна вмісту додаткових відомостей про задачу

При встановленні календарних термінів для вирішення критичних питань активується майстер планування, який дозволяє прийняти поправки із можливих варіантів У програмі можна змінити значення полів, що входять у групу обмежень операції (тип обмеження та дата – момент часу, до якого задача має бути завершена). Наприклад, на рис. 2.12 показано, як встановлені обмеження для задачі «Обстеження об'єкту автоматизації» (тривалість, дата початку).

Режи задач	Название задачи	Длительность	Начало	Окончание	Предшественники	Кв. 3, 2023
1	Обследования объекта автоматизации	60 дней	Ср 01.06.22	Вт 23.08.22		Июн Июл Авг
2	Сведения о задаче					
3	Общие   Предшественники   Ресурсы   Дополнительно   Заметки   Настраиваемые поля					
4	Название:	Обследования объекта автоматизации	Длительность:	60 дней	<input type="checkbox"/> Предв. оценка	
5	Ограничение задачи					
6	Крайний срок:	НД				
7	Тип ограничения:	Начало не ранее	Дата ограничения:	Ср 01.06.22		

Рисунок 2.12 – Послідовність операцій по часу

Таким чином, етап ініціалізації та обговорення проекту після визначення переліку і тривалості, а також послідовності робіт, встановлення ієрархії окремих операцій завершений і має візуальне та програмне втілення в середовищі MS Project.

Для порівняння було проведене WBS-планування в середовищі відомої СУП Wrike [41], яка також передбачає керівництво проектами різних видів і складності (рис. 2.13).

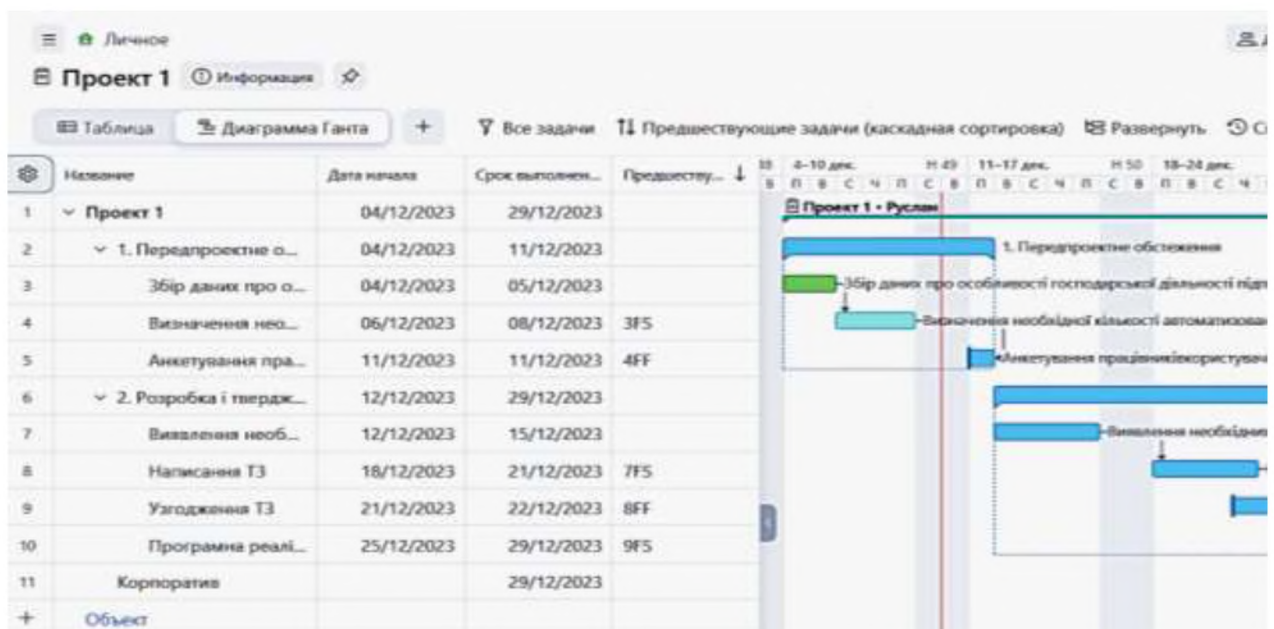


Рисунок 2.13 – Приклад альтернативного плану робіт в середовищі Wrike

Варто відмітити, що програма дійсно має чимало зручних налаштувань, сучасно виглядає, передбачає необхідний інструментарій візуалізації (діаграма Ганта).

Так само дозволяє показати архітектуру задач, позначити порядок виконання через встановлення попередників, а також коментарі на діаграмі Ганта. Конкурує з MS Project.

## **Висновки до розділу 2**

Одним із головних видів інформаційної підтримки реалізації проєктів є системи управління проєктами - спеціалізовані інформаційні системи, які дозволяють планувати, контролювати та відстежувати прогрес виконання завдань, планування ресурсів та бюджету проєкту, забезпечують ефективну комунікацію між учасниками проєкту, сприяють реалізації проєктів у визначені терміни та в межах бюджету. Для виконання проєктів різних моделей та видів можуть бути застосовані різні системи управління проєктами. Порівняльний аналіз функціональних можливостей найбільш рейтингових систем (Jira, Microsoft Project, Asana, Trello, Worksection, Wrike) показав, що всі системи мають сильні і слабкі сторони. Jira, Wrike та MS Project є досить потужними та функціональними системами, які дозволяють детально планувати та контролювати проєкти. Однак, не всі мають засоби підтримки командної роботи.

Для прикладу розглянуто процес планування робіт за проєктом впровадження системи класу ERP на умовному підприємстві в середовищі Microsoft Project та Wrike з демонстрацією головних елементів функціоналу. Прототипом задачі слугували практичні кейси компанії-розробника ПК «Універсал» - ТОВ «Софт ПРО». За основу взято чотирифазну модель ЖЦ впровадження, створено концепцію проєкту, складено WBS в середовищі систем Microsoft Project та Wrike.

Головним результатом підтримки програм MS Project і Wrike є можливість фіксації паралельних робіт і відображення їх на діаграмі Ганта, коригування тривалості робіт і часу запізнення. Календар дозволяє розрахувати критичний шлях виконання проєкту від раннього початку до пізнього завершення, побудувати первинний звіт.

## РОЗДІЛ 3

### ПРАКТИЧНІ АСПЕКТИ ЗАСТОСУВАННЯ СИСТЕМ УПРАВЛІННЯ ПРОЄКТАМИ НА ПРИКЛАДІ ПРОЄКТУ ВПРОВАДЖЕННЯ ERP- СИСТЕМИ

#### 3.1 Технологія планування ресурсного забезпечення задач проєкту в середовищі MS Project

Більшість проєктів, окрім планування графіку й послідовності виконання робіт за допомогою спеціалізованої інформаційної системи, потребують автоматизації планування й розподілу ресурсів за задачами, які необхідні для його успішного виконання. Наприклад, у відповідності до обраного прикладу в якості таких ресурсів можуть розглядатися обчислювальна техніка, команда спеціалістів: аналітики, адміністратори ІС, системні програмісти, тестувальники, інженери і т. ін. Зміст основних реурсів, які викоистовують у проєктах інформаційних систем, наведено в табл. 3.1.

Таблиця 3.1 – Види ресурсів проєктів ІС та їхня характеристика

Група ресурсу	Характеристика, властивості ресурсу	Тип ресурсу
Людські ресурси	Всі спеціалісти, які беруть участь у розробці, впровадженні та підтримці ІТ-проєкту -команда	Відновлювальні (робочий час працівників, основні фонди та ін.)
Інформаційні ресурси -	Дані, знання, документація та інша інформація, яка використовується для планування, аналізу, проєктування, тестування та управління ІТ-проєктом.	
Матеріальні ресурси -	Обладнання, програмне забезпечення, мережева інфраструктура та інші фізичні засоби, необхідні для реалізації ІТ-проєкту.	Невідновлювальні (витратні)
Фінансові ресурси	Грошові кошти, які виділяються на фінансування ІТ-проєкту	

Для прикладу, в MS Project передбачена робота з відновлювальними ресурсами і лише з одним видом невідновлювальних ресурсів – фінансами, тобто передбачається, що будь-які види невідновлювальних ресурсів можуть бути зведені до грошового еквіваленту. Особливим видом відновлювальних ресурсів

є трудові, обсяг яких обраховується у відсотках. Наприклад, робота одного фахівця протягом повного робочого часу встановлюється як 100 %. Якщо працюють 2 таких спеціаліста, то потреба записується як 200 %. При цьому вартість такого ресурсу обчислюється в грошовому вимірі заробітної плати (погодинно, подово залежно від типу встановленого календаря цього ресурсу). Правильність обчислення ресурсів складає основу вартості проєкту і його обґрунтування, є висхідним джерелом для складання кошторису проєкту.

Для подальшого розроблення плану проєкту зробимо кілька припущень, що базуються на вивченні аналогічних кейсів, і розділимо всі ресурси на такі категорії: матеріальні, трудові, техніка. Наявність (за припущенням) ресурсів для тестового проєкту (обчислювальна техніка, витратні матеріали, кількість професіоналів тощо) представлена в табл. 3.2. Повний перелік потрібних ресурсів для кожної задачі (див. рис. 2.5) були попередньо розподілені у вигляді таблиці відповідності ресурсів і задач, які вони забезпечують (додаток Б).

Таблиця 3.2 – Відомості про наявні ресурси, їхню кількість і початкову вартість у тестовому проєкті

Найменування ресурсу	Наявність, одиниць	Обчислення обсягу ресурсу, %	Вартість використання одиниці ресурсу / год.
Обчислювальна техніка	7 одиниць	-	10
Копіювальна техніка	2 одиниць	-	5
Аналітик	3 працівника	300%	190 грн.
Системний програміст	2 працівника	200%	250 грн.
Розробник ПЗ	4 працівника	400%	250 грн.
Інженер	3 працівника	300%	160 грн.
Тестувальник	3 працівника	300%	200 грн.
маркетолог	2 працівника	200%	220 грн.
Папір ксероксний	4 упаковки	-	110 грн.
Зовнішній накопичувач HDD Aracser, 2 ТБ	1 одиниця	-	0,5 грн.

Спираючись на дані табл. 3.2 та додатку Б, створюємо базу ресурсів у MS Project у формі «Лист ресурсов», яка відкривається через вкладку РЕСУРС, список «Візуальний оптимізатор ресурсов». Перелік ресурсів вноситься просто у форму (рис. 3.1).

Название ресурса	Тип	Единицы измерения/материал	Краткое название	Группа	Макс. единиц	Стандартная ставка	Ставка сверхурочн	Затраты на исполн.	Начисление	Базовый календарь
Обчислювальна техніка	Материальні	одиниця	О	Техніка		10,00р.		100,00р.	По окончании	
Копіювальна техніка	Материальні	одиниця	К	Техніка		5,00р.		0,00р.	В начале	
Аналітик	Трудовой		А	Аналітики	300%	190,00р./час	200,00р./час	500,00р.	Пропорционал	Стандартный
Системний програміст	Трудовой		СП	Системні програмісти	200%	250,00р./час	400,00р./час	500,00р.	Пропорционал	Стандартный
Розробник ПЗ	Трудовой		РПЗ	Програмісти	400%	250,00р./час	400,00р./час	500,00р.	Пропорционал	Стандартный
Інженер	Трудовой		І	Монтажна	250%	160,00р./час	200,00р./час	500,00р.	Пропорционал	Стандартный
Тестувальник	Трудовой		Т	Тестувальна	100%	200,00р./час	250,00р./час	500,00р.	Пропорционал	Стандартный
маркетолог	Трудовой		м	Маркетолог	100%	190,00р./час	220,00р./час	400,00р.	Пропорционал	Стандартный
Папір ксероксний	Материальні	Упаковка	П	Матеріали		110,00р.		0,00р.	Пропорционал	
Зовнішній HDD накопичувач	Материальні	одиниця	НП	Техніка		2 480,00р.		0,00р.	В начале	
Арасер	Трудовой									

Рисунок 3.1 – Загальний перелік ресурсів проекту, обсяги, вартість в MS Project

При уведенні використовуються списки підстановки окремих типів даних. Кількість одиниць, вартість проставляються вручну. Також можна використати додаткові форми ресурсу, які активують кнопкою «Сведения» у розділі «Ресурси» головного меню (рис. 3.2).

Рисунок 3.2 – Форма для кожного ресурсу із розширенням інформації

Як видно (див. рис. 3.2), у формі передбачені поля і вкладки для більш повної характеристики кожного з ресурсів: особиста інформація, календар зайнятості і т.ін. Таку форму використовують для оптимізації розподілу ресурсів.

Останній крок – розподіл ресурсів за передбаченими у проекті задачами. Для кожної задачі вводять дані про необхідні види ресурсів (рис. 3.3).

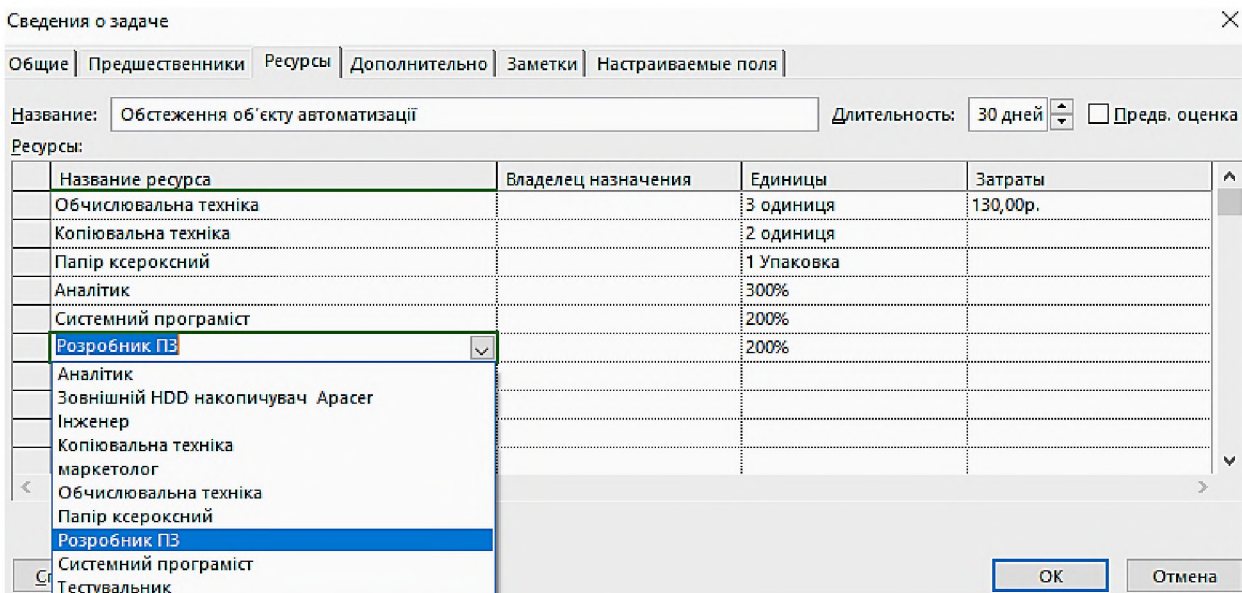


Рисунок 3.3 – Приклад закріплення ресурсів із переліку для обраної задачі

При цьому назву ресурсів обирають зі списку, одиниці згідно потреб, затрати обчислюються автоматично. Оновлений список операцій разом із ресурсами буде виглядати, як показано на рис. 3.4.

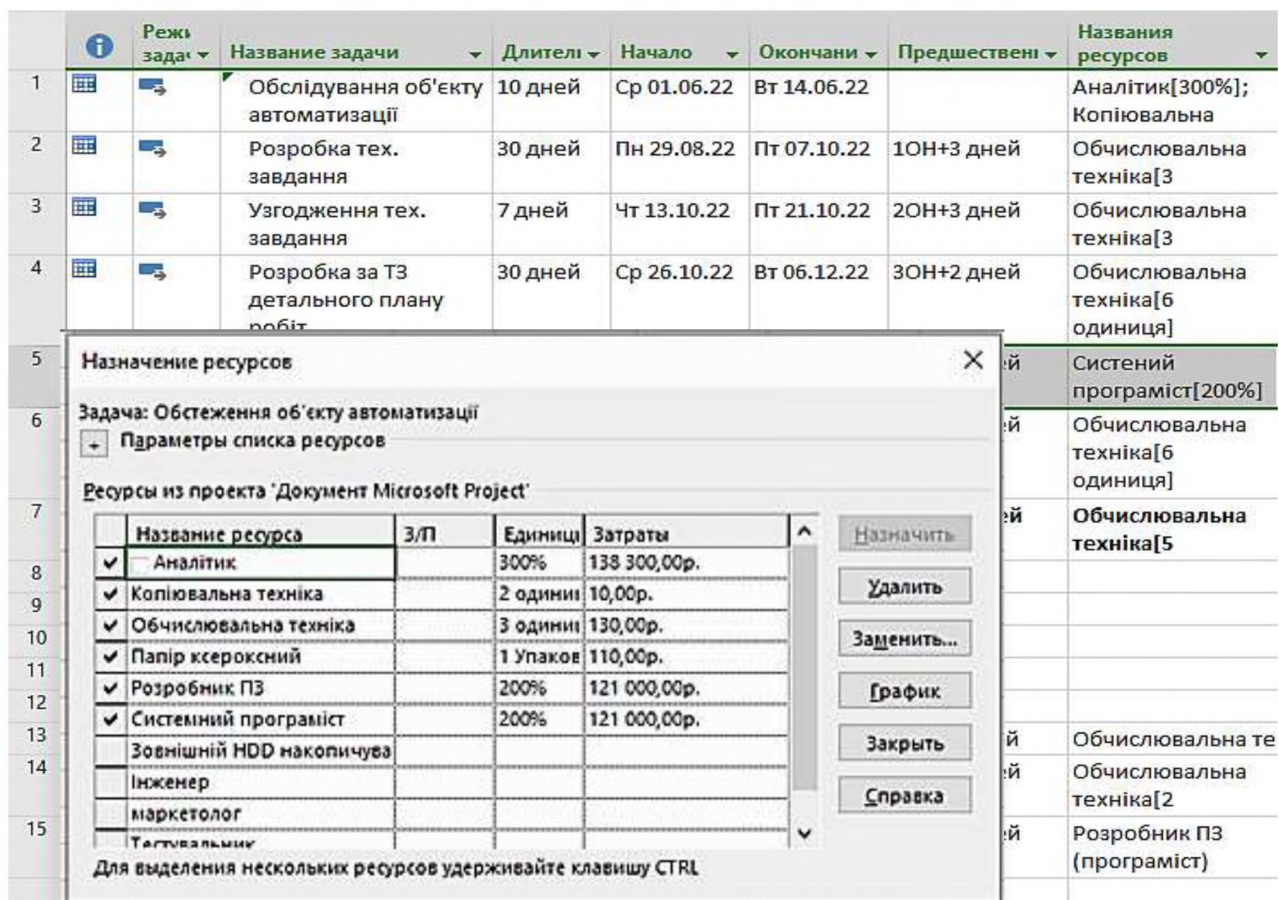


Рисунок 3.4 – Вигляд призначення ресурсів та їх параметрів для обраної задачі

Діаграма Ганта, яка показувала порядок і тривалість робіт, при цьому доповнюється підписами з переліком ресурсів по кожній операції (рис. 3.5).

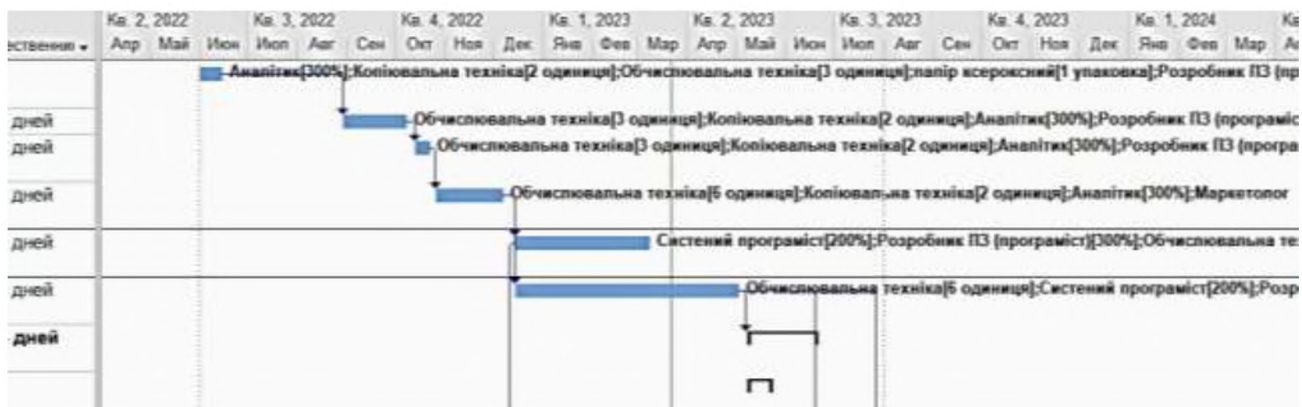


Рисунок 3.5 – Загальний вигляд плану розподілу ресурсів на діаграмі Ганта

Після проведення загальної оцінки потреби в ресурсах, складання переліку та проекту, графіку робіт у вигляді діаграми Ганта, слідує етап зіставлення наявних ресурсів і потреб, тобто виявлення нестачі або надлишків з метою коригування і усунення конфлікту ресурсів. На прикладі тестового проекту виявлено задачі з недоступністю певних трудових ресурсів (помічені червоним в списку ресурсів). Для коригування в MS Project є автоматизований засіб, який доступний із контекстного меню проблемної задачі – так званий інспектор задач, який дає підказки щодо усунення проблеми (рис. 3.6).

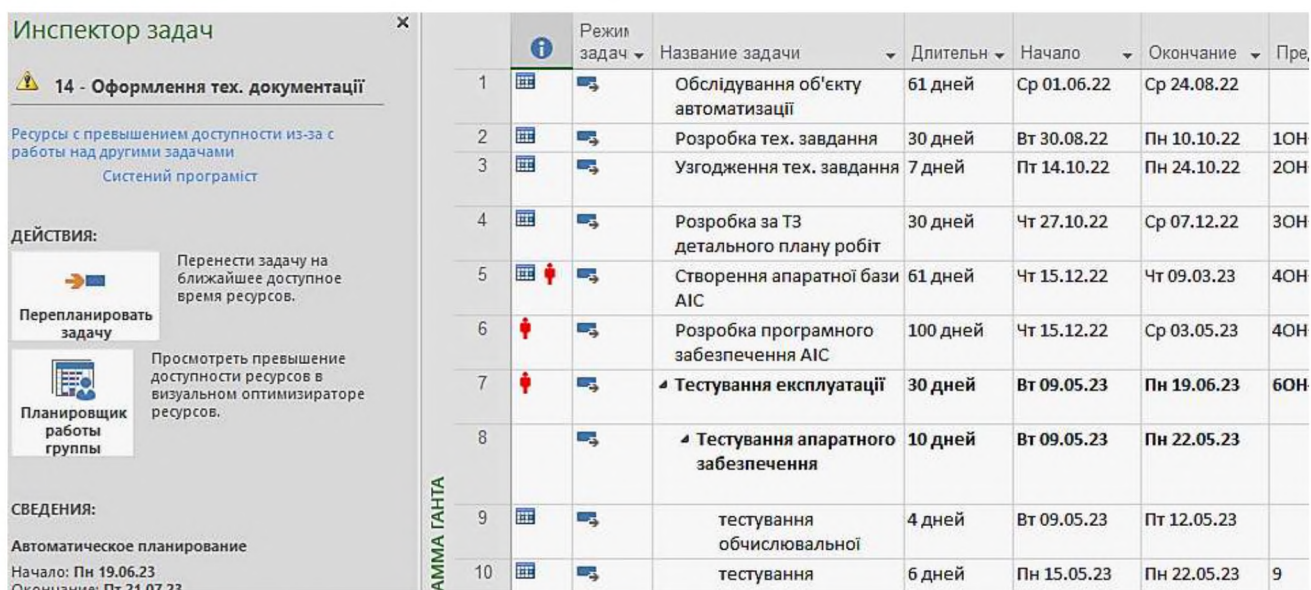


Рисунок 3.6 – Інспекція проблемних ресурсів із варіантами рішень

Безпосередньо в інспекторі задач виявлена проблема календарного планування – накладка по ресурсу (робота спеціаліста над іншими задачами), і можна обрати нову дату автоматично або вручну (див. рис. 3.6), тобто змінити або терміни використання ресурсу, або ж додати ресурс. Зміни можна побачити після коригування (рис. 3.7).

	Режи задач	Название задачи	Длительность	Начало	Окончание	Предшественники
10		тестування комунікаційного обладнання	6 дней	Пн 15.05.23	Пн 22.05.23	9
11		Роздільне тестування підсистеми програмного	10 дней	Вт 23.05.23	Пн 05.06.23	
12		Комплексне тестування автоматизованої	10 дней	Вт 06.06.23	Пн 19.06.23	11
13		Навчання персоналу	60 дней	Пт 05.05.23	Вт 15.08.23	5НН+3 дней
14		Оформлення тех. документації	25 дней	Пн 19.06.23	Пт 21.07.23	6ОН+2 дней
15		Підписання акту	3 дней	Чт 27.07.23	Пн 31.07.23	6ОН+2 дней

Рисунок 3.7 - Вигляд зміни термінів робіт після коригування ресурсів

Зайнятість ресурсів можна простежувати через візуальний оптимізатор ресурсів. На рис. 3.8 показано участь системного програміста у всіх видах робіт.

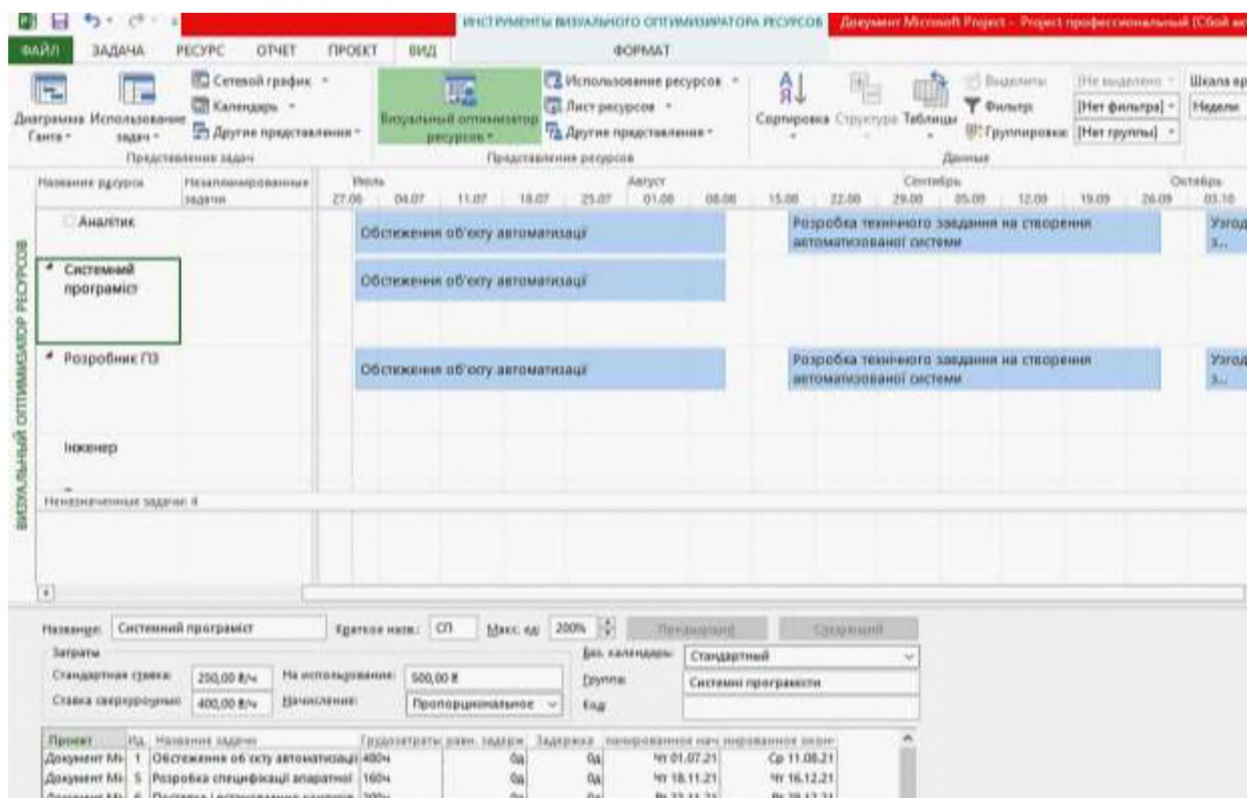


Рисунок 3.8 - Представлення на часовій шкалі зайнятості ресурсів

Інші проблеми можуть виникати через перевищення доступності ресурсів, що відображено в списку ресурсів червоними записами: розробник ПЗ та системний програміст представлені відповідно як 400 % і 300 % (1 спеціаліст на повний робочий день – 100 %). Проблему доступності найбільш наочно демонструють діаграми доступності ресурсів. Червоним кольором показано періоди недоступності ресурсів, види робіт, з якими це пов'язано (Додаток В). Системного програміста в тестовому прикладі не вистачає з грудня по березень, а потім потрібно лише 2.

Варіантів вирішення проблеми з доступністю ресурсів існує кілька: додати ресурси або запланувати понаднормову роботу (рекомендовано на короткий період). Можна скористатися автоматичним вирівнюванням ресурсів за допомогою засобу «Вирівняти ресурс». Програма через оптимізатор ресурсів запропонує варіант рішення: його можна прийняти, або відмінити й знайти інше рішення (рис. 3.9).

Назва ресурсу	Тип	Доступність	Ціна за годину	Вартість
Обчислювальна техніка	Матеріальні	100%		
Копіювальна техніка	Матеріальні	100%		
Аналітик	Трудові	100%		
<b>Системний програміст</b>	<b>Трудові</b>	<b>400%</b>	<b>200,00 €/час</b>	<b>400,00 €/час</b>
<b>Розробник ПЗ (програміст)</b>	<b>Трудові</b>	<b>300%</b>	<b>150,00 €/час</b>	<b>300,00 €/час</b>
Інженер	Трудові	300%	120,00 €/час	200,00 €/час
Тестувальник	Трудові	300%	150,00 €/час	300,00 €/час

Рисунок 3.9 – Відмітки в списку задач для перепланування кількості ресурсів

Спеціалізоване програмне забезпечення дозволяє ефективно автоматизувати більшість операцій планування та обчислень за проектом. На прикладі програмістів – розробників ПЗ на графіку чітко видно, що достатньо додати 100 % трудового ресурсу, тобто одного програміста з розробки програмного забезпечення.

Здійснення оптимального та якісного планування проекту дозволяє знизити ризику та досягти раціонального використання всіх видів ресурсів. Загалом, в MS Project надає значну кількість засобів для аналізу плану проекту, представлень майбутнього процесу. Якщо існує основна база проекту – календар, перелік задач, перелік і розподіл ресурсів, то легко представити план за різними показниками. Наприклад, можна відмітити за методом СРМ критичні задачі проекту, показати критичний шлях, показати віхи і т. ін. Це можна зробити за допомогою фільтрів: задати показ проекту у вигляді діаграми Ганта і вибрати фільтрацію, клацнувши на списку задач. У розділі фільтри просто обрати «Критические задачи». Або на вкладці «Формат» панелі «Інструменти діаграми Ганта» встановити прапорці по черзі «Критичні задачі», а потім «Задачі з резервом часу» (рис. 3.10).

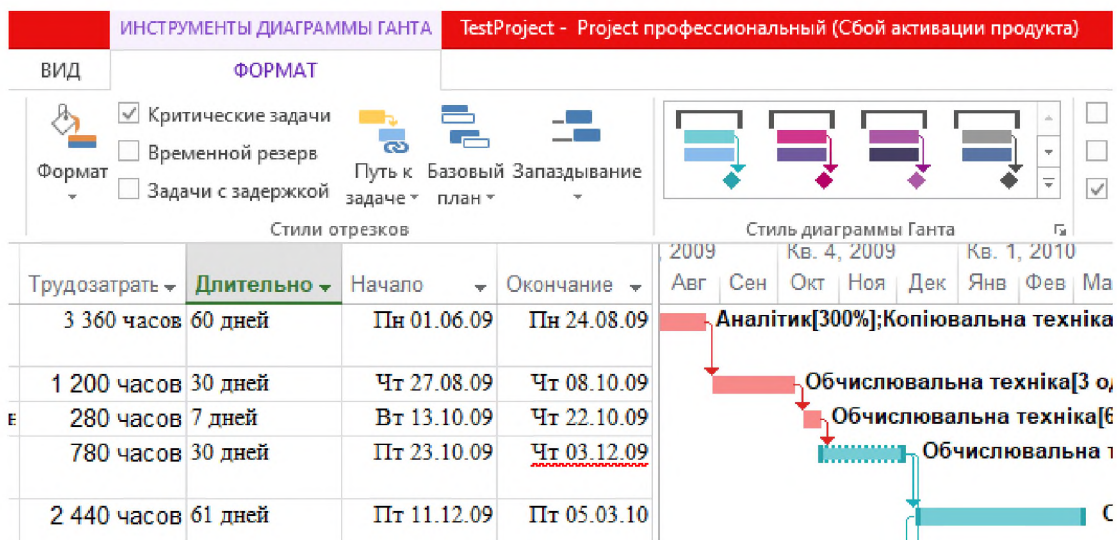


Рисунок 3.10 – Виділення програмою критичних задач

Для виведення окремих операцій для аналізу й коригування використовуються різноманітні фільтри. Також можуть бути відмічені віхи, сформовані звіти на конкретну дату. За допомогою програм управління проектами не лише складають план, але й проводять контроль виконання, використовують звіти по видам ресурсів, фінансів тощо.

Зрештою, створений план важливо зафіксувати, як опорний. Для фіксації певного плану необхідно зберегти варіант базового плану проекту. Це можна зробити за допомогою кнопки «Базовий план» на панелі «Проект» (рис. 3.11).

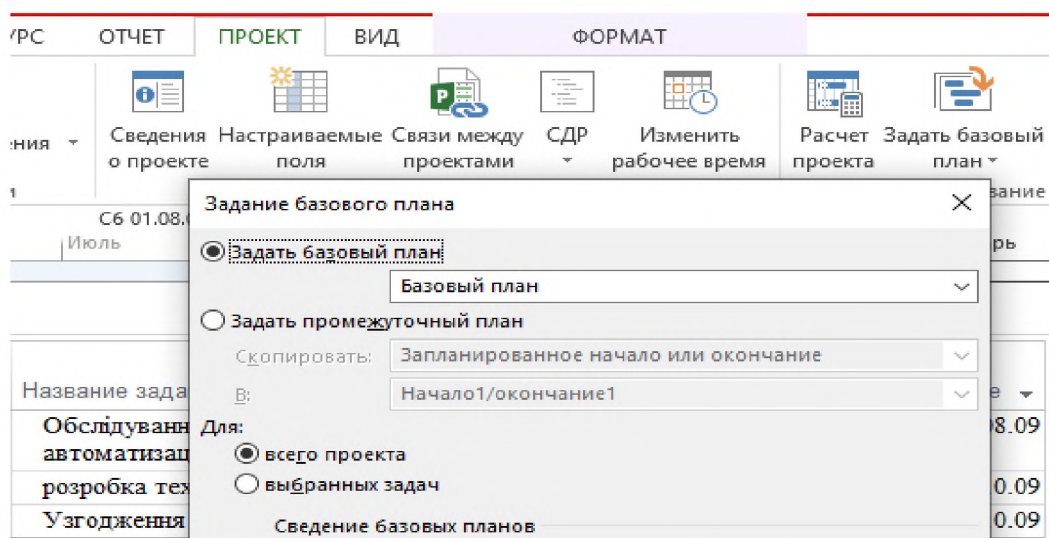


Рисунок 3.11 - Збереження базового плану

Візуалізувати базовий план можна за допомогою кнопки «Базовий план». На діаграмі Ганта з'явиться зображення схеми плану.

Використання інструментів діаграми Ганта досліджено також в системі Worksection (додаток Г). Особливими інструментами діаграми Ганта в обраній системі є можливість безпосередньо на відрізках діаграми управляти тривалістю задач, просто перетягуючи їх мишкою у розділі «Діаграма». При цьому календар підказує відтерміновану дату (рис. 3.12).

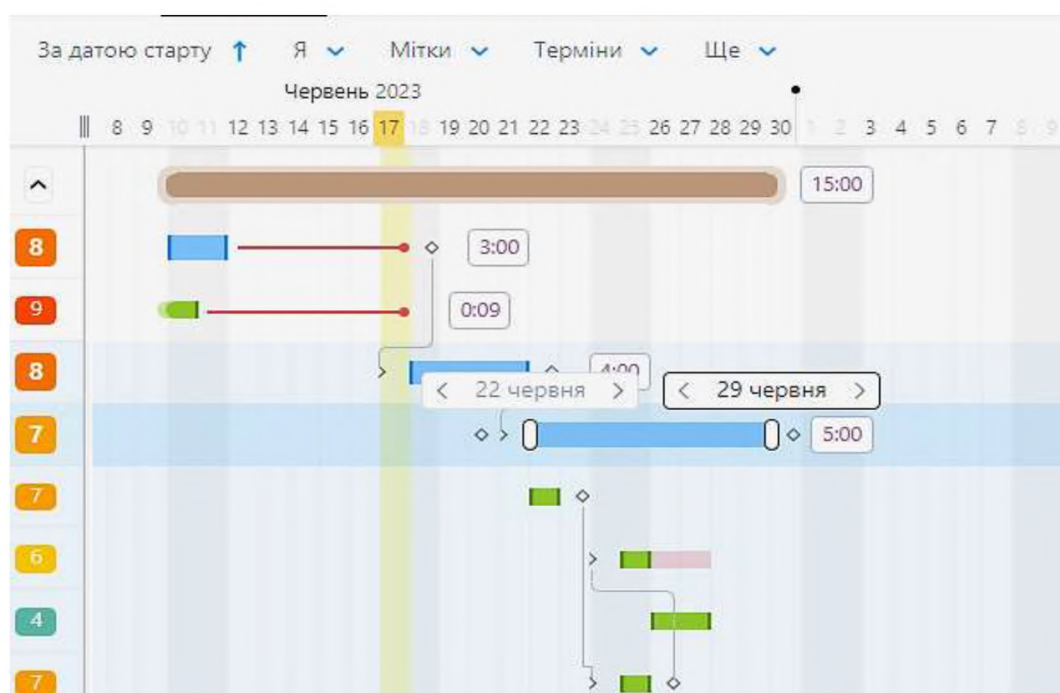


Рисунок 3.12 – Приклад налаштування тривалості задачі на діаграмі Ганта

Експерименти в різних системах показали, що управління ресурсами є більш потужною функцією програми MS Project. Це підкреслюється «захопленням» ресурсів і на діаграму Ганта. У той же час, в більшості інших систем діаграма Ганта виступає як візуалізатор порядку і тривалості задач. У Worksection реалізовано цікавий прийом – регулювання тривалості просто перетягуванням відрізків на діаграмі Ганта. Це дає можливість експериментувати з тривалістю етапів проекту.

### 3.2 Застосування програм управління проектами для керування задачами команди проекту на прикладі Worksection

Серед багатьох цінних функцій одним із недоліків системи MS Project для розробників програмного забезпечення, які здебільшого працюють за гнучкими методиками, є те, що вона не містить інструментарій для візуального керування задачами учасників проекту, тобто не відповідає тайм-трекінгу. Не лише інструментарій планування робіт і ресурсів проекту та засоби візуалізації, але й засоби комунікації, дистанційного управління часом працівників, контролю, комунікацій є необхідними також для командної роботи. В якості такої додаткової системи обрано систему українських розробників Worksection [42]. Система відповідає головним критеріям щодо управління проектами, має зрозумілий і привабливий інтерфейс і добре структурований інформативний вебсайт як основний інструмент роботи. Головне меню показано на рис. 3.13.

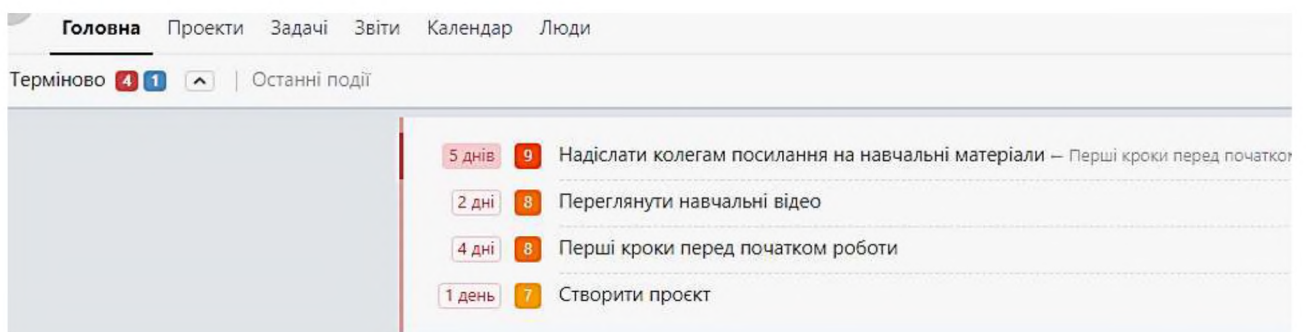


Рисунок 3.13 – Головне меню для початку роботи і приклад планування задач

З рис. 3.13 можна визначити, що обсяг функцій в даній програмі значно менше у порівнянні з MS Project або Jira. Робота команди, як і в попередніх прикладах, передбачає планування певних задач і розподіл за виконавцями. Фрагмент переліку задач окремих виконавців показано на рис. 3.14, повноекранний скрін наведено в додатку Д.

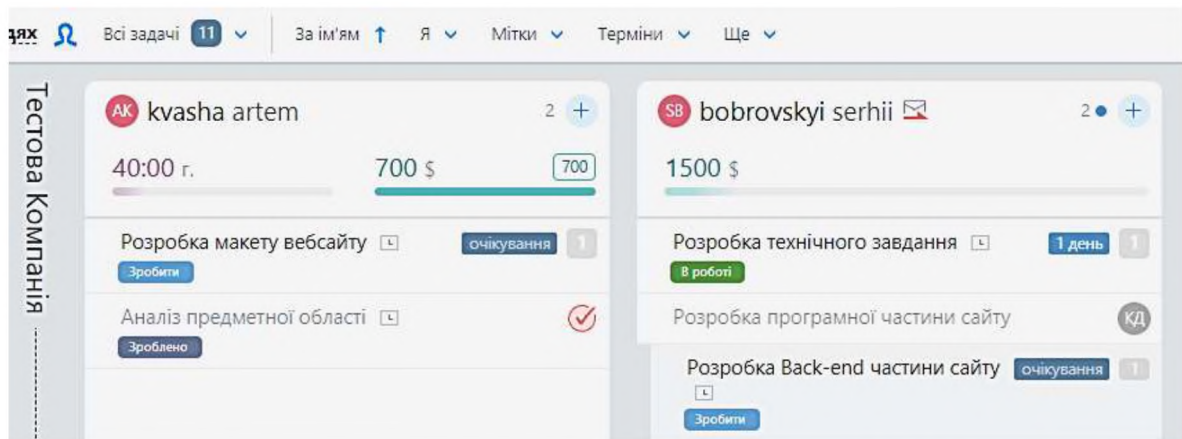


Рисунок 3.14 – Приклад закріплення задач та інформації про задачі у розділі «По людях» в системі Worksection

Прагнення ефективного управління задачами в компанії – одна з причин застосувати спеціальне програмне забезпечення для більш чіткого представлення робіт та їхніх основних атрибутів (рис. 3.15).

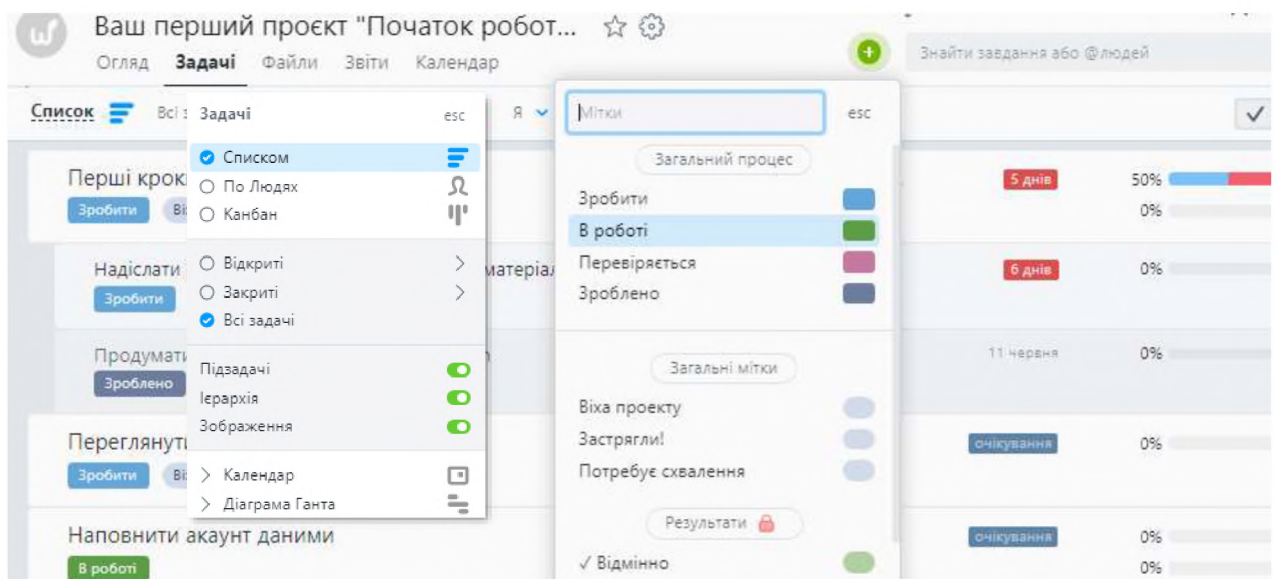


Рисунок 3.15 – Вигляд сторінки «Задачі» у Worksection та меню представлення

Як бачимо (див. рис. 3.15), у Worksection пункт меню «Задачі» дозволяє не лише різними способами представити перелік задач (списком, по людям, Канбан-дошка), але й відслідковувати їхнє виконання, ставлячи відповідні кольорові мітки: «Зробити», «В роботі», «Перевіряється», «Зроблено». Окрім того, можна створювати ієрархії задач і доручати частини різним виконавцям. В цьому ж вікні наочно за допомогою індикатора (справа) відображено відсоток виконання задач.

У багатьох ІТ-компаніях прийнята методологія Agile, яка передбачає гнучкий підхід до планування робіт і управління командною роботою. Одним із варіантів організації й контролю за виконанням робіт, який реалізовано у Worksection, є дошка Канбан – потужний метод управління задачами, які важливо відслідковувати по стадіям [43]. Task-трекінг показано на рис. 3.16.

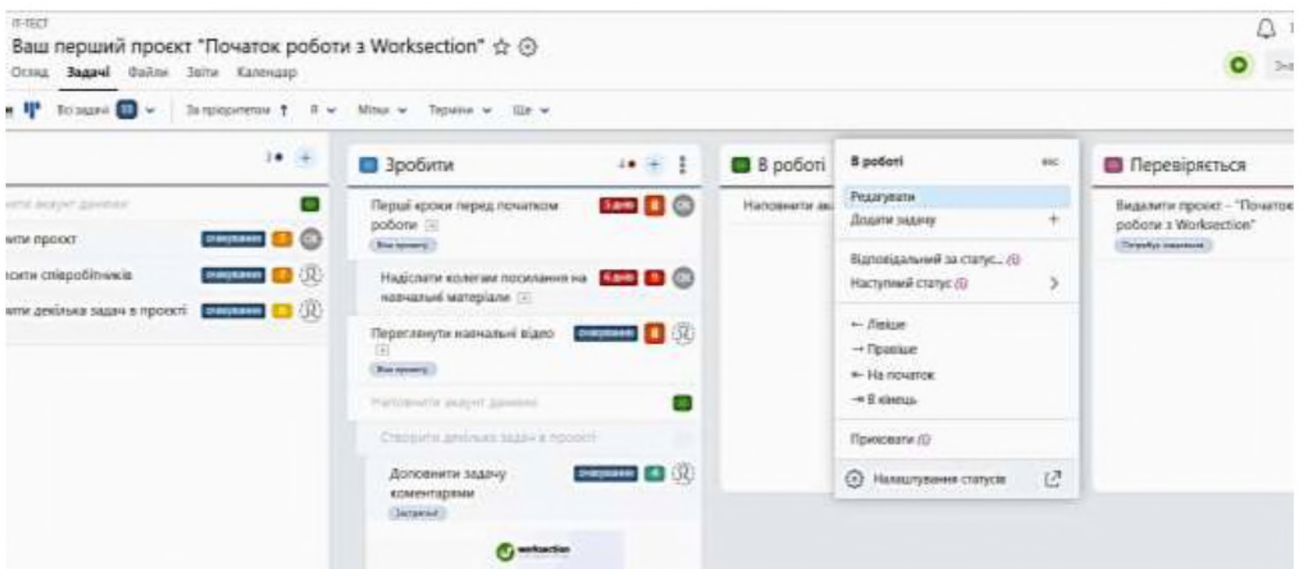


Рисунок 3.16 – Фрагмент вікна керування задачами у Канбан

Для керування командою призначений окремий пункт меню «Люди»: Команду додають саме в цьому вікні, а потім набирають в неї людей з числа співробітників. Вигляд вікна команди наведено на рис. 3.16.

Цікавим є можливість формувати звіти, як і в MS Project. На вкладці «Огляд» можна побачити багато інформації по проекту, зокрема: учасників, швидкість вирішення задач, кількість відкритих/закритих задач, грошові та часові витрати, активність, графіки задач по статусу та по виконавцям, найближчі події (рис. 3.17).

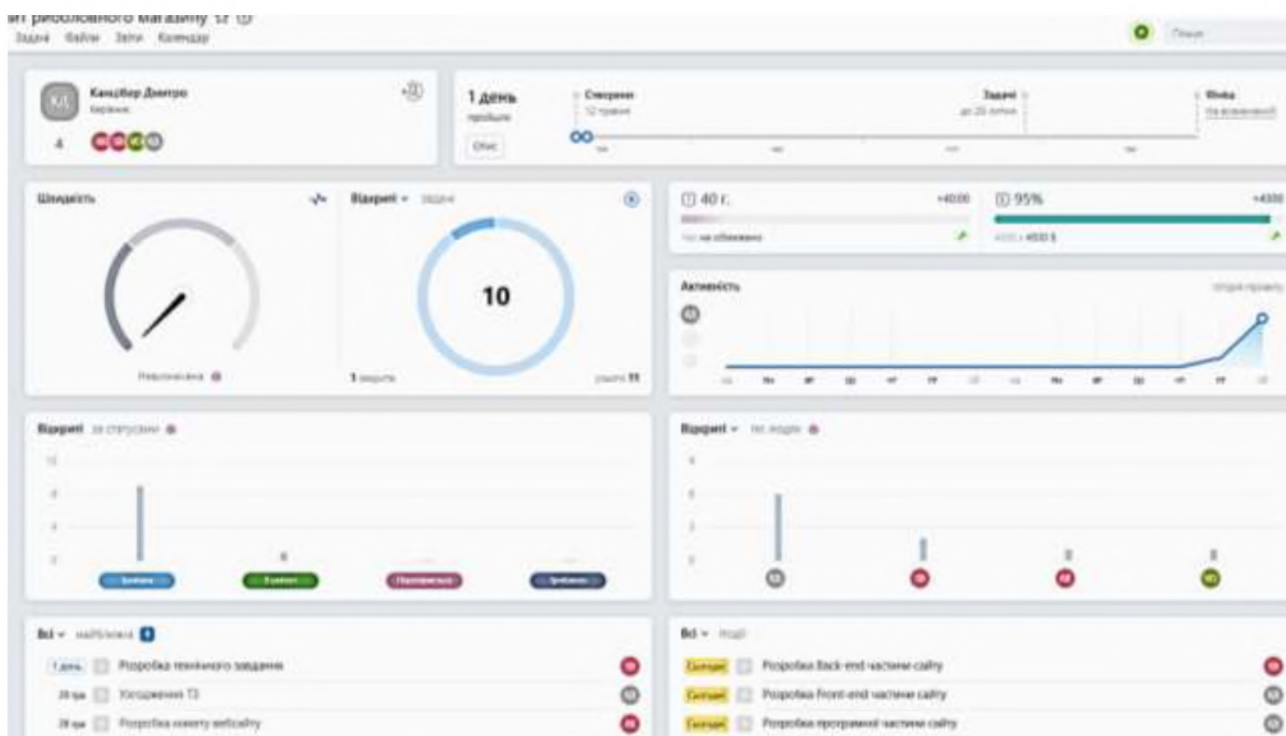


Рисунок 3.17 – Огляд проекту

У процесі управління компанією контакти відбуваються як між співробітниками і клієнтами, так і між співробітниками самої компанії (внутрішні комунікації), які отримують від керівників підрозділів завдання із встановленими термінами виконання (рис. 3.18).

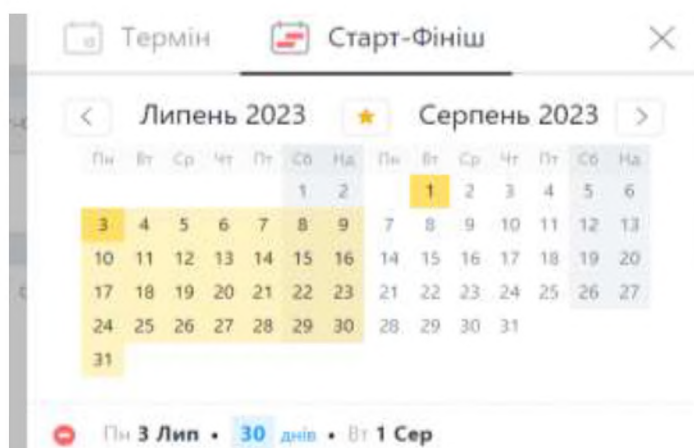


Рисунок 3.18 - Вікно встановлення термінів виконання (календар)

Як ми могли переконатися, для створення нового завдання потрібно обрати відповідний розділ відкрити пункт «Задачі» Новій задачі надати назву, обрати виконавця, термін виконання (в різних представленнях самої задачі). Після цього

зберегти дане завдання. Після збереження задачі буде відображатися в списку задач. Завдання є дієвим інструментом організації роботи персоналу та контролю робочого часу, візуальні інструменти організують працівників, а менеджер бачить усю картину зайнятості.

Обрана система особливо видається зручною для компаній в області ІТ, хоча серед користувачів зареєстровані представники досить різних видів діяльності. Така думка пов'язана з тим, що в цій системі є особливі інструменти для управління проектами. На момент реєстрації було автоматично створено перший проєкт «Вивчення системи», в рамках якого ми змогли протестувати основний функціонал. За потреби компанія може додавати й інші проєкти, задаючи їм назву, дати початку і завершення, запрошуючи учасників.

Підбиваючи попередні результати, можна назвати додаткові функції системи Worksection, які додають їй більшої привабливості та аргументів для багатьох бізнес-компаній почати користуватися під час роботи з проєктами.

1. В нових версіях системи передбачена інтеграція з тими додатками і месенджерами, які в компанії вже використовуються (GoogleDocs, Slack, Telegram), завдяки відкритому API, який дозволяє обмінюватися даними з Worksection. Це дозволяє полегшити і зв'язати роботу.

2. Задачам приділена особлива увага як основі успішності всіх бізнес-процесів компанії. Доступні фільтрація і сортування завдань, а також групові операції з завданнями, швидка зміна виконавців, термінів в умовах змін.

3. Передбачена робота з документами, надання колективного доступу, редагування за допомогою Google Docs, створення документів docx, xlsx і pptx.

4. Компанія ретельно зберігає дані більше 1000 компаній-користувачів, за 12 років не мали витоку інформації. Система Worksection працює в кластері з 4-х дата-центрів Tier-класу, рознесених географічно. Сервіс доступний 24/7 та 98,8 % на рік. За протоколом робляться щоденні копії даних. Якщо підключити FTP-сервер до Worksection, можна автоматично створювати резервні копії на сервері компанії. Кількість підключених клієнтів не обмежується.

### 3.3 Автоматизований контроль витрат та економічне оцінювання ефективності проєкту впровадження системи ERP на підприємстві

Важливою складовою реалізації проєкту є планування і контроль витрат. Планувати витрати потрібно так, щоб вони могли задовольнити потреби у фінансових ресурсах протягом усього періоду реалізації проєкту. Загалом, витрати по проєкту поділяються на інвестиційні та поточні. До інвестиційних належать витрати на інвестиції до основного капіталу (придбання приміщень та споруд, купівля технологій та обладнання), передвиробничі витрати на потреби в обіговому капіталі. Поточні витрати – витрати на випуск продукції, що містять витрати на придбання сировини, основних та допоміжних матеріалів, оплату праці, загальнозаводські та накладні витрати за звітний період.

При складанні витратного бюджету використовують масиви різних даних - економічну інформацію. Наприклад, джерелом нарахувань заробітної плати, оцінювання вартості проєкту є дані про трудовитрати (рис. 3.19).

	Назва ресурса	Трудовитрати	Табл.	Подробности	01 Май '22		10 Июл '22		18 Сен '22		27 Ноя '22		05 Дек '23	
					П	С	Ч	В	В	П	С	П	С	Ч
3	Аналітик	3 016 часоє	A	Трудовитр.			240ч		96ч	552ч	432ч	528ч		
	Обслування ат	240 часоє	A	Трудовитр.			240ч							
	Розробка тех. з	720 часоє	A	Трудовитр.					96ч	552ч	72ч			
	Узгодження тел	168 часоє	A	Трудовитр.							168ч			
	Розробка за ТЗ б	720 часоє	A	Трудовитр.							192ч	528ч		
	Навчання персон	720 часоє	A	Трудовитр.										
	Оформлення те	400 часоє	A	Трудовитр.										
	Підписання акт	48 часоє	A	Трудовитр.										
4	Системний програміст	3 520 часоє	B	Трудовитр.			160ч					672ч	736ч	640ч
	Обслування ат	160 часоє	B	Трудовитр.			160ч							
	Створення апар	960 часоє	B	Трудовитр.								336ч	368ч	256ч
	Розробка проєкт	1 600 часоє	B	Трудовитр.								336ч	368ч	384ч
	Тестування експ	240 часоє	B	Трудовитр.										
	Навчання персон	360 часоє	B	Трудовитр.										
	Оформлення те	200 часоє	B	Трудовитр.										
5	Розробник ПЗ (проєкт)	7 016 часоє	A	Трудовитр.			160ч		64ч	368ч	160ч	1 176ч	1 288ч	1 152ч
	Обслування ат	160 часоє	A	Трудовитр.			160ч							
	Розробка тех. з	480 часоє	A	Трудовитр.					64ч	368ч	48ч			
	Узгодження тел	112 часоє	A	Трудовитр.							112ч			
	Створення апар	1 440 часоє	A	Трудовитр.								504ч	552ч	384ч
	Розробка проєкт	3 200 часоє	A	Трудовитр.								672ч	736ч	768ч

Рисунок 3.19 – Представлення даних по трудовитратам фахівців по задачах

Спеціальний режим представлення в Microsoft Project (див. рис. 3.18) дає повну інформацію про затрати часу персоналу.

Відправними документами для планування витрат на проект є кошторис, календарний план проекту, опис наявних та необхідних ресурсів із вказанням вартості. На основі кошторису визначаються обсяги капітальних вкладень, які включають витрати на придбання технологічного, енергетичного та іншого обладнання, пристроїв, інструменту та виробничого інвентарю, необхідного для функціонування підприємства, роботи по монтажу цього обладнання, розробка проектної документації та ін.

Складання кошторису на роботи – це процес планування по відповідних статтях усіх витрат, що виникають під час виконання проекту. Завдяки використанню при плануванні проекту програми MS Project планування витрат здійснюється в середовищі електронного проекту на основі відомих функцій та алгоритмів. Сформувавши кошторис можна через додавання стовпчика з обчисленням сумарних планових витрат по операціям (на основі вартості і термінів використання ресурсів) та включення опції «суммарная задача проекта» у вкладинці «Формат». Отриманий розрахунок переглядають, наприклад, у переліку задач (рис. 3.20).

	Рез зад	Название задачи	Базовые затраты	Трудозатраты	Длитель	Начало	Окончани	Предш	Названия ресурсов
0		msproj11	2 866 653 €	15 712 часов	315 дней	Ср 01.06.22	Вт 15.08.23		
1		Обслідування об'єкту	103 959 €	560 часов	10 дней	Ср 01.06.22	Вт 14.06.22		Аналітик[3009 Копіювальна
2		Розробка тех. завдання	206 869 €	1 200 часов	30 дней	Пн 29.08.22	Пт 07.10.22	10Н+3 дней	Обчислюваль техніка[3
3		Узгодження тех. завдання	50 469 €	280 часов	7 дней	Чт 13.10.22	Пт 21.10.22	20Н+3 дней	Обчислюваль техніка[3
4		Розробка за ТЗ детального	143 884 €	960 часов	30 дней	Ср 26.10.22	Вт 06.12.22	30Н+2 дней	Обчислюваль техніка[6 одиниця]
5		Створення апаратної	656 880 €	3 840 часов	60 дней	Ср 14.12.22	Вт 07.03.23	40Н+5 дней	Систений програміст[20
6		Розробка програмного забезпечення	963 080 €	4 800 часов	100 дней	Ср 14.12.22	Вт 02.05.23	40Н+5 дней	Обчислюваль техніка[6 одиниця]
7		Тестування експлуатації	218 575 €	1 200 часов	30 дней	Вт 09.05.23	Пн 19.06.23	60Н+3 дней	Обчислюваль техніка[5
13		Навчання пер	326 575 €	1 800 часов	59 дней	Пт 05.05.23	Вт 15.08.23	5НН+3 дн	Обчислюваль

Рисунок 3.20 – Список задач проекту із підрахунком витрат по кожній задачі та сумарних економічних показників і вартості проекту

Згідно отриманих даних сумарної задачі розроблений проєкт буде тривати 315 днів (майже рік), а його вартість складатиме 2866,7 тис. грн (на основі попередніх припущень та планування). Інформація про витрати на виконання окремих задач в графічному вигляді може бути отримана на основі побудови звітів (рис. 3.21).

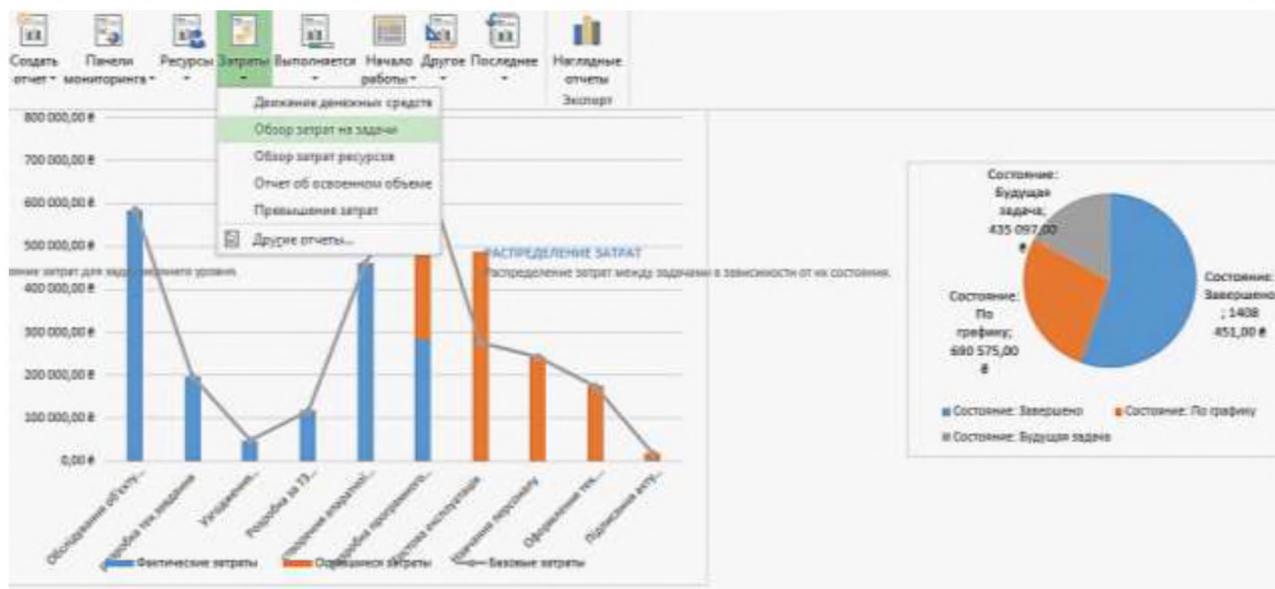


Рисунок 3.21 – Графік витрат по ресурсам на основі автоматизованого звіту

На основі звітів, наприклад, можна визначити періоди інвестиційних фаз проєкту, скоригувати розподіл фінансів та прийняти відповідні рішення. Окремо проводяться розрахунки за станом освоєння фінансів, контроль на дату і інші види моніторингу. Слід зауважити, що формування звітів можливо задати в режимі конструктора із власними налаштуваннями, вивести на друк.

Наприклад, до звіту про вартість ресурсів отримаємо окремо сформовану таблицю з аналізом вартості кожного виду ресурсів. Аналіз цих показників на задану дату дозволяє сформулювати про стан справ під час виконання проєкту та вплинути на подальші рішення.

Завдяки налаштуванням звіту на певну дату проводиться моніторинг використання ресурсів по всіх операціям та витратам, які заплановані до цього моменту, визначається відсоток незавершених операцій. Наприклад, діаграма дає можливість оцінити частку використаних і майбутніх потреб в ресурсах як

обсягах робочого часу, так і у відсотках (рис. 3.22). Аналіз гістограми показує, що залишилися операції, в яких будуть задіяні тестувальники і частково системний програміст і розробники ПЗ. Формування загальних звітів по проекту показує ефективність проведеної роботи проектною групою від ініціації до завершення роботи над планом проекту.

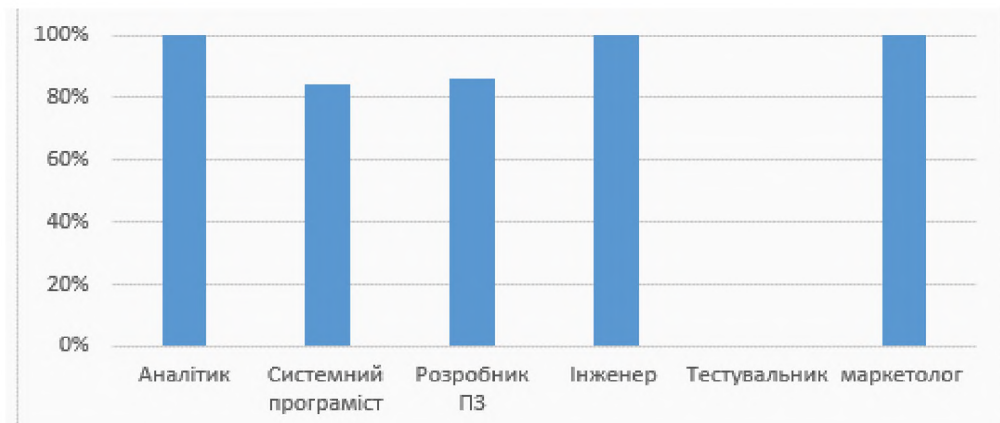


Рисунок 3.22 – Відсотки використання трудозатрат на визначену дату звіту

Загальна схема ефективності проекту включає розрахунок спільної значимості проекту та комерційну ефективність. Обґрунтування соціальної значимості було викладено в 2.1, тому оцінюються комерційна ефективність.

Основним показником ефективності є показник чистої теперішньої вартості NPV (Net Present Value). Чиста теперішня вартість проекту дорівнює різниці між майбутньою вартістю нинішніх і наступних витрат протягом усього його циклу:

$$NPV = \sum_{t=1}^n \frac{B_t - C_t}{(1+i)^t}, \quad (3.1)$$

$$NPV = \sum_{t=1}^n \frac{B_t}{(1+i)^t} - \sum_{t=1}^n \frac{C_t}{(1+i)^t}, \quad (3.2)$$

де:  $B_t$  – вигоди проекту в рік;

$C_t$  – витрати проекту на рік;

$i$  – ставка дисконту;

$n$  – тривалість проекту.

Якщо NPV позитивна, то проект можна рекомендувати до впровадження. Основна перевага показника NPV полягає в тому, що всі розрахунки проводяться

на основі грошових потоків, а не чистих доходів [44]. Однак, цей показник може бути застосований до поточного проєкту як наближений, оскільки його тривалість менше року (315 днів). Альтернативними показниками ефективності є термін окупності проєкту.

Досягнення успіху при плануванні проєкту стало можливим завдяки вибору та використанню спеціального програмного забезпечення.

### **Висновки до розділу 3**

Використання системи управління проєктами MS Project для планування та налаштування ресурсів проєкту, контролю їх доступності та автоматичного коригування ресурсного забезпечення показало високу результативність, зрозумілість та охоплення всіх критеріїв завдяки вбудованим інструментам візуалізації та обробки даних.

Альтернативне випробування системи Wrike показало наявність ефективної діаграми Ганта і зв'язку із завданнями, але ресурсне планування не таке всебічне.

В розділі зроблено короткий огляд застосування іншої системи управління проєктами, яка більше орієнтована на гнучкі методології розробки ПЗ – систему планування роботи й комунікацій команди Worksection. Серед переваг даної системи можна назвати чіткий інструментарій та зрозумілий інтерфейс при наборі команди проєкту, постановці й контролю виконання завдань. Система має такі інструменти візуалізації робіт і їхніх характеристик, як діаграму Ганта і дошку Канбан. Однак, інструментарій планування ресурсів доволі скромний у порівнянні з Jira або MS Project.

Ефективність проєкту при наявності реальних даних обчислюється за показником чистої теперішньої вартості NPV. Проєкт приймається в роботу, якщо цей показник буде більше 0 при тривалості більше 1 року. Оцінювання тривалості проєкту впровадженні інформаційної системи показало період 315 днів, тобто менше року.

## ВИСНОВКИ

У кваліфікаційній роботі на основі узагальнення значної кількості джерел інформації розкрито етапи становлення управління проектами як наукового напрямку, роль міжнародних інституцій та основоположних праць у структуруванні термінів, моделей, методів проектів інформаційних систем. Проаналізовано функціональні характеристики найбільш рейтингових систем управління проектами. За результатами роботи сформовані наступні висновки.

1. Управління проектами інформаційних систем та розробки програмного забезпечення мають спільні керівні засади, які базуються на теорії управління проектами як порівняно новому науковому напрямку. Проекти класифікують за різними критеріями та ознаками. Проекти інформаційних систем є одним із видів технічних інформаційних проектів. Як і всі інші, проекти інформаційних систем мають сформовану концепцію, мету, завдання, вирішують певне коло проблем, мають методологію досягнення цілей.

2. Життєвий цикл проекту є базовим, вихідним поняттям для дослідження проблем реалізації проекту, фінансування робіт, прийняття рішень про доцільність капіталовкладень та деталізації проекту.

3. На основі складання характеристики основних методів розробки ІТ-проектів виявлено сильні і слабкі сторони цих методів, у т.ч. Agile, Microsoft Solutions Framework, Personal Software Process / Team Software Process, DevOps. Розгляд методів враховував важливі компетентності, які повинен мати кожен учасник/виконавець ІТ-проекту: унікальних знань, метод критичного шляху, розуміння ієрархічної структури робіт, PERT-метод та ін.

4. Системи управління проектами – спеціалізовані інформаційні системи, які дозволяють планувати, контролювати та відстежувати прогрес виконання завдань, планування ресурсів та бюджету проекту, забезпечують ефективну комунікацію між учасниками проекту, дозволяють реалізувати проект у визначені терміни та в межах бюджету. Для виконання окремих етапів проектів можуть бути застосовані різні види програмного забезпечення. Порівняльний

аналіз функціональних можливостей найбільш рейтингових систем (Jira, Microsoft Project, Asana, Trello, Worksection) показав, що всі системи мають сильні і слабкі сторони. Jira та MS Project є досить потужними та функціональними системами, які дозволяють детально планувати та контролювати проекти. На основі проведеного аналізу в якості середовища планування та управління проектом обрано MS Project з метою забезпечення програмної підтримки виконання проекту на етапах життєвого циклу.

5. Розроблено реалістичну модель проекту впровадження інформаційної системи на тестовому підприємстві на прикладі хмарної ERP-системи. Обґрунтування вибору системи базується на даних розробника системи компанії «СофтПро» та її офіційного вебсайту.

6. Для опису і розробки проекту впровадження ERP-системи «Універсал 9» обрано чотирьохфазну модель життєвого циклу на основі досвіду впровадження такої системи на різних підприємствах.

7. Використовуючи інструментарій MS Project та альтернативних систем Wrike, Worksection були визначені WBS проекту, візуалізація на діаграмі Ганта та сіткові моделі, різні види представлення задач, ресурсів, витрат. Прикладна частина роботи над проектом включала моніторинг наявних ресурсів, виявлення обсягів потреби для завдань проекту та розподіл ресурсів між відповідними задачами. В системі управління проектами MS Project використано всі наявні інструменти як для ведення плану ресурсів (форми ресурсів, ресурси по задачам, діаграми доступності ресурсів), так і можливості коригування за допомогою інспекторів задач і управління всіма видами ресурсів.

8. Зроблено короткий огляд застосування системи управління проектами, яка більше орієнтована на гнучкі методології розробки ПЗ – систему планування роботи й комунікацій команди Worksection. Серед переваг даної системи можна назвати чіткий інструментарій та зрозумілий інтерфейс при наборі команди проекту, постановці й контролю виконання завдань. Система має такі інструменти візуалізації робіт і їхніх характеристик, як діаграму Ганта і дошку

Канбан. Обране програмне забезпечення MS Project дозволяє здійснити поточний моніторинг та підсумковий контроль за виконанням проекту.

9. На основі усереднених розцінок по видам робіт, вартості окремих ресурсів проведені фінансові розрахунки щодо вартості кожної з операцій та загальної вартості проекту. Сформовані різні звіти про стан використання ресурсів та коштів на обрану звітну дату.

10. Сформовані рекомендації щодо вибору системи управління проектами в разі масштабних проектів впровадження інформаційних систем за каскадною чотирифазною моделлю життєвого циклу проекту. В разі потреби управління командою проекту та наявності гнучких підходів може підійти Jira Wrike. Для невеликих гнучких проектів, які потребують команди розробників, ефективною буде система Worksection.

Приклад розробленого проекту може бути обраний за основу при впровадженні ERP-систем, зокрема «Універсал 9.0», на реальних підприємствах.

Результати кваліфікаційної роботи можуть бути використані на курсах, тренінгах з управління проектами.