

Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна
Факультет комп'ютерних наук
Кафедра теоретичної та прикладної системотехніки

«Затверджую»
Зав. кафедри теоретичної та
прикладної системотехніки
_____ д.т.н., проф. С. І. Шматков
«__» _____ 2024 р

Пояснювальна записка

до кваліфікаційної роботи
бакалавра

на тему: «**АВТОМАТИЗОВАНА КОМП'ЮТЕРНА МОДЕЛЬ
УПРАВЛІННЯ ПРОЄКТАМИ**»

Захищено на засіданні
Атестаційної комісії № 42
протокол № __ від __.06.2024 р.
Оцінка ____ / ____
Голова Атестаційної комісії
_____ **СКОБ Ю. О.**
(підпис) (прізвище та ініціали)

Виконав:
студент 4 курсу, групи КУ– 41
Галузь знань: 15 – Автоматизація та
приладобудування
Спеціальність: 151 – «Автоматизація та
комп'ютерно-інтегровані технології»
Бут Богдан Олександрович _____
підпис

Керівник: к.т.н, доцент кафедри ТПС
БУЛАВІН Дмитро Олексійович _____
підпис

Рецензент:

Харків – 2024

АНОТАЦІЯ

Пояснювальна записка до кваліфікаційної роботи бакалавра складається зі вступу, трьох розділів, висновків, списку використаних джерел і додатків. Загальний обсяг роботи складає 76 сторінок, із яких 50 сторінок основної частини з 18 рисунками, 7 таблицями, 50 найменуванням списку використаних джерел та додатками.

Метою кваліфікаційної роботи є підвищення ефективності управління проектами за допомогою розробленої інноваційної моделі.

Об'єктом дослідження є процес автоматизованого управління проектами за допомогою комп'ютерної моделі, що включає програмні засоби та технології для планування, виконання, контролю та аналізу проектних дій.

Предметом дослідження є принципи, методи та інструменти, що використовуються в комп'ютерних моделях автоматизованого управління проектами.

Проблема, яка вирішується в кваліфікаційній роботі, полягає в підвищенні ефективності та прозорості управління проектами через розробку та впровадження автоматизованої системи, яка забезпечує ефективне планування, розподіл ресурсів, аналіз продуктивності та управління ризиками.

Область застосування – розробка та впровадження автоматизованих систем управління проектами, що можуть бути використані в різних галузях для покращення якості управління проектами та підвищення ефективності діяльності організацій.

Ключові слова: управління проектами, автоматизація, планування, розподіл ресурсів, моніторинг, аналіз продуктивності, управління ризиками, програмне забезпечення, Java, СРМ, метод Монте-Карло.

ABSTRACT

The explanatory note to the bachelor's qualification work consists of an introduction, three chapters, conclusions, a list of references, and appendices. The total volume of the work is 76 pages, including 50 pages of the main part with 18 figures, 7 tables, 50 references, and appendices.

The purpose of the qualification work is to develop an innovative model for effective project management.

The object of the research is automated computer models for project management as systems that include software tools and technologies for planning, execution, control, and analysis of project activities.

The subject of the research is the principles, methods, and tools used in automated computer models for project management.

The problem addressed in the qualification work is to improve the efficiency and transparency of project management by developing and implementing an automated system that ensures effective planning, resource allocation, performance analysis, and risk management.

The area of application includes the development and implementation of automated project management systems that can be used in various industries to improve the quality of project management and increase organizational efficiency.

Keywords: project management, automation, planning, resource allocation, monitoring, performance analysis, risk management, software, Java, CPM, Monte Carlo method.

Зміст

ВСТУП	5
РОЗДІЛ 1: АНАЛІЗ АВТОМАТИЗАЦІЇ ПІДХОДІВ ДО УПРАВЛІННЯ ПРОЕКТАМИ	8
1.1. Поняття та сутність автоматизованої комп'ютерної моделі управління проектами	8
1.2. Потенціал та можливості автоматизації підходів до управління проектами за допомогою комп'ютерних моделей	9
1.3. Огляд існуючих існуючих комп'ютерних моделей управління проектами	11
Висновки до розділу 1.	15
РОЗДІЛ 2: ПРОЄКТУВАННЯ УЗАГАЛЬНЕНОЇ АВТОМАТИЗОВАНОЇ КОМП'ЮТЕРНОЇ МОДЕЛІ УПРАВЛІННЯ ПРОЄКТАМИ	16
2.1. Аналіз основних засобів автоматизації управління проектами	16
2.2. Релевантність теоретичних концепцій для розробки ефективних комп'ютерних моделей	20
Висновки до розділу 2.	27
РОЗДІЛ 3: ПРАКТИЧНА РЕАЛІЗАЦІЯ КОМП'ЮТЕРНИХ МОДЕЛЕЙ УПРАВЛІННЯ ПРОЕКТАМИ	28
3.1. Вимоги до нової комп'ютерної моделі для оптимізації управління проектами	28
3.2. Виклики впровадження комп'ютерних моделей управління проектами	30
3.3. Стратегії для успішного впровадження комп'ютерних моделей управління проектами.	34
3.4. Вплив автоматизації управління проектами на продуктивність	39
3.5. Майбутні тенденції та інновації в комп'ютерних моделях управління проектами	44
Висновки до розділу 3.	51
ВИСНОВКИ	52
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	54
ДОДАТКИ	60

ВСТУП

Актуальність дослідження. У сучасному світі, де темпи життя надзвичайно швидкі, а конкуренція безперервно зростає, ефективне управління проєктами стає ключовим фактором успіху для організацій будь-якого рівня і галузі. Автоматизовані комп'ютерні моделі управління проєктами надають можливість не лише оптимізувати витрати часу та ресурсів, але й значно полегшують процес планування, виконання та контролю над проєктами. Ця дипломна робота присвячена дослідженню та аналізу автоматизованих комп'ютерних моделей управління проєктами, їхнього впливу на ефективність управління та можливостей їх використання в сучасному бізнес-середовищі. Вона ставить за мету розкрити основні принципи, переваги та недоліки таких моделей, а також визначити напрямки подальших досліджень та розвитку в цій області.

Дослідження автоматизованих комп'ютерних моделей управління проєктами спрямоване на виявлення та аналіз інноваційних підходів та технологій, що застосовуються в цій сфері. З урахуванням швидкого технологічного розвитку, розуміння сучасних тенденцій та використання передових інструментів є важливими для забезпечення конкурентоспроможності організацій.

Загалом, дослідження автоматизованих комп'ютерних моделей управління проєктами є актуальним і важливим завданням у контексті підвищення ефективності управління та досягнення успіху в сучасному бізнес-середовищі.

Актуальність дослідження автоматизованих комп'ютерних моделей управління проєктами не може бути переоцінена в контексті сучасного бізнес-середовища. Зростаюча складність та об'єм проєктів, швидкі темпи змін у технологіях та вимоги ринку ставлять перед організаціями серйозні виклики у плануванні, виконанні та контролі за проєктами.

Використання автоматизованих комп'ютерних моделей управління проектами дозволяє підвищити ефективність та точність процесів управління, зменшити ризики затримок та перевищення бюджету, а також оптимізувати розподіл ресурсів. Це особливо важливо в умовах постійних змін та нестабільності, коли потрібна швидка реакція на зміни у внутрішньому та зовнішньому середовищі.

Застосування таких моделей не лише сприяє підвищенню продуктивності та конкурентоспроможності компаній, але й є ключовим фактором у забезпеченні успішного виконання стратегічних цілей та впровадженні інновацій. Таким чином, дослідження автоматизованих комп'ютерних моделей управління проектами має велике значення для практики управління та розвитку сучасних бізнес-організацій.

Розробкою даної проблеми займалися багато науковців, представники різних галузей науки. Такі науковці, як: В.Є. Козаченко, Ю.Г. Турло, В.В. Морозов, А.А. Балан, С.В. Філіппова, О.С. Балан, Є.І. Масленніков, Т. Вілфрід та ін. задавалися проблемою дослідження та створення сучасних інформаційних технологій. Науковці мали за мету: проаналізувати особливості управління проектами підприємства; поглибити теоретичні положення щодо ефективності управління проектами; виявити чинники підвищення ефективності управління проектами; дослідити особливості розробки інформаційної технології; розробити інформаційну технологію управління мультипроектами підприємства.

Але, незважаючи на це, сьогодні існує потреба у дослідженні, яке б узагальнило, систематизувало існуючі відомості з даної проблеми.

Враховуючи все вищесказане, нами і була обрана тема дипломної роботи: "Автоматизована комп'ютерна модель управління проектами".

Об'єкт дослідження – процес автоматизованого управління проектами за допомогою комп'ютерної моделі, що включає програмні засоби та технології для планування, виконання, контролю та аналізу проектних дій.

Предмет дослідження - Принципи, методи та інструменти, що використовуються в автоматизованих комп'ютерних моделях управління проєктами.

Мета дипломної роботи - підвищення ефективності управління проєктами за допомогою розробленої інноваційної моделі.

Відповідно до мети були визначені наступні **завдання**:

1. Провести огляд літератури та аналіз наукових досліджень щодо автоматизованих комп'ютерних моделей управління проєктами.
2. Вивчити основні принципи та методики автоматизованого управління проєктами, їхні переваги та недоліки.
3. Розглянути практичні приклади впровадження автоматизованих комп'ютерних моделей управління проєктами в різних сферах бізнесу та проектної діяльності.
4. Здійснити порівняльний аналіз різних автоматизованих комп'ютерних моделей управління проєктами за їхніми характеристиками та ефективністю.
5. Визначити перспективи подальшого розвитку та вдосконалення автоматизованих комп'ютерних моделей управління проєктами.

Для розв'язання поставлених завдань нами були використані такі **методи дослідження**: теоретико-критичний аналіз літератури з теми дослідження; зіставлення, узагальнення і синтезування здобутої інформації тощо.

Робота може бути використана студентами ВНЗ для підготовки до семінарських занять, також може бути використана викладачами для проведення лекції, практик тощо.

РОЗДІЛ 1.

АНАЛІЗ АВТОМАТИЗАЦІЇ ПІДХОДІВ ДО УПРАВЛІННЯ ПРОЕКТАМИ

1.1. Поняття та сутність автоматизованої комп'ютерної моделі управління проектами

У сучасних умовах ефективне управління проектами стає неможливим без застосування передових програмних засобів, оскільки розміри та складність проектів постійно зростають. Перші програми для управління проектами з'явилися у 60-х роках, базуючись на алгоритмах сіткового планування та методі критичного шляху. З часом додалися функції ресурсного та бюджетного планування, а також засоби контролю за прогресом виконання проекту. Особливо активний розвиток цих систем відбувся у 80–90-х роках минулого століття.

Автоматизовані системи управління проектами включають:

- Інструменти для календарно-сіткового планування.
- Функціонал для розробки бюджетів, аналізу ризиків, управління контрактами тощо.
- Засоби для обмеження доступу до проектних даних та їх спрощення.
- Засоби для організації комунікацій між учасниками проекту.
- Функціонал для інтеграції з іншими програмами [14, с. 54].

Універсальні системи управління проектами, доступні на ринку, надають основний набір функцій:

1. Інструменти для створення структури робіт проекту та планування за методом критичного шляху.
2. Функціонал для планування ресурсів і витрат, включаючи організаційну структуру виконавців і витрат, призначення ресурсів по роботах та контроль за використанням ресурсів.
3. Інструменти для контролю за прогресом виконання проекту.

4. Засоби для графічного подання структури проекту та створення звітів (діаграми Гантта, сіткові діаграми).
5. Функціонал для організації групової роботи та співпраці між учасниками проекту.

За останнє десятиліття автоматизовані системи управління проектами використовуються не лише у традиційних сферах, таких як великі будівельні чи оборонні проекти. Ринок насичений універсальними програмними пакетами для персональних комп'ютерів, які автоматизують функції планування та контролю за проектами. Сьогодні навіть доступні програмні пакети можуть ефективно підтримувати планування проектів з десятками тисяч завдань і тисячами видів ресурсів [41, с. 65].

Економічне зростання вимагає стійкого розвитку підприємства в умовах зростаючої конкуренції. Управління проектами розглядає проект як систему пов'язаних процесів, спрямованих на досягнення запланованих результатів за обмеженими ресурсами та часом. Зростання конкуренції на ринку змушує підприємства бути гнучкими та оперативними, виконуючи проекти швидко, ефективно та з високою якістю. Досягнення стратегічних цілей компанії вимагає правильного вибору проектів та їхнього успішного втілення, що можливе завдяки компетентному управлінню та використанню інформаційних технологій.

1.2. Потенціал та можливості автоматизації підходів до управління проектами за допомогою комп'ютерних моделей

Інформаційні технології відіграють ключову роль у реалізації управління мультипроектами, що передбачає ведення кількох проектів одночасно. Завдання ІТ у мультипроектному управлінні включають опис робіт, прибутків, витрат та ресурсів проектів, складання розкладу робіт з визначенням термінів та критичних операцій, розрахунок бюджету та ресурсів, аналіз ризиків та ведення обліку за кожним етапом проекту. Також мультипроектне управління включає архівування даних, аналітичні функції та контрольні-ревізійні процедури.

Інформаційні системи управління проектами вирішують завдання розробки розкладу робіт, визначення критичного шляху, аналізу ризиків та ведення проекту. Вони допомагають у класифікації проектів за значущістю та пріоритетами, що дозволяє ефективніше використовувати ресурси компанії та оптимізувати розклад проектів. Переваги використання таких систем включають підвищений контроль, передачу досвіду та чітке планування робіт [40, с. 53].

Інформаційні технології дозволяють автоматизувати різні аспекти управління проектами: планування, управління ресурсами та витратами, аналіз ризиків та комунікації. Автоматизовані системи зазвичай базуються на моделі проекту, що складається зі структури робіт, ресурсів та матриці призначення ресурсів на роботи.

Структура робіт проекту включає етапи і завдання з їхніми взаємозв'язками та часовими рамками. У автоматизованому режимі програма визначає дати початку та завершення робіт, розраховує календарний графік та резерви часу. Це дозволяє оптимізувати проект та відстежувати зміни графіка в процесі виконання.

Структура ресурсів проекту включає людські ресурси, обладнання, матеріали та кошти. Параметри ресурсів, такі як кількість, продуктивність та вартість, можуть бути описані у електронних таблицях. Матриця призначень містить інформацію про ресурси та їх використання на різних етапах проекту. Після оформлення даних програма автоматично перераховує календарний план з урахуванням обмежень ресурсів.

Менеджер та команда проекту, використовуючи програмне забезпечення, можуть значно підвищити ефективність та швидкість управлінських рішень. Пакети програмного забезпечення дозволяють автоматизувати всі основні операції, включаючи розробку розкладу, аналіз ризиків, оцінку виконання та прогнозування параметрів проекту.

Ефективність систем управління проектами визначається сукупністю витрат та доходів. Навіть невеликі помилки в розрахунках можуть призвести до

значних втрат у бюджеті проекту. Тому правильне управління проектом вимагає уважного аналізу і розрахунків на всіх етапах його реалізації.

Впровадження ІТ у проектне управління вимагає комплексного підходу, що включає планування робіт і контроль за їх виконанням. Основною складовою цього підходу є розробка детального плану, який враховує усі аспекти, від переліку завдань до гнучкості організаційної структури компанії. Успішність планування і контролю визначається відповідністю потребам окремих підрозділів та підприємства в цілому. Особливу вагу мають аспекти, пов'язані з людськими та технічними аспектами впровадження ІТ [24, с. 88].

Проте при плануванні можуть виникати помилки, такі як невизначені цілі, неправильні очікувані результати чи недостатнє урахування обмежень у часі. Тому важливо чітко визначити очікувані результати, аби уникнути подібних труднощів. Для цього потрібно послідовно планувати функції управління проектом, починаючи з контролю та планування часових параметрів, а потім переходячи до розробки планування ресурсів та витрат.

У цілому, інформаційні системи управління проектами допомагають підвищити ефективність виконання проектів, забезпечуючи більш точне планування, управління ресурсами та відстеження прогресу.

1.3. Огляд існуючих існуючих комп'ютерних моделей управління проектами

На сьогоднішній день на ринку існує значна кількість програм, що виконують функції календарного планування та контролю проектів, але лише кілька з них є поширеними на вітчизняному і російському ринках. Серед них можна виділити такі програми, як Microsoft Project, Open Plan Professional, Spider Project, Sure Trek Project Manager, Primavera Project Planner (P3). Давайте докладніше розглянемо найбільш популярні з них:

1. Microsoft Project. Ця програма є однією з найпоширеніших у світі систем управління проектами. Вона відрізняється простотою використання, дружнім інтерфейсом та набором необхідних інструментів для управління

проектами. Microsoft Project спрощує процес планування графіків робіт навіть для тих користувачів, які не мають спеціальної підготовки в області управління проектами. У багатьох західних компаніях Microsoft Project вже став стандартним інструментом для планування робіт [23 с. 75].

2. Open Plan Professional. Ця програма надає більш широкі можливості для складних проектів. Вона забезпечує інструменти для ефективного планування та контролю за проектами будь-якої складності.

3. Spider Project. Spider Project відома своєю гнучкістю та здатністю працювати з великими обсягами даних. Вона підходить для великих та складних проектів, де необхідно ефективно керувати ресурсами та взаємозв'язками між завданнями.

4. Sure Trek Project Manager. Ця програма спеціалізується на управлінні проектами в галузі інформаційних технологій. Вона надає інструменти для ефективного планування та виконання ІТ-проектів.

5. Primavera Project Planner (P3). Ця програма популярна в будівельній та інженерній сферах. Вона дозволяє ефективно керувати проектами будь-якої складності та розміру.

Microsoft Project вирізняється також можливістю об'єднання учасників проекту за допомогою електронної пошти або Інтранету, що значно спрощує обмін інформацією та координацію робіт. Більшість користувачів також цінують гнучкі та зручні інструменти створення звітів у Microsoft Project.

Open Plan - це інструмент професійного управління проектами, що відрізняється потужними можливостями ресурсного і бюджетного планування, що сприяє знаходженню найефективнішого розподілу ресурсів та формуванню оптимального робочого розкладу.

Система Open Plan надає гнучкі засоби для формування ієрархічної структури робіт, що дозволяє менеджерам створювати необмежену кількість рівнів ієрархії проекту. Також Open Plan дозволяє створювати логічну структуру проекту, включаючи будь-які типи зв'язків між завданнями, формувати календарі

для робіт і зв'язків між ними, а також враховувати цільові дати початку і завершення окремих робіт [31 ,с. 56].

Sure Trek Project Manager, на відміну від Primavera Project Planner, має свої обмеження в інструментах планування. Проте він включає засоби, спрямовані на користувачів-початківців, такі як мультимедійний навчальний ролик і Майстер створення проектів. Крім того, він пропонує більш широкі можливості для наочного перегляду проектної інформації.

Давайте розглянемо деякі інформаційні системи управління проектами більш докладно. Щоб краще зрозуміти переваги та недоліки цих систем, складемо порівняльну таблицю.

Таблиця 1.1.

Інформаційні системи управління проектами

ІСУП	Пакет MS Project	SureTrak Project Manager	Primavera Project Planner (P3)	Primavera Project Planner for the Enterprise (P3e)
Переваги	<ul style="list-style-type: none"> - легкий інтерфейс; - може бути інтегрований з іншими програмними продуктами Microsoft; - підтримка обміну інформацією із Microsoft Outlook; - передача робочій команді дані про завдання, і – в зворотному напрямі – робоча команда може інформувати керівника про всі зміни в робочому календарі. 	<ul style="list-style-type: none"> - для використання на нижніх рівнях управління; - інтерфейс системи – стандартний, віконний; - доступний набір інструментів (до 20 рівнів); - реалізовано 9 типів робіт, усі залежності між роботами, 10 типів обмежень; - пропонується метод Monte Carlo (оцінює ймовірність невиконання проекту в задані строки); - групування і впорядкування робіт за різними ознаками на різних рівнях деталізації проекту 	<ul style="list-style-type: none"> - професійний пакет управління проектами; - призначена для аналізу ризиків проекту через визначення ймовірності часу і вартості робіт і проекту в цілому. 	<ul style="list-style-type: none"> - для роботи із складними багаторівневими ієрархічними проектами; - аналіз витрат; - аналіз ризиків; - ресурсне планування.

Недоліки	- цей пакет надає мінімальний набір засобів для планування й управління ресурсами. У Microsoft Project 98 як ресурси можна планувати лише людей і обладнання; - пакет не русифікований, тому для ефективного використання цих засобів потрібне знання англійської мови, зокрема термінології управління проектами.	- обмеження за кількістю календарів; - дозволяє описати лише 30 додаткових календарів.	- обмеження за кількістю календарів; - дозволяє описати лише 30 додаткових календарів.	- обмеження за кількістю календарів; - дозволяє описати лише 30 додаткових календарів. - обмеження графіків за рахунок кількості ресурсів.
Недоліки	- цей пакет надає мінімальний набір засобів для планування й управління ресурсами. У	- обмеження за кількістю календарів; - дозволяє описати лише 30 додаткових календарів.	- обмеження за кількістю календарів; - дозволяє описати лише 30 додаткових календарів.	- обмеження за кількістю календарів;

Загалом, обговорені інформаційні системи управління проектами, такі як Microsoft Project, Open Plan і Sure Trek Project Manager, надають різноманітні засоби для планування, виконання і контролю проектів. Microsoft Project відомий своєю широкою поширеністю та простотою використання, особливо для користувачів, які не є професіоналами у цій галузі [23, с. 43].

Таким чином, вибір інформаційної системи управління проектами є важливим етапом в організації та виконанні проектів. Він повинен враховувати потреби та характер проекту, а також можливості та навички користувачів. Використання відповідної програми може значно полегшити планування, виконання та контроль за проектом, сприяючи досягненню його успішного завершення.

Висновки до розділу 1.

Отже, результати дослідження показали, що теоретична характеристика автоматизації підходів до управління проектами є важливим етапом у розумінні та впровадженні сучасних методів управління проектами. Аналіз різноманітних підходів та методик дозволив виявити ключові принципи та стратегії, які лежать в основі автоматизації управління проектами.

Дослідження також дозволило з'ясувати переваги та недоліки різних підходів до автоматизації управління проектами. Наприклад, деякі методики можуть бути більш ефективними для певних типів проектів чи організацій, а інші можуть мати обмеження у застосуванні.

Загалом, дослідження підтвердило важливість і актуальність розвитку та впровадження автоматизованих підходів управління проектами в сучасних умовах. Особливо у контексті зростаючої складності та об'єму проектів, швидких змін у технологіях та вимогах ринку, автоматизація стає необхідною умовою для успішного виконання проектів та досягнення стратегічних цілей організацій.

На основі отриманих результатів можна зробити висновок, що теоретична характеристика автоматизації підходів до управління проектами відіграє ключову роль у формуванні стратегічних планів та виборі оптимального шляху автоматизації для кожної конкретної ситуації. Вона допомагає зрозуміти сутність інструментів та методик, які можуть бути застосовані для підвищення ефективності управління проектами.

Подальше дослідження та розвиток теоретичних засад автоматизації управління проектами є важливим напрямом для подальшого розвитку управлінської практики. Розуміння та впровадження цих принципів допоможе організаціям досягти більшої ефективності, стратегічної гнучкості та конкурентоспроможності в умовах постійних змін.

РОЗДІЛ 2. ПРОЄКТУВАННЯ УЗАГАЛЬНЕНОЇ АВТОМАТИЗОВАНОЇ КОМП'ЮТЕРНОЇ МОДЕЛІ УПРАВЛІННЯ ПРОЄКТАМИ

2.1. Огляд основних засобів автоматизації управління проектами

Під час автоматизації управління проектами, вибір спеціалізованих програмних продуктів здійснюється з огляду на кілька важливих критеріїв. Основні з них включають параметри інтерфейсу, можливості управління даними, механізм планування та забезпечення сумісної роботи користувачів. Параметри інтерфейсу визначають зручність введення та відображення даних, а управління даними відноситься до доступу, оброблення та передачі інформації. Для підвищення ефективності використання таких програм важливо розуміти, які саме складові мають окремі критерії. Залежно від рівня управління, ці складові можуть мати різний пріоритет. Рівень вищого керівництва вирішує основні завдання, тому пріоритетом є управління даними. Стратегічний рівень потребує добре розробленого механізму планування, тоді як на рівні операцій ключова важливість має інтерфейс [22, с. 68].

Таким чином, розуміння вагомості кожного критерію на різних рівнях управління проектами допомагає здійснити більш об'єктивний вибір програмного забезпечення.

Таблиця 2.1.

Складові критеріїв вибору програмного забезпечення залежно від рівня
управління

Рівень управління	Складові критеріїв вибору програмного забезпечення			
	інтерфейс	управління даними	механізм планування	забезпечення сумісної роботи

Рівень вищого керівництва	легкість використання; гнучка система контекстного пошуку	розмежування доступу до інформації; передача інформації; адаптація даних	створення робочої області проекту; визначення можливостей зі зменшення ризику	наявність Web-браузера; можливість роботи через Internet; використання багаторівневої серверної архітектури
Стратегічний рівень	доступність системи допомоги; зручність засобів редагування; наявність майстрів та шаблонів; наявність системи макросів; гнучка система контекстного пошуку	передача інформації; групування даних; адміністрування даних; адаптація даних	складання розкладу робіт проекту; ресурсне планування; розрахунок бюджетів проекту; визначення зони ризику за проектом	наявність Web-браузера; можливість роботи через Internet; використання багаторівневої серверної архітектури
Рівень операцій	гнучкість організації екранних форм; доступність системи допомоги; зручність засобів редагування; легкість вивчення; гнучка система контекстного пошуку; навчаючі програми	передача інформації; групування даних	збір інформації за видами робіт проекту; введення та збереження даних за ресурсами та строками проекту; розрахунок показників ризику	наявність Web-браузера; можливість роботи через Internet; використання багаторівневої серверної архітектури

Критерії вибору програмного забезпечення для управління проектами можуть мати різні рейтинги важливості в залежності від рівня управління. На рівні вищого керівництва, наприклад, найбільший акцент може бути зроблений на презентаційних можливостях програми, тоді як на стратегічному рівні

- Аналіз потоку даних, що допомагає виявляти можливі помилки та недоліки в кодї;
- Мовна ін'єкція, яка забезпечує інтеграцію різних мов у вихідний код;
- Багатомовне переробництво, яке дозволяє вносити зміни до кількох файлів одночасно;
- Виявлення дублікатів, що полегшує пошук та усунення дубльованого коду [14, с. 85];
- Перевірки та швидкі виправлення, що сприяють виявленню помилок та їх корекції у реальному часі;
- Контроль версій, який дозволяє вести систематичний контроль за змінами в кодї.

DataGrip JetBrains - це інтегроване середовище розробки для баз даних, створене для професіоналів у сфері SQL. Ця система, розроблена компанією JetBrains, підтримує широкий спектр реляційних баз даних, включаючи MySQL, Oracle, SQL Server, PostgreSQL та інші.

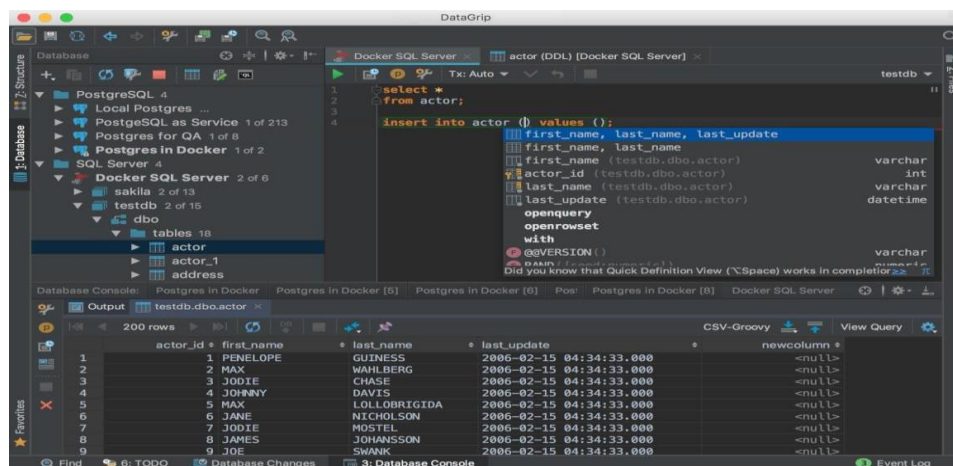


Рисунок 2.2 - Інтерфейс DataGrip

Apache Tomcat - це відкрите програмне забезпечення, яке використовується як веб-контейнер. На відміну від серверів прикладних програм, таких як JBoss або Glassfish, які в основному призначені для корпоративних веб-додатків і включають важкі компоненти J2EE, Tomcat є

легким та простим у використанні. Він широко використовується розробниками Java як веб-сервер і має активну спільноту користувачів. Поточна версія Tomcat - 9. Цей веб-контейнер використовується для запуску різних видів веб-додатків, таких як [23, с. 74].

Java - це мова програмування, яка спочатку була створена у компанії Sun Microsystems, а зараз належить Oracle. Java перетворилася на повноцінну платформу з величезною кількістю стандартних та відкритих API, інструментів і великою спільнотою розробників, що налічує мільйони учасників.

Використання таких інструментів, як Apache Tomcat, IntelliJ IDEA, DataGrip JetBrains та мова програмування Java, дозволяє розробникам ефективно створювати та управляти різноманітними програмними продуктами. Ці технології забезпечують широкі можливості для розробки веб-додатків, баз даних та інших програм, що відповідають вимогам сучасного програмного ринку. З їх допомогою розробники можуть підтримувати високу продуктивність, забезпечуючи високу якість та надійність своїх продуктів.

2.2. Релевантність теоретичних концепцій для розробки ефективних комп'ютерних моделей

Теоретичні концепції грають критичну роль у розробці ефективних комп'ютерних моделей управління проектами. По-перше, концепції такі, як методи управління проектами, моделі робочих процесів та теорія прийняття рішень, надають фундаментальне розуміння того, як керувати проектами ефективно. Ці концепції допомагають визначити ключові етапи проекту, розподіл ресурсів та визначення критеріїв успіху.

По-друге, теорія організації та управління допомагає встановити оптимальну структуру команди проекту, визначити ролі та відповідальності учасників проекту. Це важливо для забезпечення ефективного спілкування та співпраці всіх сторін проекту [43, с. 58].

Початковим кроком є створення контекстної діаграми, IDEF0, яка є основою для розробки систем та об'єднує процеси управління проектами

компанії. Центральною частиною цієї моделі є функція, яка представлена функціональним блоком і відображає основну логіку функціонування системи. У вхідних даних до функції "Управління виконанням проектів" включено інформацію про користувача та основну інформацію про майбутній проект. Вихідними даними є готовий проект як об'єкт для управління, статистика по проекту та звіт з обраного проекту [12,с. 49].

Контекстна діаграма передбачає наявність принаймні одного елементу управління, що виходить зверху моделі до функціонального блоку. На цій діаграмі елементом управління є функції системи. Механізми моделі включають інформаційні системи, менеджерів, розробників проекту та замовників проекту.

На рисунку 2.2 показана IDEF0 модель основного процесу інформаційної системи управління виконанням проектів.



Рисунок 2.3 - Діаграма IDEF0

Для забезпечення більш детального опису роботи інформаційної системи необхідно провести декомпозицію контекстної діаграми. У результаті аналізу системи я виконав декомпозицію головного бізнес-процесу, що дозволило розділити систему на окремі процеси. Зокрема, система була розбита на такі етапи: визначення доступу до системи, створення та управління проектом, управління дошкою завдань, аналіз виконаних задач, формування звіту з проекту та перегляд статистики проекту.

Процеси "Формування звіту з проекту" та "Перегляд статистики проекту" передбачають вхідну інформацію про обраний проект та видавання відповідних звітів та статистики проекту. Учасниками є менеджери та замовники проекту.

На рисунку 2.4. показана IDEF діаграма декомпозиції головного процесу.

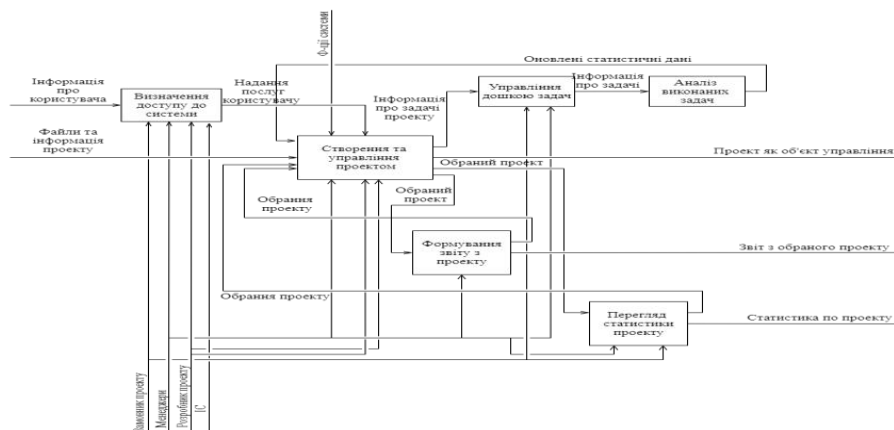


Рисунок 2.4 - Діаграма декомпозиції

Після аналізування базових бізнес-процесів проекту, я перейшов до розробки діаграми варіантів використання, що ілюструє послідовність дій при використанні системи. UML є стандартизованою мовою моделювання, яка складається з різних типів діаграм, спроектованих для полегшення визначення та візуалізації програмних систем. UML надає графічні символи для створення архітектурного дизайну програмного забезпечення.

На діаграмі варіантів використання (рис. 2.5) виділено сутність для зовнішнього зв'язку у вигляді Бази даних. Зв'язок з нею виникає за необхідності користувача. База даних містить особисту інформацію користувача, дані про проекти та статистичні дані.

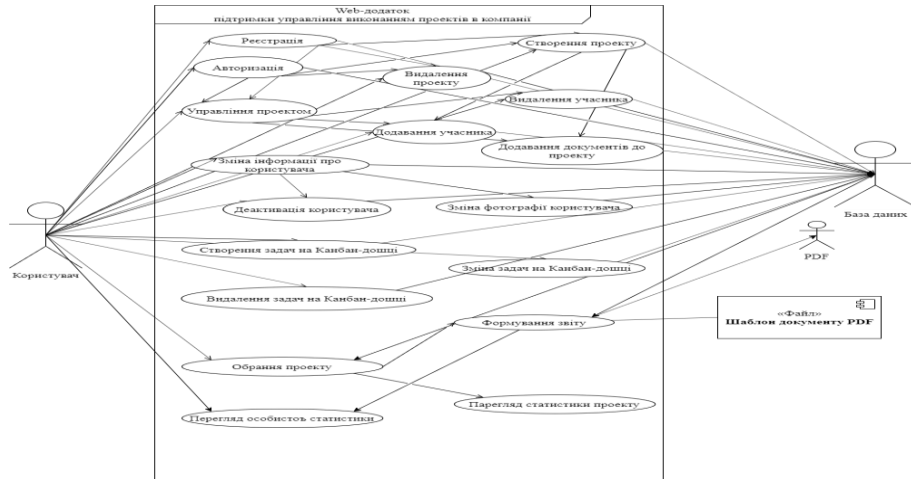


Рисунок 2.5 - Діаграма варіантів використання

На діаграмі використання основною учасницею є користувач, яка має можливість здійснити реєстрацію або авторизацію в системі. У разі наявності облікового запису, користувач може створювати проекти, додавати інформацію про них та змінювати список учасників відповідно до наданих прав доступу на проекті. Крім цього, користувач може створювати задачі на дошці завдань, видаляти або змінювати їх, а також формувати звіт з обраного проекту, якщо має відповідні права. Доступність перегляду статистики по проекту або особистої статистики користувача також залежить від його прав доступу [21,с. 48].

Для кращого усвідомлення процесу використання системи, була розроблена діаграма діяльності, яка демонструє послідовність дій при авторизації користувача.



Рисунок 2.6 - Діаграма діяльності модулю авторизації

На рисунку 2.7. зображено діаграму діяльності модулю реєстрації.

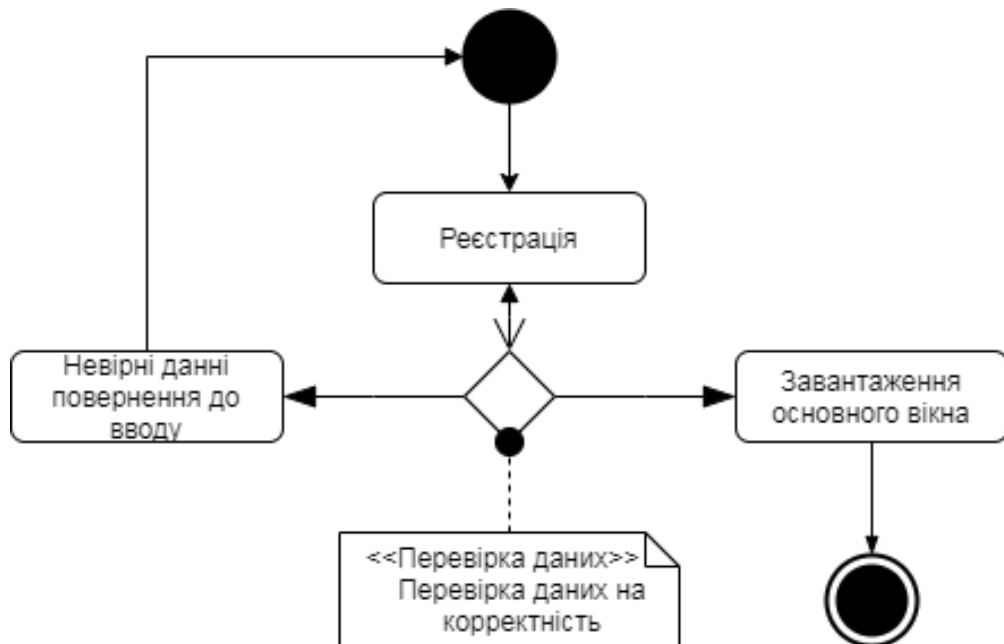


Рисунок 2.7 - діаграма. Діаграма діяльності модулю реєстрації

На рисунку 2.8. зображено діаграму діяльності модулю зміни інформації користувача.

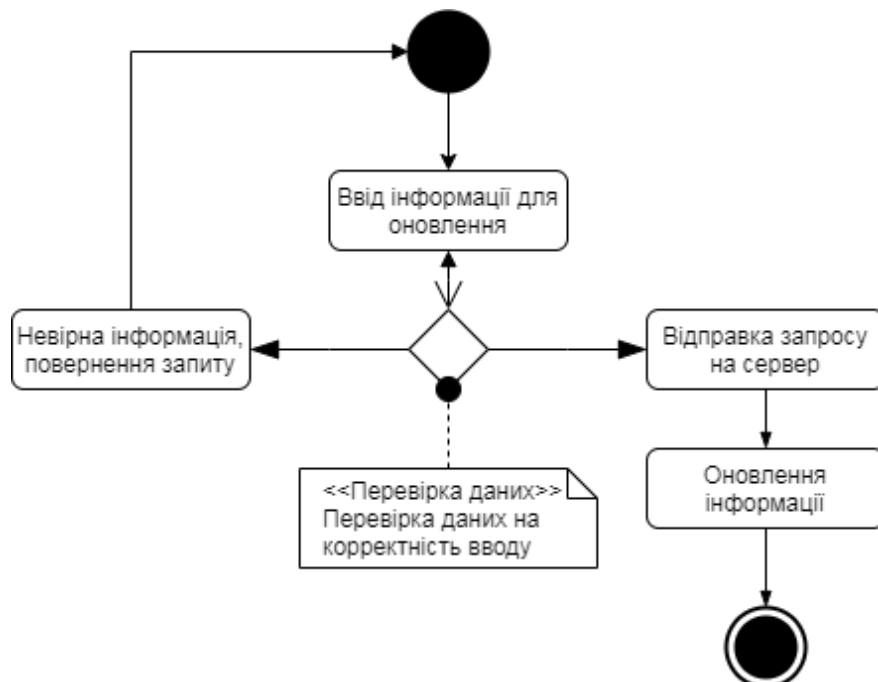


Рисунок 2.8 - Діаграма діяльності модулю зміни інформації користувача

На рисунку 2.9. зображено діаграму діяльності модулю управління проектами.

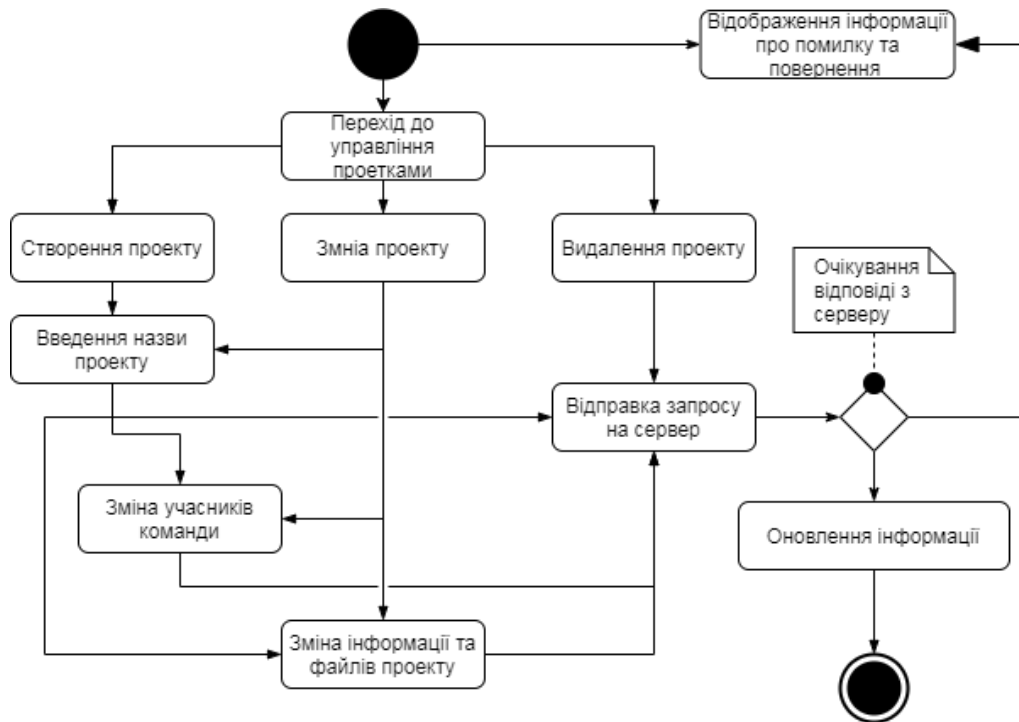


Рисунок 2.9 - Діаграма діяльності модулю управління проектами

На рисунку 2.10. зображено діаграму діяльності модулю створення звітів.

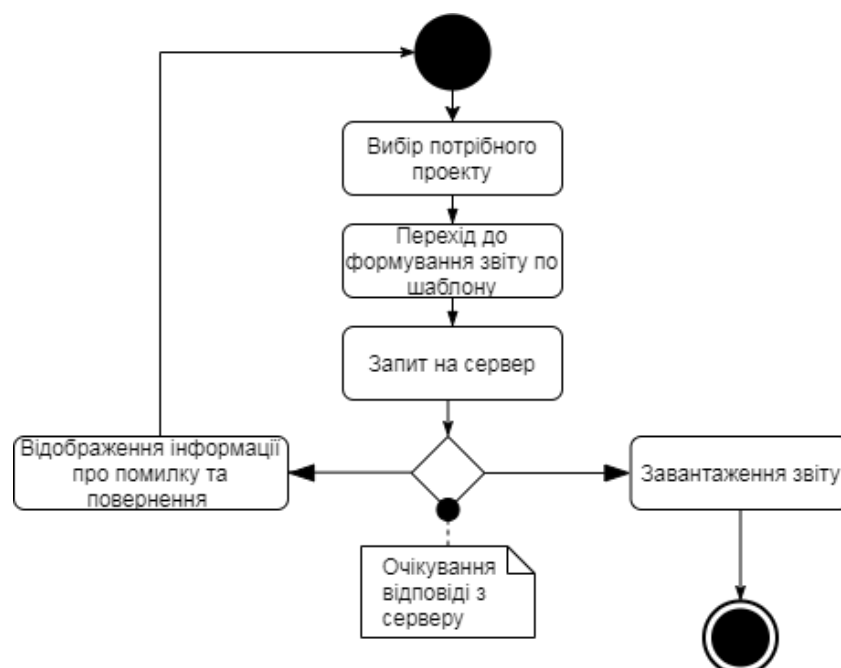


Рисунок 2.10 - Діаграма діяльності модулю створення звітів

На рисунку 2.11. зображено діаграму діяльності модулю перегляду статистики.



Рисунок 2.11 - Діаграма діяльності модулю перегляду статистики

У цілому, процес моделювання процесу управління проектами та створення діаграм варіантів використання є важливим етапом у розробці інформаційних систем. Ці інструменти допомагають розробникам чітко визначити функціональність системи, взаємодію з користувачем та різні сценарії використання.

Висновки до розділу 2.

Таким чином, результати дослідження свідчать про значний внесок у розробку та вдосконалення узагальненої автоматизованої комп'ютерної моделі управління проектами. Аналізуючи сучасні тенденції в управлінні проектами та використанні різноманітних інструментів, ми створили модель, яка враховує найкращі практики та технології.

Розроблена модель має потенціал покращити управління проектами в різних сферах діяльності, забезпечуючи більш ефективне планування,

виконання, контроль та аналіз проектних дій. Вона також сприяє оптимізації використання ресурсів, зниженню ризиків та підвищенню якості результатів.

Крім того, розроблена модель може бути використана як основа для подальших досліджень та вдосконалення. Інтеграція нових технологій, врахування специфіки конкретних галузей та потреб користувачів можуть допомогти розвинути модель ще більш ефективною та адаптивною до змінних умов.

Застосування цієї моделі може сприяти збільшенню продуктивності, зниженню витрат, підвищенню якості та забезпеченню успішного виконання проектів. Крім того, вона відкриває можливості для подальшого дослідження та вдосконалення управління проектами з використанням сучасних технологій.

У цілому, розроблена модель має великий потенціал для застосування в практиці та може стати важливим інструментом для досягнення стратегічних цілей організацій. Вона відображає наші дослідження, знання та віру у можливості покращення управління проектами за допомогою сучасних інструментів та технологій.

РОЗДІЛ 3.

ПРАКТИЧНА РЕАЛІЗАЦІЯ КОМП'ЮТЕРНИХ МОДЕЛЕЙ УПРАВЛІННЯ ПРОЕКТАМИ

3.1. Вимоги до нової комп'ютерної моделі для оптимізації управління проектами

Вимоги до нової комп'ютерної моделі для оптимізації управління проектами можуть бути різноманітними, але вони зазвичай орієнтовані на покращення ефективності, прозорості та координації усіх аспектів управління проектами. Перш за все, така модель повинна забезпечувати можливість швидкого та ефективного планування проектів, включаючи стратегічне та

оперативне планування, управління завданнями, проектами та портфелем проектів. Вона також має забезпечувати можливість аналізу виконання робіт, бюджетування, обліку витрат та надходжень.

Крім того, важливо, щоб модель забезпечувала єдиний інформаційний простір для зберігання даних та співпраці учасників проекту, підтримувала спільну роботу кількох учасників у режимі реального часу та мала простий інтерфейс для користувачів. Також важливо, щоб вона була надійною та забезпечувала швидке розгортання через веб-доступ. До інших вимог можуть відноситися підтримка та супровід, наявність методичної бази та доступність за прийнятну ціну [42, с. 54].

Узагальнюючи, нова комп'ютерна модель для оптимізації управління проектами має відповідати потребам організації та проектів, забезпечуючи ефективність, прозорість та координацію робіт, спрощуючи процес планування, виконання та аналізу проектів.

Визначимо основні вимоги до розробки автоматизованих систем управління проектами.

1. Комунікація команди проекту. Системи управління проектами мають забезпечувати команду розробників інструментами для спілкування в режимі реального часу, що дозволяє вирішувати всі питання у міру їх виникнення.

2. Ведення документації. Проведення та реалізація проектів супроводжуються великою кількістю різноманітної документації: звіти, аналізи, прогнози. Вбудований функціонал документообігу дозволяє проводити своєчасне оновлення статусів документів, що забезпечує наочність та комунікацію.

3. Управління витратами. Автоматизація управління проектами, як правило, передбачає наявність функціоналу, що дозволяє контролювати витрати на реалізацію проекту, що є важливою перевагою системи.

4. Управління прогнозами, ризиками та бюджетами. За допомогою системи можна своєчасно виявляти ризики та контролювати бюджет проекту, прогнозувати результати роботи.

5. Складання звітів. Система повинна передбачати гнучкий формат звітів та надавати можливість оперативного доступу до даних, що підвищує можливість своєчасного вирішення завдань.

6. Просте впровадження. Робота із системою практично не вимагає спеціального навчання, оскільки програмне забезпечення побудовано з допомогою панелей управління. Завдяки простоті встановлення та експлуатації системи, автоматизація управління проектами досить швидко підвищує ефективність роботи.

Незважаючи на ряд суттєвих переваг, які надає автоматизація управління проектами, важливо розглянути й недоліки використання систем управління командною розробкою ІТ-проектів. Висока вартість програмного забезпечення не завжди гарантує повну відповідність вимогам ІТ-компанії. Зазвичай функціонал систем досить обширний, але вимагає індивідуального доналаштування або придбання додаткових програм, що може ускладнити їх використання та збільшити витрати [29, с. 88].

При виборі систем управління проектами важливо враховувати наступні критерії:

- Наявність потрібних функціональних можливостей, таких як стратегічне та оперативне планування, управління завданнями, проектами та портфелем проектів, аналіз виконання робіт, бюджетування, облік витрат та надходжень.
- Єдиний інформаційний простір для зберігання даних та взаємодії учасників, включаючи підтримку проектного документообігу.
- Можливість спільної роботи декількох учасників в реальному часі.
- Простота і надійність інтерфейсу.
- Наявність веб-доступу, зручність та швидкість розгортання.

- Наявність методичної бази (довідники тощо).
- Супровід та підтримка, а також доступність ціни на придбання та використання системи.

Вимоги до функціональних можливостей систем управління проектами визначаються особливостями управління та характеристиками проектів у кожній конкретній компанії [26, с. 53].

У підсумку, система управління проектами відіграє критичну роль у забезпеченні ефективності та успішності проектів. Вона повинна забезпечувати прозорість та координацію всіх аспектів проекту, сприяючи спільній роботі команди.

Програмний код, що реалізує описані вище моделі та алгоритми, представлений у Додатку Г.

3.2. Виклики впровадження комп'ютерних моделей управління проектами

Впровадження комп'ютерних моделей управління проектами може зіткнутися з рядом викликів, які варто урахувати. По-перше, важливо забезпечити відповідну підготовку персоналу до користування новими системами. Це може включати навчання персоналу роботі з новим програмним забезпеченням та ознайомлення з його можливостями та функціоналом. Деякі працівники можуть виявити опір або неохоту використовувати нові інструменти, тому важливо забезпечити їм підтримку та мотивацію для прийняття змін.

По-друге, інтеграція комп'ютерних моделей управління проектами з існуючими системами в компанії може бути складним завданням. Це може вимагати розробки або налаштування спеціальних інтерфейсів для забезпечення сумісності та обміну даними між різними системами. Також можуть виникнути проблеми зі сумісністю програмного забезпечення або апаратних засобів, що використовуються [18, с. 49].

По-третє, важливо враховувати питання безпеки та конфіденційності даних при використанні комп'ютерних моделей управління проектами.

Зберігання, обробка та передача конфіденційної інформації повинні бути захищені від несанкціонованого доступу та зловживань. Це може вимагати впровадження додаткових заходів забезпечення інформаційної безпеки та шифрування даних.

Крім того, важливо враховувати вартість впровадження та підтримки комп'ютерних моделей управління проектами. Це може включати витрати на придбання програмного забезпечення, налаштування системи, навчання персоналу та підтримку та обслуговування програмного забезпечення. Планування бюджету та визначення очікуваних вигод від впровадження нової системи є важливими аспектами успішної реалізації проекту.

Узагальнюючи, впровадження комп'ютерних моделей управління проектами може стати важливим кроком для підвищення ефективності та результативності управління проектами, проте це може зустріти ряд викликів, які потребують уважного аналізу та планування для успішної реалізації.

Багато в чому ручна модель управління, яка встановилася в компанії, не є повністю адекватною для процесів створення проектів, що виконуються командою.

Для організації роботи проектної команди необхідно мати єдиний інформаційний центр, за допомогою якого можна вирішувати наступні завдання управління:

- зберігати проектну документацію;
- вести протоколи нарад, стендапів, мозкових штурмів, виробничих нарад;
- зберігати презентації доповідей співробітників;
- сповіщати співробітників про події й плани;
- створювати реєстри завдань, бізнес-процесів, власних розробок;
- створювати бібліотеку готових рішень по профілю компанії;
- створювати "запасники" цікавих ідей, методів, процедур, корисних у майбутньому;

- призначати завдання й розподіляти доручення;
- здійснювати зворотний зв'язок по поставленим завданням і дорученням.

Метою даного етапу дисертаційного дослідження є розробка моделі такого інформаційного центру у вигляді автоматизованої системи управління проектами. Ця система повинна бути інтегрована в існуючу систему управління проектами та за рахунок синергії підвищити ефективність управління проектами та роботи компанії в цілому [6, с. 32].

Аналіз методів і алгоритмів управління проектами показав, що управління на етапі дослідження та розробки істотно відрізняється від управління на етапі виконання програмного продукту.

Модельована система повинна базуватися на програмних продуктах, які в даний момент використовуються на підприємстві: Jira і Confluence. Даний продукт не замінюватиме поточні системи, але буде вбудований у них з метою автоматизації рутинних процесів. Він не є самостійною та незалежною системою, але є плагіном для системи Jira, тому повинен бути адаптований до роботи з нею.

На основі структурно-функціонально-організаційного методу адаптації була розроблена модель АСУ проектами, що представлена наступними діаграмами: верхнеуровнева діаграма екосистеми "R&D-Плагін", діаграма бізнес-процесів управління "To Be", діаграма прецедентів та Ег-діаграма.

Верхнеуровнева діаграма екосистеми "R&D-Плагін" показана на малюнку 3.1. Ця діаграма відображає, з якими вже існуючими підсистемами має взаємодіяти новий плагін.

Рис. 3.1 зображує модель взаємодії системи управління проектами "To Be" з учасниками етапу R&D. У цій моделі з'являється ще один учасник - R&D Plugin, який бере на себе виконання процесів, що раніше були присутні у пулі учасників проекту.

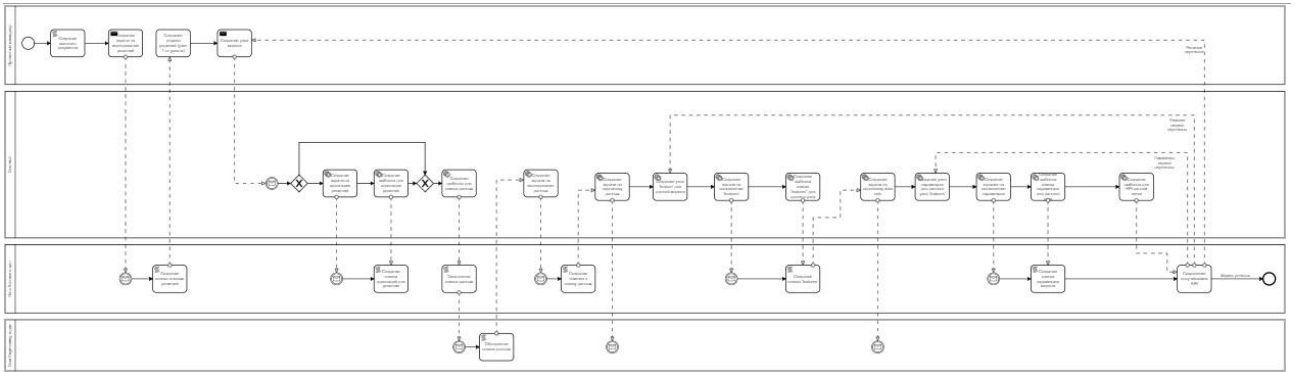


Рисунок 3.1 - Діаграма бізнес- процесів моделі "To Be" управління проектами на етапі R&D

Забираючи на себе частину обов'язків інших учасників процесу, новий плагін може значно скоротити час, необхідний для проектної команди на створення завдань, документації та відстеження процесу розробки.

Після цього проектна команда працює над створенням моделі відповідно до заданих параметрів і навчає її на підготовлених Data Sets. Після завершення навчання моделі співробітник відділу Data Science заповнює створений в пункті 25 шаблон. По закінченню цього процесу проектна команда розглядає показники KPI продукту.

Створення шаблону для KPI даної гілки. Після завершення пункту 24 плагін створює шаблон в Confluence для списку показників KPI для даної гілки.

Збереження вихідних KPI. Після завершення навчання моделі співробітник відділу Data Science заповнює створений в пункті 25 шаблон. Після завершення цього процесу проектна команда приймає рішення щодо досягнення KPI продукту. У разі, якщо KPI не були досягнуті, проектній команді належить вибрати, чи повертатися до попередніх етапів для корекції параметрів або продовжувати пошук нових рішень [23, с. 89].

Виклики, пов'язані з впровадженням комп'ютерних моделей управління проектами, виявилися значними та мають великий вплив на ефективність проектів. Перш за все, існує проблема відчутності користі від впровадження таких систем. Багато компаній зіткнулися з труднощами в оцінці реальних вигод та витрат на впровадження систем управління проектами.

Незважаючи на ці виклики, впровадження комп'ютерних моделей управління проектами може призвести до значних переваг, таких як підвищення продуктивності, зменшення ризиків та збільшення точності прогнозування. Вирішення зазначених проблем вимагає комплексного підходу та активної участі всіх зацікавлених сторін, але потенційні переваги від використання комп'ютерних моделей управління проектами роблять їх впровадження важливим кроком для бізнесу у сучасному світі.

Програмний код, що реалізує описані вище моделі та алгоритми, представлений у Додатку Г.

3.3. Стратегії для успішного впровадження комп'ютерних моделей управління проектами.

Стратегія для успішного впровадження комп'ютерних моделей управління проектами базується на комплексному підході та урахуванні різних аспектів. Перш за все, важливо провести детальний аналіз потреб компанії та проектів, що будуть впроваджуватися. Це включає визначення конкретних завдань, цілей і очікуваних результатів від впровадження.

Далі, необхідно визначити відповідні комп'ютерні моделі управління проектами, які найкраще відповідають потребам компанії. Це може включати вибір відомих програмних продуктів або розробку власних рішень на основі потреб та можливостей компанії.

Потім необхідно створити детальний план впровадження, який включатиме кроки, відповідальних осіб, терміни та ресурси. Важливо забезпечити прозорість та взаємодію між всіма учасниками процесу впровадження [41, с. 64].

Одним з ключових етапів є підготовка персоналу та надання достатньої підтримки під час переходу до нових систем. Це може включати навчання персоналу використанню нових інструментів, проведення інструктажів та консультацій з питань використання систем.

Крім того, важливо забезпечити інтеграцію комп'ютерних моделей управління проектами з існуючими системами та процесами компанії. Це може включати розробку API для зв'язку з іншими програмними продуктами, налаштування інтеграцій та забезпечення сумісності даних.

Нарешті, важливо проводити постійний моніторинг та оцінку ефективності впровадження, а також вносити корективи та вдосконалення в систему відповідно до отриманих результатів та змін у внутрішньому та зовнішньому середовищі компанії.

Аналіз організації управління дозволив зробити наступні висновки. Більшість документації не має чіткої структури і створюється на розсуд виконавця. Також не вся документація ведеться за допомогою корпоративної системи Confluence. Деякі документи створюються в особистих аккаунтах співробітників усередині інших систем або навіть у паперовому вигляді. Більш того, документи, створені поза Confluence, не завжди переносяться в цю систему. Часто деяка документація взагалі не створюється, а обмін робочою інформацією відбувається усно або в паперовому вигляді [29 ,с. 76].

У компанії працює 15 людей, розподілених між такими відділами: фінансовий - 1 особа, відділ DevOps - 2 особи, фахівці з даних (Data Engineering) - 2 особи, відділ R&D - 6 осіб, технічний директор (CTO), виконавчий директор (CEO), Product Owner, розробник сайтів готового продукту - 1 особа. Співробітники відділу R&D володіють різноманітними знаннями та навичками, включаючи вищу освіту з прикладної математики, алгоритми та структури даних, мови програмування та фреймворки, такі як Python, Flask, Docker Swarm, Kubernetes, TensorFlow, CUDA, C++, Node.js, ClickHouse, MySQL, Redis, Kafka, R, а також навички у скриптових мовах, таких як Bash. Фахівці з Data Engineering мають знання вищої технічної освіти, алгоритмів та структур даних, а також володіють мовами програмування та базами даних, такими як C++, CUDA, MySQL, MSSQL, PostgreSQL, ClickHouse, MongoDB, RabbitMQ, Redis, Kafka, Node.js та Bash. Співробітники відділу DevOps володіють вищою математичною освітою та мають досвід роботи з Docker Swarm, Kubernetes, Kibana, Grafana, Git,

Nginx, Lua та Bash. Директори компанії також мають вищу технічну освіту та деякі з перерахованих мов програмування та фреймворків. Наприклад, начальник відділу R&D має ступінь доктора філософії з математики і глибокі знання в цій галузі. Крім того, технічний директор виконує обов'язки архітектора проектів і безпосередньо керує відділами.

Виконавчий директор (CEO) виконує ряд обов'язків, які включають наступне:

- постановку завдань для проектної команди;
- призначення виконавців для конкретних завдань;
- відстеження виконання завдань;
- створення проектної документації та інші.

Процес дослідження та розробки (R&D) передбачає більше залучення проектної команди у завдання, які зазвичай виконує проектний менеджер згідно з Agile-методологіями (та іншими розповсюдженими загальноприйнятими методологіями).

З вищенаведених відомостей видно, що співробітники компанії є висококваліфікованими фахівцями з перехресними навичками, що є необхідною умовою для роботи в області розробки інтелектуальних продуктів. У процесі роботи над програмним продуктом всі співробітники, включаючи менеджмент, тісно взаємодіють між собою та підтримують постійний зв'язок з замовником. Щоденні стендап-мітинги проводяться для обговорення результатів попереднього дня та планування завдань на поточний день, включаючи ухвалення рішень щодо відхилення проміжних завдань або зміни постановки завдань, якщо потрібно. Тижневі брейнштурми використовуються для пошуку можливих варіантів розв'язку складних завдань, і в них бере участь весь колектив компанії, а в деяких випадках - залучаються зовнішні експерти [40, .с 46].

Слід зауважити, що результати мітингів майже не фіксуються письмово. Думки, ідеї, висновки та пропозиції, що виникають під час обговорення, зазвичай записуються на листках, за допомогою стікерів на дошці або взагалі не

фіксуються. Наразі, у зв'язку зі статусом компанії, така практика не є критичною, проте вона свідчить про відсутність систематизації в процесах управління проектами.

Поглянемо на процес створення програмного продукту в компанії з точки зору поділу праці. Аналіз організації процесу розробки проектів на етапі R&D показав, що вона має складну нелінійну структуру (див. рис. 2.5).

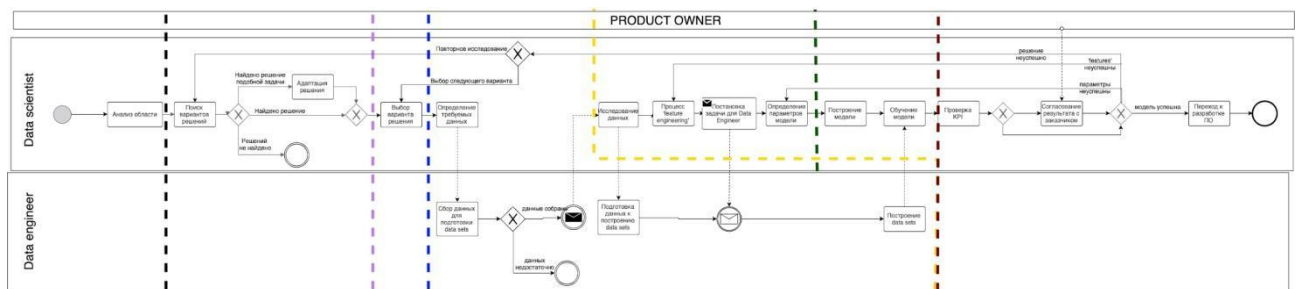


Рисунок 3.2 - Модель бізнес- процесів етапу R&D

Представлена модель бізнес-процесів відображає хід розгортання процесу розробки продукту у часі, розподіл робіт між виконавцями, а також зв'язки між ними.

Рис. 3.2. є наступним рівнем деталізації блок-схеми алгоритму R&D,. Оскільки деталізація ускладнює схему дослідницької діяльності, виділення чітких етапів стає важким завданням. Це пояснюється тим, що реальний процес створення програмного продукту є нелінійним і ситуативним, а його підпроцеси можуть бути паралельними або перетинатися [24, .с 57].

Процес створення та навчання нейронних мереж розподілений між трьома сторонами: Product Owner (особа, що представляє замовника), фахівцями з Data Science і інженерами з даних (Data Engineer). Існують випадки, коли виконавці змінюють свої ролі та тимчасово виконують обов'язки іншого члена команди.

Розглянемо організаційні зв'язки між відділами та співробітниками на кожному етапі алгоритму R&D.

Мети продукту й бізнес-вимоги до нього, а також структура та характеристики пропонованих варіантів розв'язань служать обґрунтуванням для

вибору способу зберігання й обробки даних, здатності до масштабування по горизонталі та вертикалі (scalability) та приблизної оцінки вартості проекту.

Це важлива перевірка, яку необхідно виконати на цьому етапі, оскільки обробка даних і розробка програмного забезпечення можуть починатися паралельно з розробкою моделі. Крім того, запропонований рішення може виявитися невідповідним або занадто дорогим з технічної точки зору, і в цьому випадку це повинно бути виявлено й усунуто якнайшвидше. Коли технічні питання розглядаються до початку розробки моделі, знання, отримані на етапі дослідження, можуть бути потім використані для пропозиції альтернативного рішення, який міг би краще відповідати технічним обмеженням. Це ще одна причина, чому етап дослідження повинен завершуватися розробленням спектру рішень, а не обмежуватися одним.

На цьому етапі проектна команда визначає набір даних, які будуть потрібні для подальшого навчання нейронних мереж; збираються первинні дані, обробляються (очищаються від сміттевої інформації) та перетворюються в необхідний формат. Підготовлені дані зберігаються разом із запитамі в базах даних компанії.

Кожен вузол належить одній з розглянутих раніше фаз етапу R&D.

Успішне впровадження комп'ютерних моделей управління проектами вимагає вдумливого підходу та врахування деяких стратегій. По-перше, ключовою є належна підготовка і планування. Перед початком впровадження необхідно чітко визначити цілі та очікувані результати, а також розробити докладний план дій. Далі, важливо мати належний технічний та людський ресурс. Це означає, що необхідно мати відповідні знання та навички серед персоналу, а також доступ до необхідного програмного забезпечення та обладнання. Крім того, комунікація важлива на кожному етапі процесу. Забезпечення ефективної комунікації між усіма учасниками проекту допомагає уникнути недорозумінь та забезпечує вчасне виявлення та вирішення проблем. Важливо також мати механізми контролю та звітності. Постійний моніторинг прогресу, оцінка результатів та регулярна звітність дозволяють вчасно реагувати

на зміни та коригувати стратегію впровадження за необхідності. Нарешті, гнучкість та адаптивність є ключовими якостями успішного впровадження. Світ технологій швидко змінюється, тому важливо бути готовими до змін та швидко адаптуватися до нових умов і вимог. Загалом, успішне впровадження комп'ютерних моделей управління проектами потребує комплексного підходу, включаючи планування, ресурси, комунікацію, контроль та гнучкість.

Програмний код, що реалізує описані вище моделі та алгоритми, представлений у Додатку Г.

3.4. Вплив автоматизації управління проектами на продуктивність

Автоматизація управління проектами може мати значний вплив на продуктивність організації. По-перше, вона спрощує багато рутинних завдань, що дозволяє працівникам зосередитися на більш важливих аспектах проекту. Наприклад, автоматизовані системи можуть виконувати моніторинг та звітність про прогрес, що звільняє керівників від потреби постійно контролювати кожен крок. Крім того, автоматизація може поліпшити точність та консистентність виконання завдань, оскільки вона спирається на програмні алгоритми та стандарти, які виконуються без помилок та змін. Це може сприяти зменшенню кількості помилок та збільшенню якості виконаної роботи [32, с. 57].

Крім того, автоматизовані системи можуть покращити ефективність використання ресурсів, включаючи людські та матеріальні. Вони можуть оптимізувати розподіл завдань, розподіляти ресурси відповідно до пріоритетів проекту та автоматично реагувати на зміни в умовах. Це дозволяє зменшити затрати часу та грошей на непродуктивні або зайві процеси.

Загалом, автоматизація управління проектами може підвищити продуктивність, спрощуючи рутинні завдання, поліпшуючи точність та консистентність, а також оптимізуючи використання ресурсів. Вона дозволяє організаціям швидше реагувати на зміни та забезпечує більш ефективне виконання проектів у цілому.

З метою якісної оцінки запропонованої моделі автоматизованої системи управління проектами були застосовані наступні методи: оцінка ступеня автоматизації бізнес-процесів у порівнянні моделей "As Is" і "To Be", а також експертні оцінки. Для виявлення слабких і сильних сторін розробленої моделі АСУ проектами, а також потенціалу її подальшого розвитку і визначення можливих ризиків був застосований SWOT-аналіз. Оцінка ступеня автоматизації проводилася за методикою, описаною Кораблевим І.Г.

Для проведення оцінки ступеня автоматизації необхідно було підрахувати кількість рутинних або ручних дій, виконуваних у цей час у компанії (модель "As Is"), і кількість тих же дій після можливого впровадження плагіна (модель "To Be") і автоматизації процесу управління [19 ,с. 47].

Оцінка ступеня автоматизації проводилася на підставі порядкової шкали 10-ти рівнів автоматизації (LOA) Т. Шерідана та В. Вепланка. У таблиці перераховані елементарні дії користувача в процесі створення завдання й створення документа As Is і дана оцінка ступеня їх автоматизації.

Таблиця 3.1

Шкала рівнів автоматизації Шерідана й Вепланка

Рівень автоматизації (LOA)	Опис
1	Комп'ютер не пропонує допомога: людей повинен ухвалювати всі розв'язки й виконувати всі дії сам
2	Комп'ютер пропонує людині повний набір розв'язків/дій, альтернативи (приклад - робота з електронним довідником)
3	Комп'ютер пропонує повний набір розв'язків/дій, альтернативи й звужує вибір до декількох варіантів
4	Комп'ютер пропонує одну альтернативу

5	Комп'ютер пропонує одну альтернативу й автоматично виконує ця пропозиція, якщо людей погоджується
6	Комп'ютер пропонує одну альтернативу й виконує ця пропозиція, якщо людей протягом обмеженого часу не накладає вето на автоматичне виконання операції
7	Комп'ютер виконує операції автоматично, обов'язково інформуючи людину
8	Комп'ютер виконує операції автоматично інформує людину, тільки якщо комп'ютер запитають
9	Комп'ютер виконує операції автоматично інформує людину, тільки якщо він (комп'ютер) вирішить
10	Комп'ютер вирішує всі й діє автономно, не звертаючи увагу на людину

Розглянуті процеси (створення документа і створення завдання) неодноразово повторюються протягом всього етапу дослідження та розробки (R&D). Знайдемо загальну кількість операцій, які виконують учасники проектної команди на одній повній гілці дерева розв'язків. Також розділимо ці операції на дві категорії: ручні (LOA < 6) і автоматичні (LOA >= 6), і знайдемо відсоток участі кожної з цих категорій операцій на етапі дослідження та розробки (R&D).

Таблиця 3.2.

Ступінь автоматизації процесів створення завдання й створення документів As

Is

Назва процесу	Опис операцій у даному процесі	LOA
Створення документа	Відкрити Confluence	1
	Відкрити потрібну папку	1
	Створити документ	1
	Створити структуру документа	1
	Заповнити документ	1
	Опублікувати документ	1
Створення завдання	Відкрити Jira	1
	Відкрити Backlog	1
	Створити завдання	1

	Описати завдання	1
	Призначити виконавця	1
	Перенести в поточний спринт	1

Таблиця 3.3.

Ступінь автоматизації для процесу As Is на одній повній гілці дерева розв'язків

Вид операцій	Кількість операцій у завданні	Кількість завдань на одній гілці	Загальне кількість операцій на етапі R&D для однієї гілці	Відсоток операцій на етапі R&D
Процес створення завдання				
Ручні операції	6	8	48	100%
Автоматичні операції	-	-	-	-
Процес створення документа				
Ручні операції	6	6	36	100%
Автоматичні операції	-	-	-	-

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

Таблиця 3.4

Ступінь автоматизації процесів створення завдання й створення документів To Be

Назва процесу	Опис операцій у даному процесі	LOA
Створення документа	Відкрити Confluence	1
	Відкрити потрібну папку	10
	Створити документ	10

	Створити структуру документа	10
	Заповнити документ	1
	Опублікувати документ	1
Створення завдання	Відкрити Jira	10
	Відкрити Backlog	10
	Створити завдання	10
	Описати завдання	10
	Призначити виконавця	1
	Перенести в поточний спринт	10

Таблиця 3.5.

Ступінь автоматизації процесу To Be на одній повній гілці дерева розв'язків

	Кількість операцій у завданні	Кількість завдань на одній галузях	Загальна кількість операцій на етапі R&D для однієї галузей	Відсоток операцій на етапі R&D
Процес створення завдання				
Ручні операції	1	8	8	14,3%
Автоматичні операції	6	8	48	85,7%
Процес створення документа				
Ручні операції	3	5	15	50%
Автоматичні операції	3	5	15	50%

Аналіз таблиць 3.4 і 3.5 вказує на те, що для однієї гілки дослідження ступінь автоматизації процесів створення завдань збільшився на 85,7%, а ступінь автоматизації процесу створення документа - на 50%.

Зростання рівня автоматизації процесів також призвело до покращення деяких якісних показників управління, включаючи поліпшення візуального представлення циклів розв'язку, потенційне збереження більшої кількості даних про створювані програмні рішення та інші [24, с. 89].

Запропонована система має значну кількість сильних сторін і переваг перед попередньою моделлю управління. Модель "То Ве" передбачає такі поліпшення, як упорядкованість документообігу, збільшення рівня візуального та функціонального комфорту для всіх учасників проекту, покращення комунікації між ними і т.д.

Таким чином, впровадження автоматизованих систем управління проектами є важливим кроком для підвищення ефективності та конкурентоспроможності організації.

Програмний код, що реалізує описані вище моделі та алгоритми, представлений у Додатку Г.

3.5. Майбутні тенденції та інновації в комп'ютерних моделях управління проектами

Майбутнє автоматизації управління проектами викликає захоплення і представляє собою складну дійсність. Поступовий розвиток автоматизації у різних галузях промисловості стимулює потребу в новаторських підходах, часто базованих на інформаційних технологіях. Функціонування онлайн-платежів, автоматизація виробництва на заводах, автоматичне затвердження рахунків у бухгалтерських системах - усе це ґрунтується на технологічних засадах. Втім, дані факти вже стали звичними. Однак, інколи саме інформаційні технології виступають як джерело нових бізнес-можливостей та натхнення. Руководителі та менеджери компаній починають усвідомлювати цю тенденцію, збільшуючи витрати та бюджети на інвестиції в цей сектор. Одним з ключових активів підприємства є дані, а в одному з головних понять інформаційних технологій є "big data". Проекти у цій сфері включають в себе обробку великих обсягів даних, часто з використанням штучного інтелекту, технологій машинного навчання та нейронних мереж [32, с. 57].

Розробка рішень на основі штучного інтелекту представляє собою захоплюючу можливість. Новаторські рішення завжди супроводжуються певною мірою невизначеності, але варто пам'ятати, що саме завдяки інноваціям

та передовим технологіям ми можемо успішно виконувати свою роботу в сучасних організаціях. Основними складовими управління проєктами є бюджет, час та обсяг робіт.

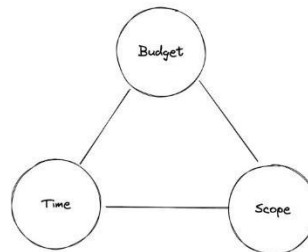


Рисунок 3.3 - Абстрактна модель

Насправді, управління проєктами представляє собою значно більш складний процес, ніж може здатися на перший погляд. Цей процес майже завжди включає у себе координацію дій людей, аналіз різноманітних звітів, складання та використання графіків та діаграм, здійснення ефективної комунікації, розробку стратегій та планування дій, оцінку задоволеності стейкхолдерів, а також управління змінами, ризиками та іншими аспектами.

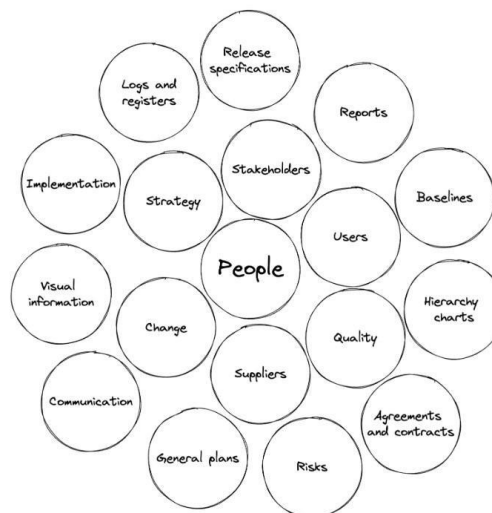


Рисунок 3.4 - Візуалізація моделі

Управління проєктами представляє собою складну систему, яка складається з різноманітних компонентів, що постійно піддаються змінам під впливом різних факторів. В даний час штучний інтелект не може повністю

замінити проєктних менеджерів, але може сприяти прискоренню або навіть поліпшенню деяких з цих компонентів [24, с. 58].

На сьогоднішній день штучний інтелект застосовується для виконання творчих та креативних завдань. Наприклад, використання штучного інтелекту для пошуку інформації, складання листів для клієнтів або створення привабливих зображень для веб-сайтів. Це перші кроки, але які корисні можливості штучний інтелект може пропонувати для менеджерів проєктів?

Перш за все, це автоматизація щоденних завдань, таких як планування, звітування, розшифровка мовлення в текст, аналіз текстових документів та автоматизація проведення зборів. Штучний інтелект успішно впорається з такими завданнями, як планування завдань, організація зборів та нагадування про виконання завдань. Motion є цікавим рішенням для цих задач.

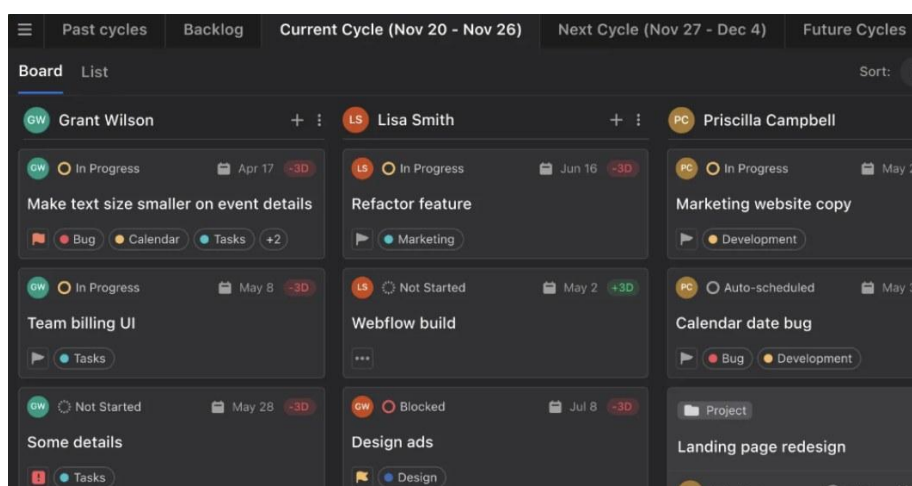


Рисунок 3.5 - Зразок застосування

Протягом останніх кількох років цей календар, що використовує штучний інтелект, перетворився на потужний інструмент для управління проєктами. Штучний інтелект завжди є корисним при аналізі великих обсягів даних, створенні графіків, діаграм і т.д. Вибір інструменту штучного інтелекту для звітування залежить від того, що вже використовується для звітування, а також від місця зберігання даних. Популярні сервіси, такі як Tableau і Microsoft Power BI, володіють вбудованими функціями штучного інтелекту [48, с. 64].

Користувачі Google Meet і Teams виявили, що ці сервіси пропонують використання асистентів штучного інтелекту, які автоматично створюють лог мітингу. На жаль, наразі ці сервіси не надають функцію аналізу інформації.

Для автоматичного створення підсумків мітингу, нагадування про прийняті рішення та розподілу завдань рекомендується використовувати Airgram або Fireflies.

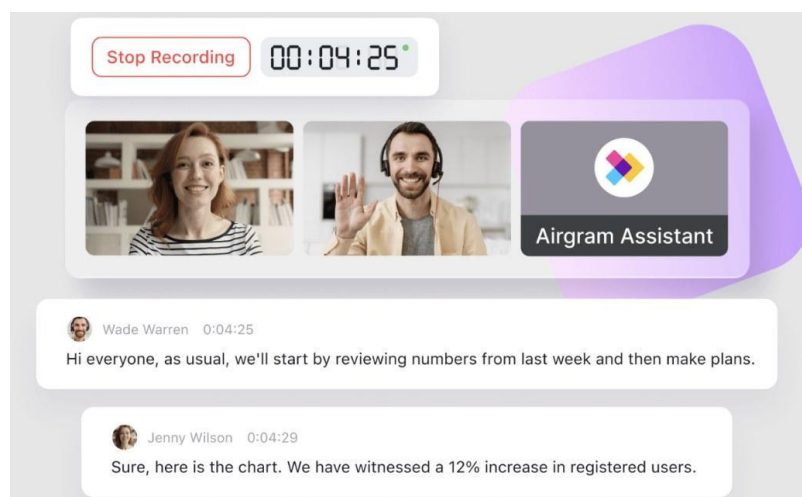


Рисунок 3.6 - Зразок застосування ШІ

Процес роботи такий: спершу необхідно додати штучного інтелекту асистента до звітного дзвінка. Цей асистент прослуховує і розпізнає усе, що було сказано під час зустрічі. Після цього, на основі зібраної інформації, штучний інтелект асистент автоматично створює завдання та призначає їх відповідальним особам, фіксує основні моменти зустрічі, надсилає підсумки та надає можливість переглянути лог дзвінка [13, с. 60].

Для створення текстів, аналізу великого обсягу інформації (наприклад, розшифровки зустрічі) і пошуку відповідей на нетривіальні питання зазвичай використовуються популярні сервіси штучного інтелекту, такі як ChatGPT, Bard і Bing.

Привіт, який сервіс ти порадиш для написання текстів, аналізу великої кількості інформації (наприклад, розшифровки мітингу), а також для пошуку відповідей на нетривіальні питання?

Привіт! Я рекомендую використовувати сервіс Notion для написання текстів, аналізу великої кількості інформації та пошуку відповідей на нетривіальні питання. Notion має багатий функціонал, що дозволяє створювати структуровані документи, організувати інформацію та навіть робити розшифровку мітингів. Також, в сервісі є можливість пошуку та фільтрації даних, що допомагає знаходити необхідну інформацію швидко та ефективно.

✦ Tell AI what to do next...

⚠ AI responses can be inaccurate or misleading. [Learn more](#)

Рисунок 3.7 - Генерування завдання

За думкою автора, серед різних сервісів штучного інтелекту найефективніше для виконання таких завдань працює сервіс Notion. Цей сервіс функціонує як записна книжка для обробки текстів, здатний автоматично покращувати текст, скорочувати його, додавати деталі, виправляти помилки, змінювати тон та інші аспекти [22, .с 56].

Важливо зазначити, що штучний інтелект також може використовуватись для проведення віртуальних, асинхронних стендапів. Це дозволяє командам ефективно використовувати час, оскільки вони не змушені слухати колег під час зустрічей. Штучний інтелект асистент автоматично відправляє особисті повідомлення у чат (наприклад, у Slack або Teams) з питаннями про прогрес роботи, існуючі блокери та інші важливі аспекти. Після цього, штучний інтелект збирає дані про команду, аналізує їх та ідентифікує проблеми, які потребують уваги. Наприкінці, штучний інтелект підготовлює підсумки асинхронного стендапу та надсилає їх керівнику команди або проєктному менеджеру для подальшого аналізу.

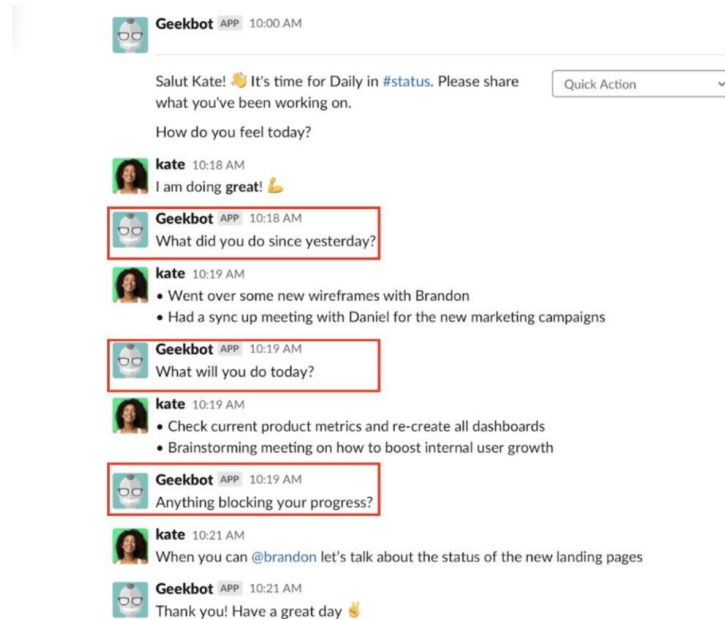


Рис. 3.8. Тестування можливостей

Для оцінки ефективності використання штучного інтелекту в управлінні проектами рекомендується випробувати платформи ClickUp і Monday. Ці системи представляють собою комплексні екосистеми, побудовані на базі технологій штучного інтелекту. За думкою автора, платформа ClickUp вирізняється вищою якістю та більшим спектром функціональності, однак справжня перевага полягає не лише в цьому.

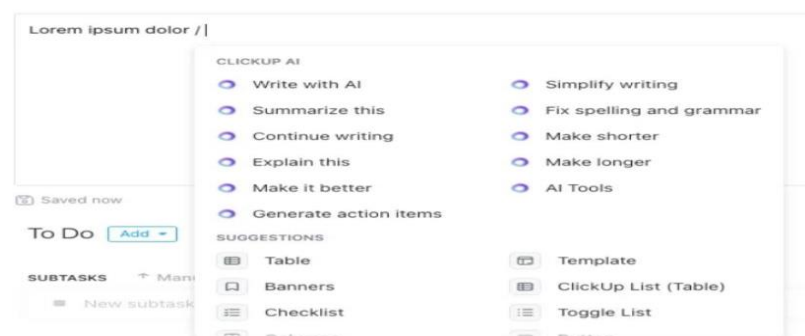


Рис. 3.9. Практична реалізація ШІ

Усі запити в системах штучного інтелекту реєструються та, ймовірно, використовуються для подальшого вдосконалення системи. Коли ви ділитесь на зустрічі інформацією про нові функції вашого продукту, ця інформація стає доступною віртуальному проектному менеджеру, до якого ви добровільно

приєдналися для дзвінка з метою отримання підсумків зустрічі. Однак спосіб, яким штучний інтелект використовуватиме цю інформацію, і чи стане вона відома потенційним клієнтам чи конкурентам, залишається питанням [19, с. 76].

Навіть у штучного інтелекту можуть виникати галюцинації, як і у людей. Ці галюцинації виникають, коли штучний інтелект використовує правильну інформацію, що не пов'язана з темою вашого запиту. Практично всі сервіси штучного інтелекту на ринку можуть демонструвати галюцинації, а деякі навіть можуть вигадувати (або генерувати) нереальні дані для задоволення вимог запиту. Для досягнення відмінних результатів з використанням штучного інтелекту необхідно ретельно готувати не лише вхідні дані, але й ретельно перевіряти результат його роботи.

Майбутнє з урахуванням поточних викликів та розробок у галузі штучного інтелекту виглядає перспективним. Технології і надалі будуть диктувати темпи змін у галузі та підтримуватимуть управління проектами. Важливо, щоб галузь була готова до цих змін і могла проактивно використовувати їх.

Програмний код, що реалізує описані вище моделі та алгоритми, представлений у Додатку Г.

Висновки до розділу 3.

Практична реалізація комп'ютерних моделей дозволяє організаціям оптимізувати процеси планування, контролю, та звітності, що забезпечує кращий контроль над проектами та сприяє досягненню їхніх цілей. Додатково, впровадження таких моделей може допомогти зменшити витрати та ризики, що часто пов'язані з недостатнім управлінням проектами.

Наша дипломна робота підтверджує, що практична реалізація комп'ютерних моделей управління проектами є важливим кроком для підвищення ефективності бізнесу та досягнення стратегічних цілей організацій. Вона дозволяє організаціям пристосовуватися до змінних умов ринку та швидко реагувати на нові виклики та можливості.

У підсумку, практична реалізація комп'ютерних моделей управління проектами виявляється важливою складовою успішного управління проектами у сучасному бізнес-середовищі. Вона створює платформу для ефективного управління ресурсами, забезпечує збалансований підхід до ризиків та дозволяє організаціям досягати своїх стратегічних цілей.

Отже, практична реалізація комп'ютерних моделей управління проектами виявляється не лише корисною, але і необхідною для сучасних організацій, що працюють в умовах постійної зміни та конкурентного середовища. Наша дипломна робота підкреслює важливість впровадження сучасних технологій управління проектами для досягнення успіху та стабільного розвитку підприємств.

ВИСНОВКИ

Таким чином, автоматизація управління проектами - це процес використання різноманітних інформаційних технологій та програмних засобів для автоматизації процесів планування, виконання, контролю та аналізу проектів. Використання автоматизованих комп'ютерних моделей управління проектами спрощує рутинні операції, забезпечує більш точне прогнозування результатів та дозволяє швидше реагувати на зміни в умовах проектної діяльності.

У даній дипломній роботі було ретельно проаналізовано сучасні тенденції та інструменти автоматизованої управління проектами. Дослідження показало, що використання таких моделей може значно полегшити процес управління проектами, знизити ризики затримок та перевищень бюджету, а також підвищити ефективність використання ресурсів. Основними перевагами автоматизованих комп'ютерних моделей управління проектами є їхній високий рівень точності та швидкість обробки інформації.

Застосування таких моделей дозволяє підвищити продуктивність та конкурентоспроможність організацій у сучасному бізнес-середовищі. Однак, наш аналіз також виявив деякі обмеження та виклики, пов'язані з імплементацією та використанням автоматизованих комп'ютерних моделей управління проектами. Зокрема, це може включати високі витрати на впровадження та обслуговування таких систем, необхідність навчання персоналу та вирішення питань щодо захисту конфіденційності даних.

У цілому, дослідження підтвердило важливість використання автоматизованих комп'ютерних моделей управління проектами для підвищення ефективності та успішності реалізації проектів. Рекомендації, які випливають з цього дослідження, можуть бути корисними для організацій, що прагнуть оптимізувати свою проектну діяльність та досягти більшого успіху у своїх проектах.

У першому розділі нашої дипломної роботи було проведено детальний аналіз теоретичних аспектів автоматизації управління проектами. Ми розглянули різні підходи та методики, які використовуються в управлінні проектами, і проаналізували їхню придатність для автоматизації. Особлива увага була приділена інформаційним технологіям та програмним засобам, які використовуються для автоматизації управління проектами. На основі отриманих даних було визначено ключові принципи та стратегії автоматизації управління проектами, які є базою для подальшого розвитку дослідження.

У другому розділі нашої дипломної роботи ми зосередились на розробці узагальненої автоматизованої комп'ютерної моделі управління проектами. Використовуючи отримані теоретичні знання та практичний досвід, ми розробили модель, яка враховує найкращі практики та сучасні технології управління проектами. Наша модель включає в себе інтегровану систему для планування, виконання, моніторингу та звітування про проекти, що дозволяє забезпечити ефективне управління проектами в різних сферах діяльності.

У третьому розділі нашої дипломної роботи ми зосередились на практичній реалізації комп'ютерних моделей управління проектами. Ми вивчили конкретні випадки впровадження таких моделей в різних організаціях та сферах діяльності. На основі аналізу цих випадків ми зробили висновки про ефективність та перспективи застосування комп'ютерних моделей управління проектами в практичній діяльності. Наші дослідження підтвердили, що використання автоматизованих комп'ютерних моделей управління проектами сприяє підвищенню продуктивності та ефективності управління проектами, що відкриває нові можливості для розвитку бізнесу та досягнення стратегічних цілей організації.

Отже, результати дослідження свідчать про важливість та актуальність застосування автоматизованих комп'ютерних моделей управління проектами в сучасному бізнес-середовищі. Автоматизовані моделі дозволяють підвищити ефективність управління проектами, зменшити ризики та оптимізувати використання ресурсів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Абраменко І.Г., Абраменко Д. І. Теорія автоматичного управління. – Харків: ХНАМГ, 2018. – 178 с.
2. Башинська І.О. Управління ризиками в проєктах / І.О. Башинська, Д.О. Макарець // Економіка, фінанси, право. Щомісячний інформаційно-аналітичний журнал. – К.: Аналітик, 2017. – №6 – С. 3-5.
3. Башинська І.О., Новак Н.Г. Ефективне управління проєктами підприємства / Інфраструктура ринку: електронний науково-практичний журнал. – 2017. – №6. – С. 113-117.
4. Бобух А.О.. Автоматизовані системи управління технологічними процесами : Навч. посібник. – Харків: ХНАМГ, 2006. - 185 с
5. Болтак, О. Л. Шляхи вдосконалення системи управління підприємством [Електронний ресурс] / О. Л. Болток // Відкритий міжнародний університет розвитку людини «Україна». - Режим доступу: \www/URL: <http://nauka.zinet.info/9/boltak.php>
6. Василевська А. Управління проєктами підприємства із використання інформаційних технологій / [монографія]. КТНЕУ. 2012. С. 99.
7. Вілфрід Т. Проєктний менеджмент: конспект лекцій и семінарів / Т. Вілфрід. – Тернопіль: Економічна думка. 2001. С. 95.
8. Возна Н.Я. Методологія та інформаційна технологія побудови моделей "блоксхема алгоритму руху даних" в розподілених комп'ютерних системах реального часу. Збірник наукових праць Бучацького інституту менеджменту і аудиту. – Бучач. – 2011 - №7.-С.336-339.
9. Возна Н.Я. Методологія та техніка формування техніко-економічних даних в автоматизованих системах управління // Вісник Хмельницького національного університету. – 2005. – №4. – Т.2, Ч.1. – С. 131-133.
10. Герасимяк Р. П. Теорія автоматичного управління. Збірник задач: навчальний посібник / Р.П. Герасимяк. – О.: Наука і техніка, 2003. – 108 с.

- 11.Гладчук, О. Інноваційна економіка [Текст] / О. Гладчук // Науково-виробничий журнал. - 2013. - № 10(48). - С. 167–174.
- 12.Гоголюк П. Ф. Теорія автоматичного управління: навч. посіб. / П.Ф. Гоголюк, Т.М. Гречин. – Львів: Видавництво Львівської політехніки, 2009. – 280 с.
- 13.Гоголюк П.Ф., Гречин Т.М.С Теорія автоматичного управління: Підручник. – Львів: Видавництво Національного університету "Львівська політехніка", 2018. – 285 с
- 14.Гордієнко І.В. Інформаційні системи і технології в менеджменті. К. КНЕУ, 2003.
- 15.Гужва В.М. Інформаційні системи і технології на підприємствах: Навч. посібник. - К.: КНЕУ, 2001. – 400 с.
- 16.Гушко С.В., Шайкан А.В. Управлінські інформаційні системи. Навчальний посібник. - Львів: "Магнолія Плюс", 2006. - 320 с.
- 17.Демиденко М.А. Управління проектами інформатизації : навч. посіб. / М.А. Демиденко ; Нац. гірн. ун-т. — Електрон. текст. дані. – Д. : 2014. – 114 с.
- 18.Дубовой В. М., Моделювання та оптимізація систем: підручник / В.М. Дубовой, Р.Н. Кветний, О.І. Михальов, А.В.Усов. – Вінниця: «ГД «Еднльвейс», 2017. – 804 с.
- 19.Зайцев Г.Ф., Стеклов В.К., Бріцький О.І. Теорія автоматичного управління. – К., Техніка, 2012. – 688 с.
- 20.Іванов А. О. Теорія автоматичного управління: Підручник. - Дніпропетровськ: Національний гірничий університет. - 2014. - 250 с
- 21.Інтегрування моніторингу і оцінки інвестиційного проекту з енергозбереження в систему контролю діяльності підприємства на засадах контролінгу [Електронний ресурс] / А.А. Балан, С.В. Філіппова // Економіка: реалії часу. Науковий журнал. – 2014. – № 4 (14). – С. 180-185.
- 22.Інформаційні системи і технології в економіці: Посібник. /За ред. В.С. Пономаренка. – К.: ВЦ. “Академія”, 2002.

23. Кім Д.П. Теорія автоматичного управління. Том 2. Багатовимірні, нелінійні, оптимальні й адаптивні системи/ Д.П. Кім. – Фізматліт, 2004. – 32 с.
24. Кобиляцький Л.С. Управління проектами: Навч. посібник. К.: МАУП. 2002. С. 200.
25. Козаченко В.Є. Управління загальною вартістю володіння КІС // Корпоративні системи. № 2 2002. С. 13.
26. Комп'ютерне моделювання процесів та систем. Чисельні методи: підручник / С.П. Вислоух, О.В. Волошко, Г.С. Тимчик, М.В. Філіппова. – Київ: КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2021. – 228 с.
27. Лєвошич О. Л., Крак Ю. В. Елементи теорії управління. Навчально-методичний посібник для студентів факультету кібернетики спеціальності «Інформатика». – К. : Видавничо-поліграфічний центр «Київський університет», 2002. – 85 с.
28. Лукінюк М. В. Автоматизація типових технологічних процесів: технологічні об'єкти управління та схеми автоматизації: навч. посіб. для студ. вищ. навч. закл., які навч. за напр. «Автоматизація і комп'ютерно-інтегровані Технології» / М.В. Лукінюк ; Нац. техн. ун-т України «Київськ. політехн. ін-т». – К. : КПІ, 2008. – 236 с.
29. Морозов В.В. Управління проектами розвитку підприємств: навч. посібник / В.В. Морозов, О.В. Кальніченко, Ю.Г. Турло. – К.: Університет економіки та права «КРОК». 2011. – С. 232.
30. Николайчук Я.М., Возна Н.Я., Пітух І.Р. Проектування спеціалізованих комп'ютерних систем / Навчальний посібник / - Тернопіль: ТзОВ "Тернограф". 2010. – 392с., іл.
31. Николайчук Я.М., Пітух І.Р., Возна Н.Я. Теорія моделей руху даних розподілених комп'ютерних систем / Монографія - Тернопіль: ТзОВ "Тернограф", 2008 – 216 с.

32. Нікулін О.А. Основи теорії автоматичного управління. Частотні методи аналізу та синтезу систем. / О.А. Нікулін. – СПб.: БХВ-Петербург, 2004. – 640 с.
33. Основи побудови комп'ютерно-інтегрованих систем [Електронний ресурс] : навч. посіб. для студ. спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології», освітньо-професійна програма «Автоматизація та комп'ютерно-інтегровані технології кіберенергетичних систем» / Укладачі: С. В. Любицький, П. В. Новіков ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1,5 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2020. – 77 с.
34. Пасічник В.В., Литвин В.В., Шаховська Н.Б. Проектування інформаційних систем. Навчальний посібник (затв. МОН України) Львів: 2013.– 380 с.
35. Побережець О.В. Теоретико-методологічні та практичні засади дослідження системи управління результатами діяльності промислового підприємства: [моногр.] / О.В. Побережець. – Херсон: Видавництво: Грінь Д.С., 2016. – 500 с.
36. Попович М. Г. Теорія автоматичного управління: Підручник / М.Г. Попович, О.В. Ковальчук. – Київ: «Либідь», 2007. – 656 с. - ISBN: 966-06-0447
37. Попович М. Г., Ковальчук О. В. Теорія автоматичного управління: Підручник. – 2-ге вид., перероб. і доп. – К.: Либідь, 2007. – 656 с.
38. Посібник з лекцій із дисципліни «Автоматизовані системи управління технологічними процесами» напрям підготовки 6.050202 «Автоматизація та комп'ютерно-інтегровані технології» / Укладач : Карташов В.В. – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2017 – 148 с.
39. Проць Я.І., Данилюк О.А., Лобур Т.Б. Автоматизація неперервних технологічних процесів: Навчальний посібник для технічних спеціальностей вищих навчальних закладів. Тернопіль: ТДТУ ім. І. Пулюя, 2008. – 239 с.


- 40.Рекомендації щодо розроблення навчальних планів / Уклад. В. П. Головенкін. – К. : Нац. техн. ун-т України «Київ. політех. ін-т», 2012. – 23 с.
- 41.Репнікова Н. Б. Теорія автоматичного управління: класика і сучасність; підручник / Н. Б. Репнікова. – К. : НТУУ «КПІ», 2011. – 328 с.
- 42.Системи автоматичного проектування САПР [Електронний ресурс]. - Режим доступу: \www/URL: <http://joiner.org.ua/2rozrjad/2009-07-08-13-19-32/2009-07-24-08-10-10/2009-07-24-08-42-32.html>. - Загол. з екрану.
- 43.Талюпа, Н. Сучасні підходи до удосконалення технології управління [Текст] / Н. Талюпа // Інвестиції: практика та досвід. - 2009. - № 8. - С. 49–50.
- 44.Тарасюк Г.М. Управління проектами: навч. посібник для студентів Вищих Навчальних Закладів / Г.М. Тарасюк. 2-е вид. К.: Каравела. 2006. С. 320.
- 45.Теорія автоматичного управління: Навчальний посібник./Л.М. Артюшин, Б.В. Дурняк,О.А. Машков, М.С. Сівов. – Львів:Вид-твоУАД, 2004. – 272 с.
- 46.Теорія автоматичного управління: Підручник/За ред. Г.Ф. Зайцева. – К.:Техніка, 2002. – 668 с.
- 47.Трегуб В.Г. Автоматизація технологічних процесів: Курс лекцій для студентів напряму 0925 “Автоматизація та комп’ютерно-інтегровані технології” заочн. форми навчання – К.: НУХТ, 2007. – 42 с.
- 48.Чорноволенко І.Ф. Економіка інформаційних систем. Донецьк: [б.в.], 2002. С. 117-118.
- 49.Шапуров, О. Сутність, роль і об’єктивна необхідність удосконалення управління підприємствами [Текст] / О. Шапуров // Актуальні проблеми економіки. - 2008. - № 8. - С. 138–146.
- 50.Investment projects at industrial enterprises: accounting and implementation control [Електронний ресурс] / O.S. Balan, O.V. Verber // Економіка: реалії часу. Науковий журнал. – 2013. – № 2 (7). – С. 126-134.

ДОДАТКИ

Додаток А

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Харківський національний університет імені В. Н. Каразіна

Факультет комп'ютерних наук
Кафедра теоретичної та прикладної системотехніки
Рівень вищої освіти (освітньо-кваліфікаційний рівень) **бакалавр**
галузь знань: 15 – Автоматизація та приладобудування
спеціальність: 151 – Автоматизація та комп'ютерно-інтегровані технології

ЗАТВЕРДЖУЮ
Завідувач кафедри теоретичної
та прикладної системотехніки
 д.т.н., проф. Шматков С. І.
«21» грудня 2023 року

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

БУТ Богдан Олександрович

1. Тема роботи «**Автоматизована комп'ютерна модель управління проектами**»

керівник роботи Булавін Дмитро Олександрович, доцент, к.т.н.
затверджені наказом по університету від «03» травня 2024 року № 4101-5/909

2. Строк подання студентом роботи 31 травня 2024 року

3. Перелік питань, які потрібно розробити

- 1) Аналіз методів управління проектами та можливості їх автоматизації
- 2) Аналіз комп'ютерних моделей управління проектами: переваги та недоліки
- 3) Постановка задачі створення нової комп'ютерної моделі оптимізації управління проектами?
- 4) Створення комп'ютерної моделі оптимізації управління проектами
- 5) Тестування та можливості впровадження комп'ютерної моделі в компаніях
- 6) Аналіз майбутніх тенденцій та інновацій в комп'ютерних моделях управління проектами

4. План роботи

№ з/п	Назви етапів роботи	Термін виконання етапів роботи
1	Аналіз научної літератури	21.12.2023 - 25.01.2024
2	Аналіз існуючих методологій управління проектами	19.12.2023 - 2.01.2024
3	Проектування комп'ютерної моделі управління проектами	2.01.2024 - 2.02.2024
4	Розробка та тестування комп'ютерної моделі	2.01.2024 - 2.02.2024
5	Створення моделі навчання в неформальному форматі	3.02.2024 - 30.03.2024
6	Розробка та тестування моделі неформального навчання	3.03.2024 - 30.04.2024
7	Впровадження та оцінка ефективності моделі	1.05.2024 - 10.05.2024
8	Розробка пояснювальної записки	15.04.2024 - 27.05.2024

5. Дата видачі завдання 21.12.2023

Студент

Б.О. Бут

ініціали, прізвище



 підпис

Керівник роботи

Д.О. Булавін

ініціали, прізвище



 підпис

Технічне завдання
на розробку програмного виробу
«Автоматизована комп'ютерна модель управління проектами»

Назва розділу	Назва і зміст підрозділу
1. Введення	<p>1.1. Назва проекту: Автоматизована комп'ютерна модель управління проектами</p> <p>1.2. Галузь застосування: Управління проектами, проектне планування</p>
2. Підстава для розробки	<p>2.1. Освітній курс за спеціальністю 151 – Автоматизація та комп'ютерно-інтегровані технології.</p> <p>2.2. Завдання на дипломну роботу бакалавра, затверджено наказом ХНУ імені В. Н. Каразіна № xxxx-xx/xxx від xx.xx.2024 р. (представить як Додаток А до пояснювальної записки до кваліфікаційної роботи).</p>
3. Призначення розробки	<p>3.1. Мета: Розробка інноваційної моделі для ефективного управління проектами.</p> <p>3.2. Призначення: Застосування в корпоративному секторі та управлінні державними проектами.</p> <p>3.3. Початкові дані для розробки: Існуючі методики і підходи управління проектами.</p>
4. Технічні вимоги до програмного виробу	<p>4.1. Функціональні характеристики: Підтримка планування, відстеження та аналізу проектів.</p> <p>4.2. Надійність: Висока надійність та стійкість до помилок.</p> <p>4.3. Умови експлуатації: Працездатність у різних операційних системах та пристроях.</p> <p>4.4. Технічні засоби: Сумісність із стандартним офісним обладнанням.</p> <p>4.5. Сумісність: Інтеграція з іншими програмами управління проектами.</p> <p>4.6. Маркування та упаковка: Ліцензійний ключ та цифрова доставка.</p> <p>4.7. Транспортування та зберігання: Не застосовується, продукт цифровий.</p>

	4.8. Спеціальні вимоги: Відповідність міжнародним стандартам безпеки даних.
5. Вимоги до програмної документації.	Повний набір документів, включаючи користувацьку інструкцію, технічну специфікацію, інструкції з інсталяції та використання.
6. Техніко-економічні показники	Оцінка вартості розробки, експлуатації та потенційного доходу від впровадження.
7. Стадії і етапи розробки	
8. Порядок контролю і приймання	<ol style="list-style-type: none"> 1. Внутрішній контроль: Регулярні перевірки на відповідність технічному завданню. 2. Зовнішній контроль: Незалежне тестування функціоналу. 3. Приймання робіт: Фінальне затвердження системи після демонстрації відповідності всім вимогам.

Виконавець

студент групи КУ-41

Бут Б.О.



підпис

Замовник

к.т.н. доцент кафедри ТПС

Булавін Дмитро Олексійович



підпис

Програма і методика випробувань програмного виробу

«Автоматизована комп'ютерна модель управління проєктами»

1 Об'єкт випробувань

1.1 Назва: Автоматизована комп'ютерна модель управління проєктами

1.2 Область застосування: Корпоративне управління проєктами

2. Мета випробувань

Загальна мета: Оцінка ефективності моделі у реальних умовах.

Специфічні цілі: Перевірка точності планування, відстеження виконання проєктів, і аналізу ефективності ресурсів.

3. Загальні положення

3.1 Підстави для проведення випробувань

Вимоги освітньої програми.

3.2 Місце і тривалість випробувань

Тестування здійснюється в умовах університетської лабораторії протягом місяця.

3.3 Обсяг випробувань

Повне тестування всіх функцій системи.

3.4 Організації, які беруть участь у випробуваннях

Університет, факультет комп'ютерних наук.

4. Вимоги до програми або програмного виробу

4.1. Функціональні характеристики: Підтримка планування, відстеження та аналізу проєктів.

4.2. Надійність: Висока надійність та стійкість до помилок.

4.3. Умови експлуатації: Працездатність у різних операційних системах та пристроях.

4.4. Технічні засоби: Сумісність із стандартним офісним обладнанням.

4.5. Сумісність: Інтеграція з іншими програмами управління проєктами.

4.6. Маркування та упаковка: Ліцензійний ключ та цифрова доставка.

4.7. Транспортування та зберігання: Не застосовується, продукт цифровий.

4.8. Спеціальні вимоги: Відповідність міжнародним стандартам безпеки даних.

5. Вимоги до програмної документації

Повний набір документів, включаючи користувацьку інструкцію, технічну специфікацію, інструкції з інсталяції та використання.

6. Засоби і порядок випробувань

6.1 Засоби випробувань

Комп'ютерні системи та програмне забезпечення для симуляції управління проектами.

6.2 Порядок проведення випробувань

Тест 1: Верифікація функціональності модулю авторизації

- **Мета тесту:** Перевірка ефективності процедур авторизації користувачів.
- **Опис процедури:** Тестування процесу авторизації через різні методи введення даних, включаючи валідні та невалідні сценарії.
- **Використання зображення:** "Діаграма діяльності модулю авторизації" для ілюстрації тестових випадків та потоків авторизації.

Для кращого розуміння використання системи було визначено наступні варіанти використання:

- **Реєстрація** - дозволяє користувачу зареєструватись у системі та отримати доступ до функціоналу.
- **Авторизація** - дозволяє користувачу авторизуватись у системі та продовжити роботу під існуючим ~~акаунтом~~.
- **Створення проекту** - дозволяє створити новий проект та розпочати його роботу.
- **Управління проектом** - надає можливість змінювати інформацію про проект, якщо на це є відповідні права.
- **Видалення проекту** - дозволяє користувачу видалити проект та всю пов'язану інформацію, якщо має відповідні права.
- **Додавання/видалення учасників** - надає можливість додавати або видаляти учасників проекту відповідно до прав доступу.

- Додавання документів до проекту - дозволяє користувачу завантажувати файли та іншу інформацію до проекту.
- Зміна інформації про користувача - надає можливість оновлювати особисті дані користувача.
- Зміна фотографії користувача - дозволяє користувачу змінити своє фото в профілі.
- ~~Деактивація користувача~~ - функція, що дозволяє користувачу видалити свій обліковий запис.
- Створення/зміна/видалення задач на ~~Канбан-дошці~~ - надає можливість управління задачами на дошці завдань.
- Обрання проекту, формування звіту та перегляд статистики проекту - функції, що надаються користувачу для управління проектом та аналізу його статистики.

Для кращого усвідомлення процесу використання системи, була розроблена діаграма діяльності, яка демонструє послідовність дій при авторизації користувача.



Рисунок В.1 - Діаграма діяльності модулю авторизації

Тест 2: Аналіз роботи модулю управління проектами

- **Мета тесту:** Оцінка функціональності модулю управління проектами та його взаємодії з іншими компонентами системи.
- **Опис процедури:** Симуляція створення, зміни, та видалення проектних завдань, а також перегляду статистики та генерації звітів.

- **Використання зображення:** "Діаграма діяльності модулю управління проектами" та "Діаграма діяльності модулю створення звітів" для демонстрації основних дій і взаємодій в рамках модулю.

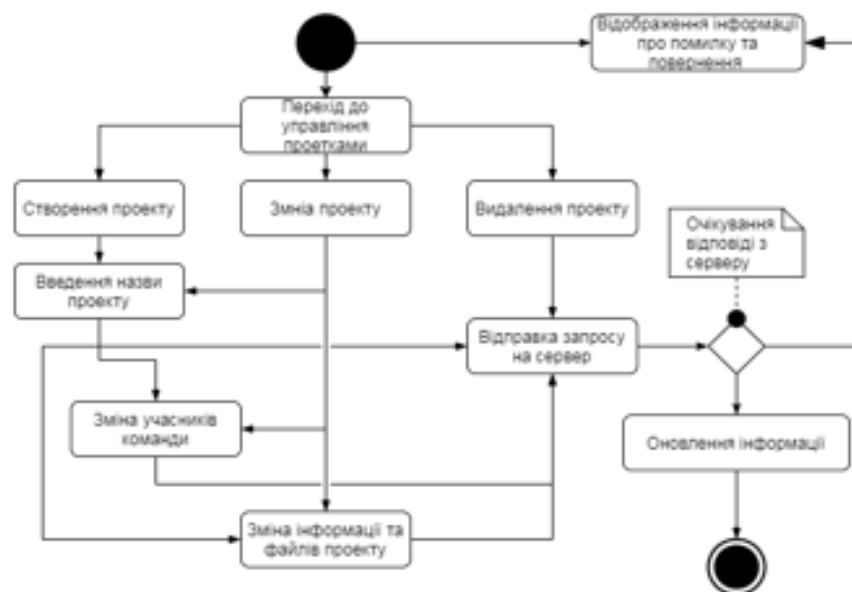


Рисунок В.2 - Діаграма діяльності модулю управління проектами

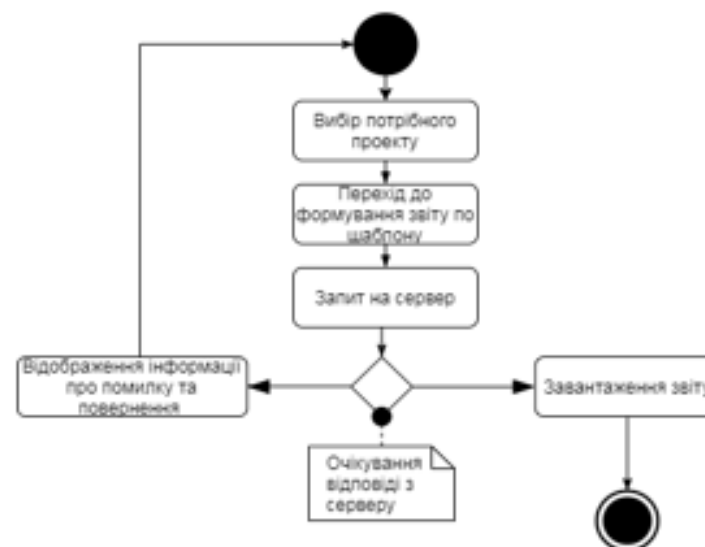


Рисунок В.3 - Діаграма діяльності модулю створення звітів

Тест 3: Перевірка інтеграції з базами даних

- **Мета тесту:** Оцінка здатності системи інтегруватися та коректно взаємодіяти з зовнішніми базами даних.
- **Опис процедури:** Проведення тестів на вставку, оновлення та видалення даних використовуючи інтерфейс DataGrip.
- **Використання зображення:** "Інтерфейс DataGrip" для показу інтерфейсу користувача при взаємодії з базою даних.

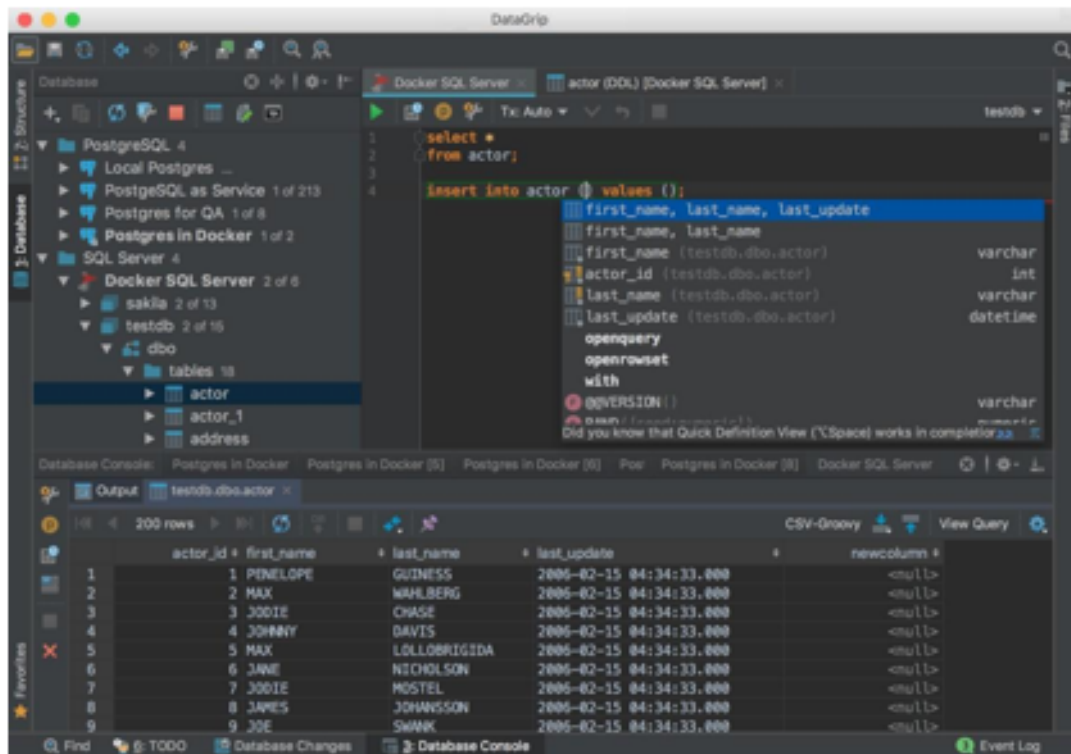


Рисунок В.4 - Інтерфейс DataGrip

Тест 4: Оцінка взаємодії користувача з системою

- **Мета тесту:** Аналіз зручності та інтуїтивності користувацького інтерфейсу, особливо в контексті навігації та доступу до функцій.
- **Опис процедури:** Запрошення групи користувачів для виконання типових задач управління проектами з використанням системи.
- **Використання зображення:** "Діаграма варіантів використання" для ілюстрації основних взаємодій користувачів з системою.

Діаграма варіантів використання є важливим етапом під час аналізу технічного завдання. Вона складається з графічного зображення, що описує акторів та прецеденти, а також специфікації, яка містить текстовий опис конкретних послідовностей дій (потоків подій), які виконує користувач у процесі взаємодії з системою.

На діаграмі варіантів використання виділено сутність для зовнішнього зв'язку у вигляді Бази даних. Зв'язок з нею виникає за необхідності користувача. База даних містить особисту інформацію користувача, дані про проекти та статистичні дані.

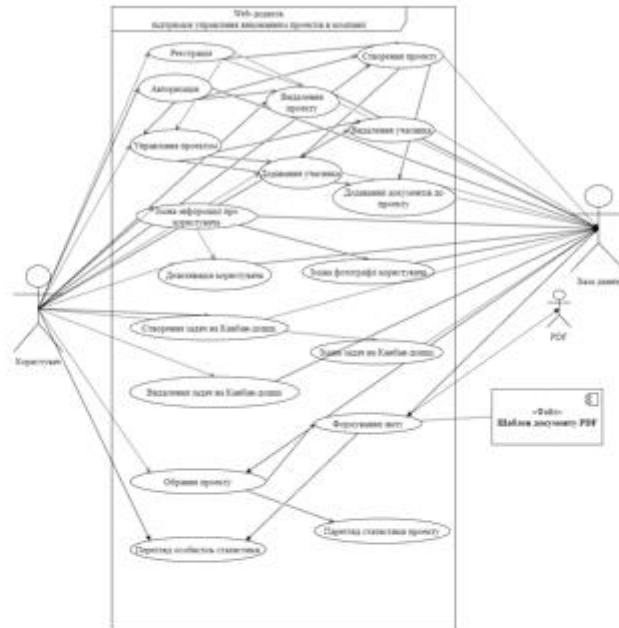


Рисунок В.5 - Діаграма варіантів використання

Висновки: при вдалому виконанні всіх 4 тестів випробування розробленого додатку вважаються успішними.

Виконавець

студент групи КУ-41

Бут Б.О.



Додаток Г

Програмний код для реалізації автоматизованої комп'ютерної моделі управління проектами User.java

```
public class User {  
    private String username;  
    private String role; // "manager" або "team member"  
  
    public User(String username, String role) {  
        this.username = username;  
        this.role = role;  
    }  
  
    public String getUsername() {  
        return username;  
    }  
  
    public String getRole() {  
        return role;  
    }  
  
    @Override  
    public String toString() {  
        return "User{username=\"" + username + "\", role=\"" + role + "\"}";  
    }  
}
```

Task.java

```
public class Task {  
    private String name;  
    private String description;  
    private boolean isCompleted;  
    private User assignedUser;  
  
    public Task(String name, String description) {  
        this.name = name;  
        this.description = description;  
        this.isCompleted = false;  
    }  
  
    public void assignUser(User user) {  
        this.assignedUser = user;  
    }  
}
```

```

public void completeTask() {
    this.isCompleted = true;
}

public String getName() {
    return name;
}

@Override
public String toString() {
    return "Task{name='" + name + "', description='" + description + "',
isCompleted='" + isCompleted + "', assignedUser='" + assignedUser + "'}";
}
}

```

Project.java

```

import java.util.ArrayList;
import java.util.List;

public class Project {
    private String name;
    private List<Task> tasks;
    private List<User> team;

    public Project(String name) {
        this.name = name;
        this.tasks = new ArrayList<>();
        this.team = new ArrayList<>();
    }

    public void addTask(Task task) {
        tasks.add(task);
    }

    public void addUser(User user) {
        team.add(user);
    }

    public List<Task> getTasks() {
        return tasks;
    }
}

```

```

public List<User> getTeam() {
    return team;
}

public String getName() {
    return name;
}

public void generateReport() {
    System.out.println("Project: " + name);
    for (Task task : tasks) {
        System.out.println(task);
    }
}
}
}

```

Communication.java

```

import java.util.ArrayList;
import java.util.List;

public class Communication {
    private List<String> messages;

    public Communication() {
        this.messages = new ArrayList<>();
    }

    public void sendMessage(String message) {
        messages.add(message);
    }

    public void showMessages() {
        for (String message : messages) {
            System.out.println(message);
        }
    }

    public List<String> getMessages() {
        return messages;
    }
}

```

ProjectManagementSystem.java

```
import java.util.ArrayList;
import java.util.List;

public class ProjectManagementSystem {
    private List<Project> projects;
    private List<User> users;
    private Communication communication;
    private Documentation documentation;

    public ProjectManagementSystem() {
        projects = new ArrayList<>();
        users = new ArrayList<>();
        communication = new Communication();
        documentation = new Documentation();
    }

    public void addProject(Project project) {
        projects.add(project);
    }

    public void addUser(User user) {
        users.add(user);
    }

    public List<Project> getProjects() {
        return projects;
    }

    public List<User> getUsers() {
        return users;
    }

    public Communication getCommunication() {
        return communication;
    }

    public Documentation getDocumentation() {
        return documentation;
    }

    public void generateSystemReport() {
        System.out.println("Project Management System Report");
        for (Project project : projects) {
```

```

        project.generateReport();
    }
}

```

CriticalPathMethod.java

```

import java.util.*;

class TaskNode {
    String name;
    int duration;
    List<TaskNode> dependencies;

    TaskNode(String name, int duration) {
        this.name = name;
        this.duration = duration;
        this.dependencies = new ArrayList<>();
    }

    void addDependency(TaskNode task) {
        dependencies.add(task);
    }
}

public class CriticalPathMethod {
    private Map<String, TaskNode> tasks;

    public CriticalPathMethod() {
        tasks = new HashMap<>();
    }

    public void addTask(String name, int duration) {
        tasks.put(name, new TaskNode(name, duration));
    }

    public void addDependency(String taskName, String dependencyName) {
        TaskNode task = tasks.get(taskName);
        TaskNode dependency = tasks.get(dependencyName);
        if (task != null && dependency != null) {
            task.addDependency(dependency);
        }
    }
}

```

```

public void calculateCriticalPath() {
    Map<TaskNode, Integer> earliestCompletionTimes = new HashMap<>();
    int projectDuration = 0;

    for (TaskNode task : tasks.values()) {
        projectDuration = Math.max(projectDuration,
calculateEarliestCompletionTime(task, earliestCompletionTimes));
    }

    System.out.println("Project Duration: " + projectDuration);
}

private int calculateEarliestCompletionTime(TaskNode task, Map<TaskNode,
Integer> earliestCompletionTimes) {
    if (earliestCompletionTimes.containsKey(task)) {
        return earliestCompletionTimes.get(task);
    }

    int earliestCompletionTime = 0;
    for (TaskNode dependency : task.dependencies) {
        earliestCompletionTime = Math.max(earliestCompletionTime,
calculateEarliestCompletionTime(dependency, earliestCompletionTimes));
    }

    earliestCompletionTime += task.duration;
    earliestCompletionTimes.put(task, earliestCompletionTime);
    return earliestCompletionTime;
}

public static void main(String[] args) {
    CriticalPathMethod cpm = new CriticalPathMethod();
    cpm.addTask("A", 3);
    cpm.addTask("B", 2);
    cpm.addTask("C", 5);
    cpm.addTask("D", 4);

    cpm.addDependency("B", "A");
    cpm.addDependency("C", "B");
    cpm.addDependency("D", "B");

    cpm.calculateCriticalPath();
}
}

```

TaskScheduler.java

```

import java.util.PriorityQueue;
import java.util.Queue;

class Task implements Comparable<Task> {
    private String name;
    private int priority; // Lower number means higher priority

    public Task(String name, int priority) {
        this.name = name;
        this.priority = priority;
    }

    public String getName() {
        return name;
    }

    @Override
    public int compareTo(Task other) {
        return Integer.compare(this.priority, other.priority);
    }

    @Override
    public String toString() {
        return "Task{name=\"" + name + "\", priority=\"" + priority + "\"}";
    }
}

public class TaskScheduler {
    private Queue<Task> taskQueue;

    public TaskScheduler() {
        taskQueue = new PriorityQueue<>();
    }

    public void addTask(Task task) {
        taskQueue.add(task);
    }

    public void scheduleTasks() {
        while (!taskQueue.isEmpty()) {
            Task task = taskQueue.poll();
            System.out.println("Executing task: " + task.getName());
        }
    }
}

```

```

}

public static void main(String[] args) {
    TaskScheduler scheduler = new TaskScheduler();
    scheduler.addTask(new Task("Design", 2));
    scheduler.addTask(new Task("Development", 1));
    scheduler.addTask(new Task("Testing", 3));

    scheduler.scheduleTasks();
}
}

```

PerformanceAnalysis.java

```

import org.knowm.xchart.*;
import java.util.Arrays;
import java.util.List;

public class PerformanceAnalysis {

    public void createPerformanceChart(List<String> taskNames, List<Integer>
taskDurations) {
        CategoryChart chart = new
CategoryChartBuilder().width(800).height(600).title("Project
Performance").xAxisTitle("Tasks").yAxisTitle("Duration").build();

        chart.addSeries("Task Durations", taskNames, taskDurations);

        new SwingWrapper<>(chart).displayChart();
    }

    public static void main(String[] args) {
        PerformanceAnalysis analysis = new PerformanceAnalysis();
        List<String> taskNames = Arrays.asList("Design", "Development", "Testing");
        List<Integer> taskDurations = Arrays.asList(5, 10, 7);

        analysis.createPerformanceChart(taskNames, taskDurations);
    }
}

```

MonteCarloSimulation.java

```

import java.io.BufferedReader;
import java.io.InputStreamReader;

```

```
import java.net.HttpURLConnection;
import java.net.URL;

public class APIIntegration {

    private static final String API_URL = "https://api.example.com/data";

    public String fetchData() {
        StringBuilder result = new StringBuilder();
        try {
            URL url = new URL(API_URL);
            HttpURLConnection conn = (HttpURLConnection) url.openConnection();
            conn.setRequestMethod("GET");
            BufferedReader rd = new BufferedReader(new
InputStreamReader(conn.getInputStream()));
            String line;
            while ((line = rd.readLine()) != null) {
                result.append(line);
            }
            rd.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
        return result.toString();
    }

    public static void main(String[] args) {
        APIIntegration apiIntegration = new APIIntegration();
        String data = apiIntegration.fetchData();
        System.out.println("Fetched Data: " + data);
    }
}
```