

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра комп'ютерних наук

Пояснювальна записка
до магістерської кваліфікаційної роботи
**на тему: «Інформаційна технологія прийняття рішень в умовах
критеріальної невизначеності»**

Виконав: студент 2 курсу,
групи 2КН-19м
спеціальності 122 «Комп'ютерні науки»
Мартишев В.О.
Керівник: к.т.н., доц. кафедри КН
Колодний В.В.,
Рецензент: к.т.н., доц. кафедри ПЗ
Романюк О.В.

Вінниця 2020

ЗАТВЕРДЖУЮ
Директор ТОВ «АСТА.МОБІ»
Стахов А.О.
(наук. ст., вч. зв., ініц. та прізви.)

(підпис)
" ____ " _____ 2020 р.

ЗАТВЕРДЖУЮ
Завідувач кафедри КН
д. т. н., проф. Яровий А.А.
(наук. ст., вч. зв., ініц. та прізви.)

(підпис)
" ____ " _____ 2020 р.

ЗАВДАННЯ

на магістерську кваліфікаційну роботу на здобуття кваліфікації магістра зі спеціальності: 122 – «Комп'ютерні науки»
(шифр – назва спеціальності)

08-22.МКР.016.19.000.ПЗ

Магістранта групи 2КН-19м Мартишева Владислава Олеговича
(назва групи) (прізвище, ім'я і по батькові)

Тема магістерської кваліфікаційної роботи: «Інформаційна технологія прийняття рішень в умовах критеріальної невизначеності»

Вхідні дані: набір альтернатив кількістю а не – менше 2, кількість експертів – не менше 2, кросплатформна реалізація, покращення ефективності прийняття рішень – не менше 1%.

Короткий зміст частин магістерської кваліфікаційної роботи:

1. Графічна: схема алгоритму роботи програмного забезпечення прийняття рішень в умовах критеріальної невизначеності, структура інформаційної технології прийняття рішень в умовах критеріальної невизначеності, головне вікно інформаційної технології прийняття рішень в умовах критеріальної невизначеності.

2. Текстова (пояснювальна записка): вступ, аналіз предметної області прийняття рішень, методи оцінювання альтернатив при прийнятті рішень, програмна реалізація інформаційної технології прийняття рішень в умовах критеріальної невизначеності, економічна частина, висновки, перелік використаних джерел, додатки.

КАЛЕНДАРНИЙ ПЛАН ВИКОНАННЯ МКР

№ етапу	Назва етапу	Термін виконання		Очікувані результати
		початок	кінець	
1	Аналіз сучасного рівня розвитку інформаційних технологій прийняття рішень. Постановка задач дослідження			Аналітичний огляд літературних джерел, задачі дослідження, розділ 1 ПЗ
2	Розробка методу комплексної оцінки альтернатив при прийнятті рішень в умовах критеріальної невизначеності			Метод комплексної оцінки альтернатив, розділ 2 ПЗ
3	Практичне застосування та оцінка ефективності розробленого методу			Розділ 3 ПЗ
4	Підготовка економічної частини			Розділ 4 ПЗ
5	Апробація та/або впровадження результатів дослідження			Тези доповідей/акт впровадження
6	Оформлення пояснювальної записки, графічного матеріалу та презентації			Пояснювальна записка, графічний матеріал, презентація

Консультанти з окремих розділів магістерської кваліфікаційної роботи:

1. Науковий керівник _____ доц. кафедри КН
(підпис) (науковий ступінь, вчене звання)
« ____ » _____ 2020 р. В.В. Колодний
(ініціали та прізвище)

2. Економічна частина _____ канд. екон.наук, доц.каф
(підпис) (науковий ступінь, вчене звання)
« ____ » _____ 2020 р. М. В. Бальзан
(ініціали та прізвище)
Дата попереднього захисту роботи “ ____ ” _____ 2020 р.

Рецензент _____ к.т.н., доц. кафедри ПЗ
(підпис) (науковий ступінь, вчене звання)
О.В. Романюк
(ініціали та прізвище)

Завдання видав керівник роботи _____ к.т.н, доц. кафедри КН,
(підпис) (науковий ступінь, вчене звання)
« ____ » _____ 2020 р. В.В. Колодний
(ініціали та прізвище)

Завдання отримав магістрант _____ В.О. Мартишев
(підпис) (ініціали та прізвище)
« ____ » _____ 2020 р.

АНОТАЦІЯ

В роботі пропонується інформаційна технологія прийняття рішень в умовах критеріальної невизначеності. Розглянуто предметну область прийняття рішень. Визначено способи прийняття рішень з використанням методів експертного оцінювання альтернатив. Розроблено структуру та алгоритм роботи інформаційної технології прийняття рішень в умовах критеріальної невизначеності. Інформаційна технологія програмно реалізована на мові JavaScript для виконання у веб-браузері. Розроблена програма упорядковує альтернативи за їх значущістю за рахунок використання комбінації методів експертного оцінювання.

ABSTRACT

The paper proposes information technology for decision making in conditions of criterion uncertainty. The subject area of decision making is considered. Methods of decision - making using methods of expert evaluation of alternatives are defined. The structure and algorithm of information technology decision making in the conditions of criterion uncertainty are developed. Information technology is software implemented in JavaScript for execution in a web browser. The developed program organizes alternatives according to their significance through the use of a combination of expert evaluation methods.

ЗМІСТ

ВСТУП.....	7
1 ОБҐРУНТУВАННЯ ДОЦІЛЬНОСТІ РОЗРОБКИ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРИЙНЯТТЯ РІШЕНЬ В УМОВАХ КРИТЕРІАЛЬНОЇ НЕВИЗНАЧЕНОСТІ	10
1.1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ПРИЙНЯТТЯ РІШЕНЬ	10
1.2 КЛАСИФІКАЦІЯ МЕТОДІВ ПРИЙНЯТТЯ РІШЕНЬ	12
1.3 АНАЛІЗ ОБ’ЄКТУ ПРОЕКТУВАННЯ.....	14
1.4 ХАРАКТЕРИСТИКА ТА АНАЛІЗ АНАЛОГІВ	15
1.5 ВИСНОВОК	18
2 МЕТОДИ ОЦІНЮВАННЯ АЛЬТЕРНАТИВ ПРИ ПРИЙНЯТТІ РІШЕНЬ.....	19
2.1 АНАЛІЗ ОСНОВНИХ МЕТОДІВ ОЦІНКИ АЛЬТЕРНАТИВ	19
2.2 МОДЕЛІ ЕКСПЕРТНИХ ОЦІНОК В УМОВАХ КРИТЕРІАЛЬНОЇ НЕВИЗНАЧЕНОСТІ	22
2.3 РОЗРОБКА МЕТОДУ КОМПЛЕКСНОЇ ОЦІНКИ АЛЬТЕРНАТИВ В УМОВАХ КРИТЕРІАЛЬНОЇ НЕВИЗНАЧЕНОСТІ.....	30
2.4 АНАЛІЗ МЕТОДІВ ОЦІНКИ КОМПЕТЕНТНОСТІ ЕКСПЕРТІВ.....	32
2.5 ВИСНОВОК	37
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРИЙНЯТТЯ РІШЕНЬ В УМОВАХ КРИТЕРІАЛЬНОЇ НЕВИЗНАЧЕНОСТІ.....	38
3.1 ОБҐРУНТУВАННЯ ВИБОРУ ПРОГРАМНО-АПАРATНОЇ ПЛАТФОРМИ.....	38
3.2 ОБҐРУНТУВАННЯ ВИБОРУ МОВИ ПРОГРАМУВАННЯ.....	48
3.3 ОБҐРУНТУВАННЯ ВИБОРУ ФРЕЙМВОРКУ ДЛЯ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ.....	52
3.4 РОЗРОБКА СТРУКТУРИ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ.....	56
3.5 РОЗРОБКА АЛГОРИТМУ РОБОТИ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ	58
3.6 АНАЛІЗ РЕЗУЛЬТАТІВ ТЕСТУВАННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ	61
3.7 ВИСНОВОК	63
4 ЕКОНОМІЧНА ЧАСТИНА	64
4.1 ОЦІНЮВАННЯ КОМЕРЦІЙНОГО ПОТЕНЦІАЛУ РОЗРОБКИ	64

4.2 ПРОГНОЗУВАННЯ ВИТРАТ НА ВИКОНАННЯ НАУКОВОЇ РОБОТИ ТА ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ.....	70
4.3 ПРОГНОЗУВАННЯ КОМЕРЦІЙНИХ ЕФЕКТІВ ВІД РЕАЛІЗАЦІЇ РЕЗУЛЬТАТІВ РОЗРОБКИ.....	75
4.4 РОЗРАХУНОК ЕФЕКТИВНОСТІ ВКЛАДЕНИХ ІНВЕСТИЦІЙ ТА ПЕРІОДУ ЇХ ОКУПНОСТІ.....	77
4.5 ВИСНОВОК	81
ВИСНОВКИ	82
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	84
ДОДАТОК А. АКТ ВПРОВАДЖЕННЯ.....	87
ДОДАТОК Б. ІНСТРУКЦІЯ КОРИСТУВАЧА	88
ДОДАТОК В. ЛІСТИНГ ПРОГРАМИ	92
Додаток Г. Графічна частина	107

ВСТУП

Актуальність теми дослідження. Актуальність створення програмного продукту для прийняття рішень в умовах критеріальної невизначеності є досить високо-пріоритетною на даний час. Кожна людина щодня стикається з необхідністю прийняття рішень. Їх так багато і приймають їх так часто, що в більшості випадків це просто не усвідомлюється. Різні сфери професійної діяльності людини пов'язані з прийняттям рішень, які зводяться до вибору оптимального варіанту поведінки з множини альтернатив.

Досить часто доводиться зустрічатися з проблемою відсутності програмних засобів для прийняття рішень в умовах критеріальної невизначеності. Тому, доцільним є розробка інформаційної технології для прийняття рішень в умовах критеріальної невизначеності.

Для розв'язання поставленої задачі використання інформаційних технологій є актуальним через те, що надає достатні можливості для редагування та налаштування різних методів під дану задачу, та як наслідок, дає змогу отримувати кращий результат у порівнянні з наявними традиційними методами.

Отже, можна стверджувати, що використання сучасних методів прийняття рішень надає змогу подальшого впровадження інформаційних технологій у систему прийняття рішень, а також є актуальною темою дослідження нових методів прийняття рішень.

Зв'язок роботи з науковими програмами, планами, темами. Магістерська робота виконана відповідно до напрямку наукових досліджень кафедри комп'ютерних наук Вінницького національного технічного університету 22 К1 «Моделі, методи, технології та пристрої інтелектуальних інформаційних систем управління, економіки, навчання та комунікацій» та плану наукової та навчально-методичної роботи кафедри.

Мета та завдання дослідження. Метою магістерської кваліфікаційної роботи є підвищення ефективності прийняття рішень в умовах критеріальної

невизначеності за допомогою використання комбінації методів експертного оцінювання та методів оцінки компетентності експертів.

Для досягнення поставленої мети необхідно розв'язати такі задачі:

- виконати огляд розв'язків відомих задач прийняття рішень в умовах критеріальної невизначеності;
- проаналізувати існуючі сервіси та методи вирішення задачі прийняття рішень в умовах критеріальної невизначеності, обґрунтувати вибір обраних методів.
- покращити метод експертного оцінювання альтернатив.
- виконати програмну реалізацію запропонованої інформаційної технології прийняття рішень в умовах критеріальної невизначеності;
- провести тестування програмного продукту та аналіз отриманих результатів.

Об'єкт дослідження – процес прийняття рішень в умовах критеріальної невизначеності.

Предмет дослідження – інформаційні технології прийняття рішень в умовах критеріальної невизначеності з використанням методів експертної оцінки.

Методи дослідження. У роботі використані наступні методи наукових досліджень: системного аналізу для аналізу структури інформаційної системи, теорії методів експертного оцінювання для інформаційної технології прийняття рішень в умовах критеріальної невизначеності, методи математичної статистики для розробки процесу обрахунків результатів експериментів над програмним засобом та об'єктно-орієнтованого програмування для автоматизації розрахунків.

Наукова новизна одержаних результатів полягає в наступному:

- вдосконалено метод експертної оцінки в умовах критеріальної невизначеності, що відрізняється від існуючих застосуванням візуального методу порівняння альтернатив, що забезпечило підвищення ефективності прийняття рішень в умовах критеріальної невизначеності;

– знайшла подальшого розвитку інформаційна технологія прийняття рішень в умовах критеріальної невизначеності, яка орієнтована на WEB технології, що забезпечило підвищення мобільності та зручності використання програмного засобу.

Практичне значення одержаних результатів полягає у тому, що розроблено новий алгоритм роботи системи порівняння альтернатив за рахунок чого досягається підвищення ефективності прийняття рішень і відповідності результатів очікуванням та розроблено програмний засіб для прийняття рішень в умовах критеріальної невизначеності. Результати даного дослідження впроваджено у ТОВ «АСТА.МОБІ» у місті Вінниця.

Достовірність теоретичних положень магістерської кваліфікаційної роботи підтверджується коректністю постановки завдання, коректністю використання методів дослідження, експериментальними дослідженнями тестування програмної реалізації інформаційної технології. Адекватність розроблених методів підтверджується результатами експериментальних досліджень.

Особистий внесок здобувача. Усі результати, наведені у магістерській кваліфікаційній роботі, отримані самостійно. Магістрантом запропоновано використання методу візуального визначення альтернатив, для підвищення швидкості прийняття рішень. Автором було створено програмний додаток, який реалізує створену методику прийняття рішень.

Апробація результатів роботи та публікації. Дані дослідження пройшли апробацію на XLIX науково-технічній конференції професорсько-викладацького складу, співробітників та студентів університету [1], а також отримано свідоцтво про реєстрацію авторського права на твір [2].

1 ОБҐРУНТУВАННЯ ДОЦІЛЬНОСТІ РОЗРОБКИ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРИЙНЯТТЯ РІШЕНЬ В УМОВАХ КРИТЕРІАЛЬНОЇ НЕВИЗНАЧЕНОСТІ

1.1 Аналіз предметної області прийняття рішень

Прийняття рішень (ПР) – це дія над множиною альтернатив, в результаті якої виходить підмножина обраних альтернатив. Найчастіше як найкращий варіант вибирається одна альтернатива [3].

Завдання прийняття рішень (ЗПР) спрямоване на визначення найкращого (оптимального) способу дій для досягнення поставлених цілей. Під метою розуміється ідеальне уявлення бажаного стану або результату діяльності. Якщо фактичний стан не відповідає бажаному, то має місце проблема. Вироблення плану дій з усунення проблеми становить сутність задачі прийняття рішень.

Проблеми можуть виникати в наступних випадках:

- функціонування системи в даний момент не забезпечує досягнення поставлених цілей;
- функціонування системи в майбутньому не забезпечить досягнення поставлених цілей;
- необхідна зміна цілей діяльності.

Проблема завжди пов'язана з певними умовами, які узагальнено називають ситуацією. Сукупність проблеми і ситуації утворює проблемну ситуацію. Виявлення та опис проблемної ситуації дає вихідну інформацію для постановки задачі прийняття рішень [4].

Суб'єктом всякого рішення є особа, яка приймає рішення (ОПР). Поняття ОПР є збірним. Це може бути одна особа - індивідуальна ОПР або група осіб, які б виробляли колективне рішення, групова ОПР. Для допомоги ОПР в зборі та аналізі інформації і формуванні рішень залучаються експерти - фахівці з розв'язуваної проблеми. Поняття експерта в теорії прийняття рішень трактується

в широкому сенсі і включає співробітників апарату управління, що підготовляють рішення, вчених і практиків.

Прийняття рішень відбувається в часі, тому вводиться поняття процесу прийняття рішень. Цей процес складається з послідовності етапів і процедур і спрямованих на усунення проблемної ситуації.

У процесі прийняття рішень формуються альтернативні (Взаємовиключні) варіанти рішень і оцінюється їх перевага. Перевага - це інтегральна оцінка якості рішень, заснована на об'єктивному аналізі (знанні, досвіді, проведенні експериментів і розрахунків) і суб'єктивному розумінні цінності, ефективності рішень.

Для здійснення вибору найкращого рішення індивідуальна ОПР визначає критерій вибору. Групова ОПР робить вибір на основі принципу узгодження

Кінцевим результатом ЗПР є рішення, яке представляє собою припис до дії. Зі змістовної точки зору рішенням може бути спосіб дії, план роботи, варіант проекту тощо рішення є одним з видів розумової діяльності і проявом волі людини і має свої характерні ознаки [3].

Рішення називається допустимим, якщо воно задовольняє обмеження: ресурсні, правові, морально-етичні. рішення називається оптимальним (найкращим), якщо воно забезпечує екстремум (максимум чи мінімум) критерію вибору при індивідуальній ОПР або задовольняє принципом узгодження при груповій ОПР.

Узагальненою характеристикою рішення є його ефективність, ця характеристика містить ефект рішення, що визначає ступінь досягнення цілей, і вартість рішення - сукупність витрат ресурсів для прийняття і реалізації рішення. Таким чином, ефективність вирішення - це ступінь досягнення цілей, віднесена до витрат на їх досягнення. Рішення тим ефективніше, чим більше ступінь досягнення цілей і менша вартість витрат.

Після вибору особи яка приймає рішення, потрібно пройти декілька етапів:

- 1) Аналіз проблемної ситуації
- 2) Визначення цілей та пріоритетів

- 3) Отримання необхідної інформації про проблему
- 4) Формування множини можливих варіантів вирішення проблеми та їх ризику
- 5) Формування критеріїв оцінки рішень та вибір алгоритму прийняття рішення
- 6) Порівняння альтернатив за визначеними критеріями
- 7) Вибір найліпшої альтернативи

Для вирішення проблеми потрібно проаналізувати ситуацію та визначити цілі, які потрібно досягнути вибраним рішенням. Відсутність певних відомостей може призвести до неправильного рішення. Важливим є виявлення усіх можливих рішень для конкретної проблемної ситуації. Після цього ОПР, залежно від поставлених цілей, визначає за якими критеріями буде визначатися, їх важливість та пріоритет альтернатив між собою [5].

Різна експертна думка по-різному оцінює важливість різних обставин проблемної ситуації. Після визначення пріоритетів альтернатив – обирається найкраща. Важливо розуміти, що теорія прийняття рішень допомагає прийняти найкраще рішення саме в певних умовах, зазвичай невизначених. У житті неможливо мати повні відомості щодо проблеми, також не завжди відомі усі можливі варіанти рішення та чіткий та однозначний пріоритет критеріїв, не кажучи про те, що не завжди напевно відомий результат того чи іншого рішення. Тому велика відповідальність лежить саме на ОПР при необхідності їх визначення

1.2 Класифікація методів прийняття рішень

Методи прийняття рішень створені аби обирати найкраще рішення відповідно своєї категорії задач. Задачі поділяються на такі, де кожна альтернатива: має один однозначний результат - визначені умови, має декілька варіантів результату та не має їх вірогідності – невизначені умови, має декілька варіантів результату та їх вірогідність – в умовах ризику. Також кожен тип задач

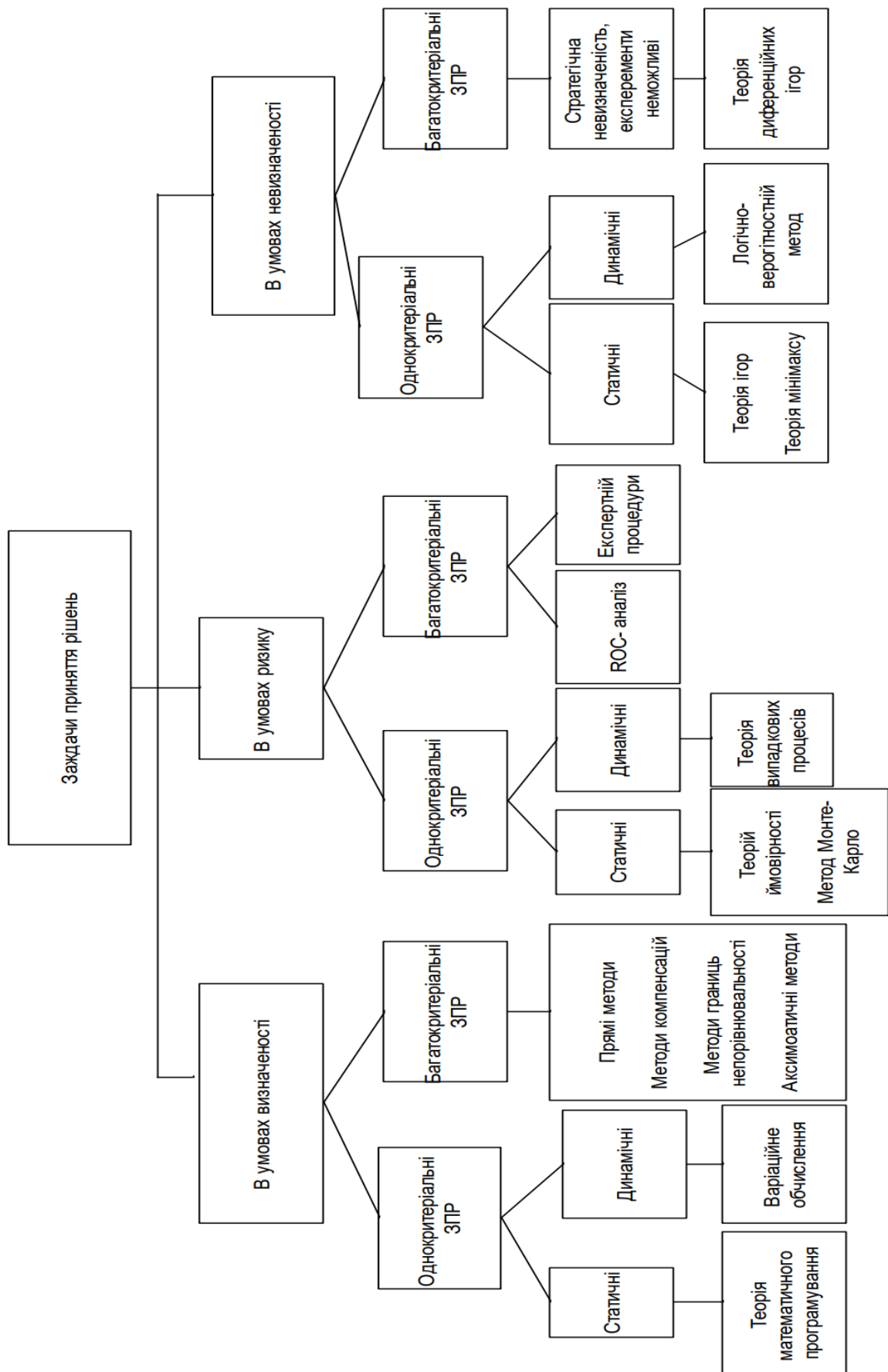


Рисунок 1.1 - Класифікація задач з прийняття рішень

поділяється на однокритеріальні та багатокритеріальні і для кожного з них є свої методи прийняття рішень [6].

Після аналізу задачі слід обрати метод прийняття рішень, залежно від типу задачі, визначити усі можливі варіанти рішення та головні критерії їх оцінок. Зазвичай кожна ситуація має низку можливих рішень, але не усі серед них мають усі мають братися до уваги при застосуванні методу прийняття рішень. Зазвичай, залежно від умови задачі, серед усіх можливих рішень задачі є так звані аутсайдери - альтернативи, що не є кращими за інші по жодному з критеріїв. В залежності від умов задачі та її типу, альтернативи-аутсайдери можуть бути різними. Відсіявши такі можна застосовувати метод прийняття рішень. Різні методи по різному розраховують вплив значення критеріїв на пріоритет альтернативи, тому визначення найліпшого методу для прийняття рішення залежить від ОПР та поставленої цілі.

1.3 Аналіз об'єкту проектування

В результаті аналізу предметної області було доведено актуальність створення інформаційної технології для прийняття рішень в умовах критеріальної невизначеності Для розробки інформаційної технології прийняття рішень в умовах критеріальної невизначеності потрібно виконати наступні задачі:

- виявлення загальних підходів до прийняття рішень;
- вибір та обґрунтування методів оцінки альтернатив;
- розробка структури інформаційної технології;
- розробка алгоритму роботи інформаційної технології;
- програмна реалізація інформаційної технології;
- аналіз результатів тестування програми.

Для роботи інформаційної технології необхідно наступне програмно – апаратне забезпечення, яке задовільняє параметри:

- персональний комп'ютер з апаратним забезпеченням (мінімальні параметрами):
- процесор з частотою 1 GHz або вище;
- не менше 4 ГБ ОЗП;
- не менше 1 ГБ HDD.
- веб-браузер

Інформаційна технологія повинна працювати на сервері на базі Linux.

Результат інформаційної технології повинен містити точний список альтернатив, відсортований в порядку їх переваги.

1.4 Характеристика та аналіз аналогів

«MindDecider» - програма для багатокритеріального аналізу і обробки інформації, наочно представлена у вигляді дерева об'єктів з різними властивостями і поведінкою, залежними від часу і/або інших об'єктів. Розробники позиціонують MindDecider, в якій спрощений інтерфейс поєднується з доступом до досить складного математичного апарату, як інструмент для аналітичної роботи, орієнтований на неспеціаліста.

MindDecider призначений для використання як в наукових дослідженнях, так і для прийняття рішень при управлінні проектами, окремими завданнями і управління часом. У своїх проектах користувач може використовувати вже підготовлені об'єкти з їх властивостями з різних предметних областей (хімічні елементи, фізичні постійні, математичні константи, акції фондових бірж та ін.) Або ж створювати свої [7].

У MindDecider є можливість кожен проект для зручності сприйняття представити у вигляді так званих «карт пам'яті», хронології, електронних таблиць або ж текстового документа.

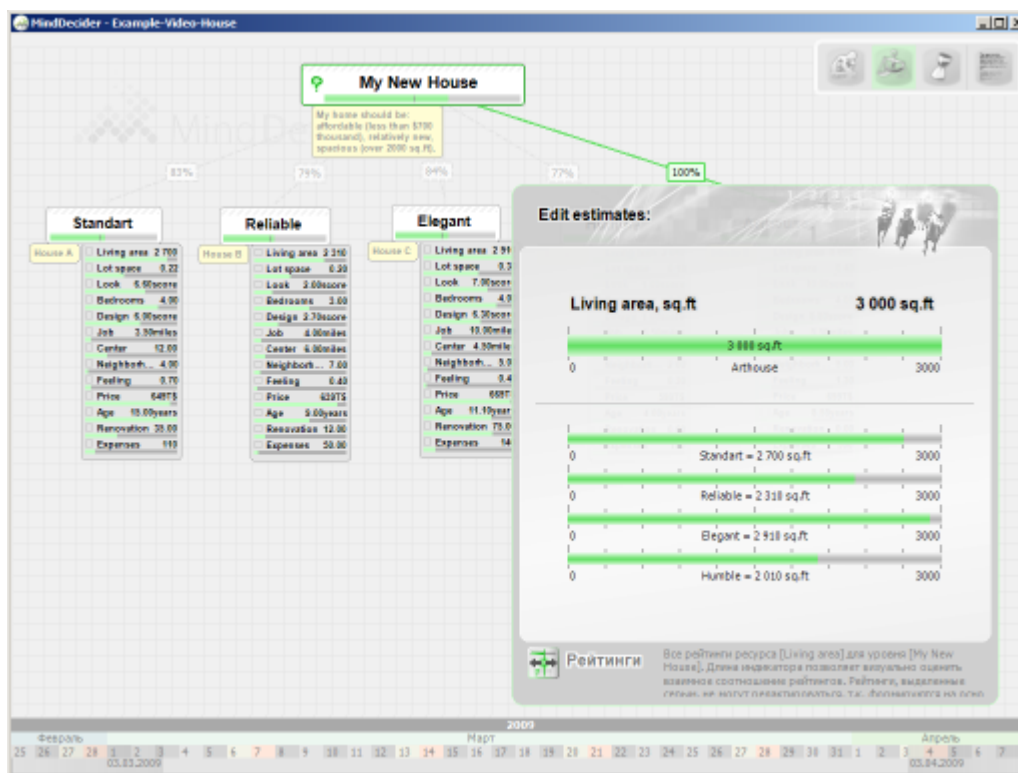


Рисунок 1.1 – Інтерфейс програми MindDecider

PriEsT є аббревіатурою «Інструмент оцінки пріоритетів (англ. Priority Estimation Tool)», який є програмним забезпеченням для прийняття рішень з відкритим вихідним кодом, який реалізує метод Analytic Hierarchy Process (AHP) – всебічний фреймворк для вирішення проблем. PriEsT [8] може допомогти ОПР, у визначенні пріоритетів варіантів, наявних у даному сценарії.

PriEsT реалізує процес аналітичної ієрархії (AHP), який широко використовувався в багатьох галузях, таких як охорона здоров'я, транспорт, телекомунікації та прийняття політичних рішень. Два типи проблем, що розглядаються PriEsT, – це проблеми ранжування та бюджетування [8].

Завдяки відкритому коду, цей інструмент також підходить для дослідницької спільноти. Наприклад, комерційні інструменти не пропонують всіх доступних методів ранжування, і зазвичай приховують технічні деталі реалізації своїх продуктів. PriEsT навпаки має відкритий код, а також дозволяє вносити зміни.

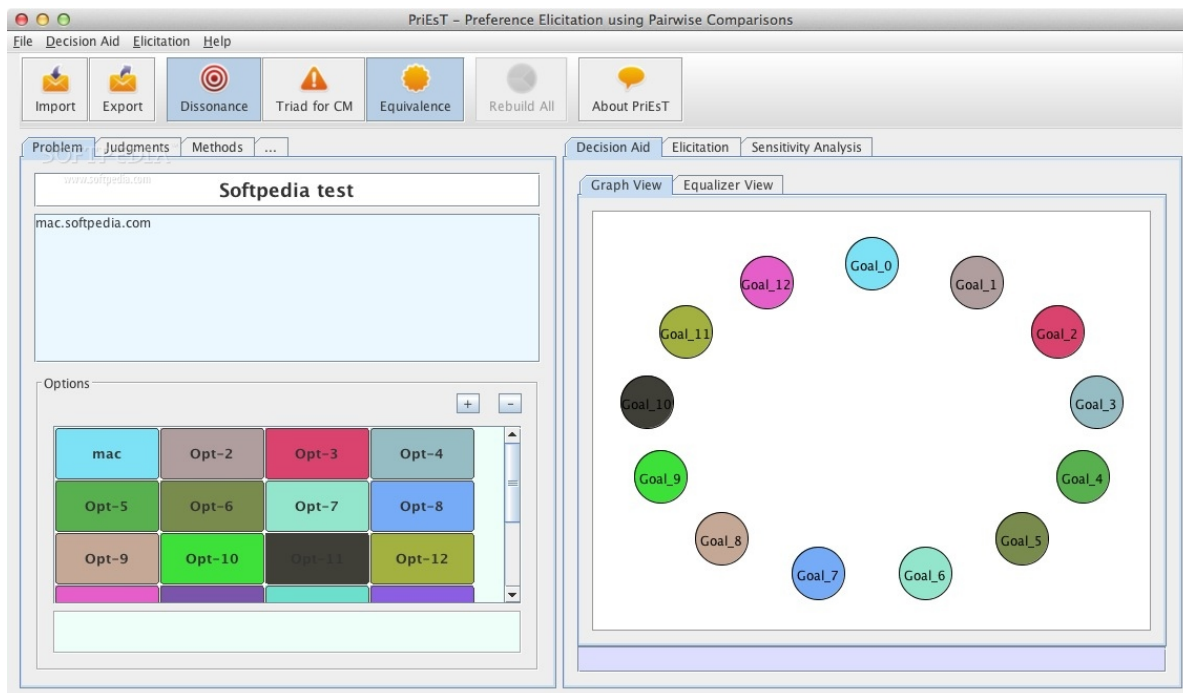


Рисунок 1.2 – Інтерфейс програми PriEsT

Отже, розглянуті програмні реалізації мають ряд недоліків що робить їх недостатньо точними, оскільки вони не здатні враховувати альтернативи, для яких важко виділити критерії для порівняння. З огляду на це, постає питання в розробці нового програмного засобу, з використанням методів оцінювання альтернатив в умовах критеріальної невизначеності.

У таблиці 1.1 наведено аналіз програм для порівняння альтернатив

Таблиця 1.1 – Аналіз програм для порівняння альтернатив

Можливість	MindDecider	PriEsT
Легкість освоєння	+	+
Об'єктивність отриманих результатів	+/-	+/-
Автоматизованість отримання рішень	-	-
Врахування альтернатив, для яких важко. Або не можливо виділити критерії	-	-
Підтримка групового прийняття рішень	-	-

1.5 Висновок

Під час роботи над розділом проаналізовано предметну область, розглянуто основні поняття теорії прийняття рішень, а також базові принципи та процеси, що використовуються при прийнятті рішень. У ході аналізу об'єкту проектування визначено: основні вимоги до інформаційної технології, вхідні та вихідні дані для програмних засобів та вимоги до програмно-апаратного забезпечення системи.

У результаті аналізу систем аналогів: MindDecider та PriEsT, проаналізовано їхні основні можливості та визначено їх переваги та недоліки.

Отже, у даному розділі магістерської кваліфікаційної роботи визначено область використання методів прийняття рішень та основні проблеми, що виникають під час прийняття рішень та засоби за допомогою яких їх можна усунути та вимоги до інформаційної технології прийняття рішень в умовах критеріальної невизначеності.

2 МЕТОДИ ОЦІНЮВАННЯ АЛЬТЕРНАТИВ ПРИ ПРИЙНЯТТІ РІШЕНЬ

2.1 Аналіз основних методів оцінки альтернатив

Бази знань інтелектуальних інформаційних систем, заснованих на правилах, у багатьох реалізаціях використовують дані, джерелами яких є фахівці - експерти. Отримання знань від експертів, обробка цих знань і внесення їх в сховище системи - відповідальна і важка робота. Від того, наскільки ефективно буде зроблена ця робота, багато в чому залежить якість (а іноді і можливість) рішення системою завдань пошуку, аналізу і синтезу інформації.

Сукупність прийомів збору та обробки знань і даних, які готові надати експерти, отримала назву методу експертних оцінок. Думки експертів, їх знання і судження щодо процесів та явищ, які чинять спротив інструментальному виміру, часом є єдиною інформацією, яка може становити зміст декларативною і процедурної компонент БЗ системи по конкретній проблемі або завданню [9].

Знання, отримані від експертів, іноді мають характер досить вільних висловлювань, міркувань або навіть графічних документів. У кожному конкретному випадку вони можуть мати самий різний вигляд. Дуже часто знання експертів надходять в систему в такій формі, що їх використання засобами машинної обробки неможливо без копіткої і важкої обробки. Але в якому б вигляді не поступали в систему знання експертів, вони представляють собою інтелектуальний ресурс, від якості попередньої обробки якого залежить повнота їх використання при вирішенні конкретних завдань.

Методом експертних оцінок вирішується одна з основних проблем ситуації вибору - формування впорядкованої послідовності альтернатив (на основі висловленої експертами системи переваг) для подальшої її обробки з метою отримання функції корисності. Ця функція дозволяє звести проблему багатовимірного аналізу «якості» альтернатив, яку представляють сукупністю показників, до досить простого завдання порівняння за критерієм, вираженого числовим значенням [10].

Працюють з експертами і одержуваними від них знаннями, в основному, аналітики ІВС. На них лежить відповідальність за організацію і проведення збору, обробки та введення експертних даних і знань в систему і їх актуалізацію (підтримка в актуальному стані). Цей вид діяльності носить назву експертиза.

Інформація, яку піддають експертизі, буває різної властивості. Це можуть бути оцінки важливості різних властивостей об'єктів для відображення їх в якості показників, що враховуються критерієм порівняльної оцінки альтернатив, або оцінки перевагу відносин між сутностями предметної області (ПО), або альтернативні формулювання правил логічного висновку тощо.

Експертиза в її вимірювальному аспекті характерна тим, що в ній інструментом вимірювання є людина. Результати вимірювань, оцінки виносяться людьми, щодо тих чи інших явищ, процесів, закономірностей, властивостей і т.п., є суб'єктивними.

Сьогодні при вирішенні завдання оцінювання альтернатив в більшості випадків використовуються дві групи методів, що ґрунтуються на вирішенні однокритеріальних і багатокритеріальних завдань [8].

Однак число конкретних математичних методів (функціональних залежностей), які можуть бути використані в практичних розрахунках, досить велике.

За кількістю критеріїв оцінки альтернатив виділяють одно і багатокритеріальні задачі. Принципова різниця між цими класами задач полягає в тому, що в умовах багатокритерійності виникає проблема порівняння, сукупного врахування вимог різних критеріїв. На відміну від завдання упорядкування альтернатив за одним єдиним критерієм вона не може бути вирішена формальним шляхом і вимагає звернення до ОПР, організації взаємодії, діалогу між членами експертної комісії [11].

Всі найбільш відомі методи доцільно розділити, як показано на рис. 2.1.



Рис. 2.1 – Класифікація методів вирішення задачі оцінювання, і впорядкування альтернатив

Хоча ця класифікація досить умовна, але вона достатньо продуктивна з двох причин:

- 1) визначається область вибору найбільш підходящого для кожного конкретного завдання методу;
- 2) формуються передумови для розробки нових методів (наприклад, шляхом модифікації, поєднання, комплексування і т.п.).

Як видно з наведеної класифікації та проведеного аналізу літератури [9, 10, 11], області застосування методів розрахунку комплексної оцінки, базуються на згортках метричних характеристик, методів введення метрик в просторі нормованих метричних характеристик і методи експертних оцінок перетинаються, тобто кілька методів можуть бути застосовані до однієї і тієї ж задачі, що обумовлює невизначеність у виборі методу рішення [12].

Найбільш доцільним для вирішення даного завдання є «експертний» підхід, який полягає в наступному:

- проводиться оцінка якості альтернатив всіма допустимими методами з розглянутих, і за її результатами формується матриця «вторинних показників» - «альтернативи на методи»;

- вважаючи результати рішень задачі оцінки усіма методами (матрицю «вторинних показників») думками висококваліфікованих віртуальних експертів («v-експертів»), для її обробки застосовуються методи експертних оцінок.

Це завдання легко могло би бути вирішено, якби був відомий найбільш застосовний метод експертної оцінки для кожного випадку (кожної матриці даних) або універсальний метод, який можна застосовувати у всіх випадках.

2.2 Моделі експертних оцінок в умовах критеріальної невизначеності

Розглянемо переваги і недоліки основних експертних методів.

Метод безпосередньої оцінки застосовується у випадках, коли об'єкти експертизи, що визначають кінцеві результати, піддаються безпосередньому порівнянню, оскільки мають однакову природу, тобто у них є спільний еталон порівняння.

Метод безпосередньої оцінки виконується в бальних шкалах. На відміну від кількісних оцінок, відповідних замірів об'єктивних показників, бальні оцінки зазвичай характеризують суб'єктивну думку.

Значення (градації) бальної шкали являють собою обмежений дискретний ряд чисел, які розташовані одне від іншого на однаковій відстані. Зазвичай при експертних оцінках в якості значень шкали вибирають початковий відрізок натурального ряду або частину ряду цілих чисел, симетричну відносно нуля.

Розрізняють два види бальних оцінок [13].

В першому випадку оцінка проводиться за об'єктивним критерієм, так що індивідуальні оцінки являється деякими флуктуаціями справжніх значень. Зазвичай при цьому є в наявності деякі загальноприйняті еталони, що відповідають градаціям шкали, з якою порівнюються розглядувані об'єкти. Чим

точніше охарактеризовані і можливі відхилення від еталонів, тим менше флуктуації в оцінках, тим більше довіри до них.

Бальна оцінка другого виду проводиться, коли не тільки немає загальноприйнятих еталонів, але й сумнівна навіть наявність якогось об'єктивного єдиного критерія, суб'єктивними відображеннями якого являються оцінки, так що безглуздом являється і саме питання про кількісне відношення оцінок. Такі, наприклад, порівняння гастрономічного смаку різних страв. В такому випадку часто оцінки розглядаються в так званій ранговій чи порядковій шкалі. Рангові оцінки має сенс порівнювати тільки у відношенні «більше – менше»; вони зберігаються при монотонних перетвореннях. Немає сенсу навіть порівнювати довжини інтервалів між оцінками.

Оскільки рангові оцінки мають сенс тільки з точністю до упорядкування по величині, їх можна давати не тільки в числових термінах, але й якості градацій використовувати символи будь-якої упорядкованої множини.

Метод безпосередньої оцінки дозволяє врахувати ступінь переваги якого-небудь одного показника по відношенню до інших, але при визначенні узагальненого показника ефективності, помилкова оцінка експертом менш значного показника, при правильному визначенні його взаємного розташування по відношенню до інших показників може призвести до значного спотворення кінцевого результату [14].

Метод ранжування. Ранжуванням називається процедура упорядкування деякої сукупності однорідних і порівнянних за своїми ознаками понять в порядку зростання або зменшення їх переваги. Результатом ранжування може бути призначення експертом (або експертами) деяких числових оцінок переваг оцінюваним поняттям або даним. Зазвичай ранжовані поняття або дані прийнято називати об'єктами або альтернативами.

Ранжування можна легко представити як оцінку в ранговій шкалі: рангом об'єкта «а» можна вважати номер місця, яке він займає у ранжуванні при зворотній нумерації місць. При цьому вважається, що рівноцінні об'єкти знаходяться на одному і тому ж місці (мають однаковий ранг).

У результаті ранжування може скластися одна з таких ситуацій:

- всі об'єкти отримали різні оцінки переваги;
- в результаті ранжування утворилися одна або кілька груп, всередині яких об'єкти мають однакові оцінки, але все такі групи різняться по загальній перевазі об'єктів що до них входять.

У першій ситуації ранжування (призначення рангів) здійснюється просто: найбільш переважаюча експертом альтернатива отримує вищий ранг, а найменш бажана - нижчий ранг. Якщо оцінювана група альтернатив – злічена множина $\{A_i\}, (i = 1, \dots, n)$, то ранг найкращої з них буде дорівнює (n) , а ранг найгіршої - буде дорівнює 1. Сума всіх рангів оцінкової множини альтернатив або об'єктів є сумою натуральних чисел від 1 до n включно.

$$\sum_{i=1}^n R_i = \frac{n(n-1)}{2},$$

де R_i – ранг i -го об'єкта; n – кількість об'єктів, що ранжуються.

Таке ранжування називається строгим порядком.

У другій ситуації для еквівалентних по перевазі об'єктів, що входять в різні по перевазі групи, ранги обчислюються так. Спочатку об'єкти ранжуються як при строгому порядку, але всі еквівалентні по перевазі об'єкти об'єднуються в групи різної переваги. Отримана таким чином послідовність називається квазістрогою послідовністю. При цьому за кожною групою закріплюється кількість рангів, яка дорівнює кількості об'єктів в групі. Значення рангів, закріплених за групою, беруться в припущенні, що дана послідовність виявилася суворою. Наприклад, може вийти така послідовність рангів: 10, 9, (8, 7, 6), 5, (4, 3), 2, 1.

Тут видно, що утворилося дві групи еквівалентних об'єктів. У першій групі три об'єкти. Їхні номери в квазістрогій послідовності - ($i = 3, i = 4, i = 5$), а зайняті групою ранги - (8,7,6). У другій групі - два об'єкти ($i = 7, i = 8$), а зайняті групою ранги - (4 і 3). Після такого угруповання визначається ранг R кожної (i - ої) групи еквівалентних по перевагу об'єктів:

$$R_i^G = \frac{\sum_{j=i}^{i+(k-1)} R_j}{k}$$

де $j = i$ - номер першого, а $j + (k - 1)$ - номер останнього об'єкта i -ї групи об'єктів;

k - число еквівалентних об'єктів в групі.

Для взятої в прикладі квазістрокої послідовності групові ранги еквівалентних об'єктів будуть відповідно рівні $(8 + 7 + 6) / 3 = 7$ і $(4 + 3) / 2 = 3,5$.

Одна з основних проблем, яка виникає при обробці експертної інформації – проблема вибору найкращої альтернативи. При удаваній простоті упорядкування об'єктів шляхом ранжування цей метод має багато підводних камнів, пов'язаних з вибором найкращої альтернативи.

Найбільш раннє дослідження відноситься до кінця XVIII століття і належить французьким вченим Кондорсе та Борда [15].

Кондорсе вперше звернув увагу на неточність процедури виявлення найкращої альтернативи за допомогою безпосереднього підрахунку голосів за правилом більшості

Наприклад, нехай розглядається 20 альтернатив a_1, \dots, a_{20} . Їх ранжування, що відповідає думкам 10-ти експертів такі:

P_1	P_2	P_3	P_4	P_5	...	P_{10}
a_2	a_2	a_1	a_3	a_4	...	a_9
a_1	a_1	a_3	a_1	a_1	...	a_1
a_3	a_3	a_4	a_4	a_3	...	a_3
a_4	a_4	a_5	a_5	a_5	...	a_4
...
a_{19}	a_{19}	a_2	a_2	a_2	...	a_2

За правилом більшості підраховується кількість експертів, що надали перевагу кожній з альтернатив, та найкращою оголошується та, яку назвали більшість експертів.

В нашому прикладі це альтернатива a_2 . Навряд чи таке рішення здасться справедливим, але a_2 можна оголосити і найгіршою альтернативою. Інколи при використанні правила більшості вводять додаткові вимоги, що дозволяють усунути даний недолік. Зокрема, найкращою може бути альтернатива, яку вважають не менше половини експертів. Але може статись так, що такої альтернативи не існує.

Кондорсе запропонував наступний принцип визначення найкращої альтернативи. Кожен експерт ранжує альтернативи по перевагах. На підставі отриманого ранжування для кожної пари альтернатив a_i, a_j підраховується S_{ij} – кількість експертів, що вважають a_i більш переважачою ніж a_j . Якщо $S_{ij} > S_{ji}$, то альтернатива a_i вважається більш переважачою ніж a_j , a_i оголошується найкращою альтернативою, якщо $S_{ij} > S_{ji}$ для всіх $j \neq i$. Але при використанні принципу вибору Кондорсе може виникнути вказаний ним же парадокс, що є наслідком нетранзитивності колективних переваг.

Це можна продемонструвати на прикладі. Нехай три експерти. Проранжували альтернативи a_1, a_2, a_3 , наступним чином:

$$P_1 = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}, \quad P_2 = \begin{pmatrix} a_3 \\ a_1 \\ a_2 \end{pmatrix}, \quad P_3 = \begin{pmatrix} a_2 \\ a_3 \\ a_1 \end{pmatrix}.$$

Тоді $S_{12} > S_{21}, S_{23} > S_{32}$, але $S_{13} > S_{31}$. Альтернативи Кондорсе в цьому випадку не існує. Як показала практика використання цього методу, рішення відсутнє в 7-10% від реальних ситуацій експертного оцінювання[16].

Необхідним враховувати ранжування при визначенні найкращої альтернативи вважав також і Борда. Спосіб, запропонований ним, полягає в наступному.

Альтернативам, проранжованим експертом, приписують числа: останній по перевазі 0, передостанній 1 і т. д. Рахується сума балів, приписана кожній із альтернатив, і альтернативи ранжуються в порядку їх сум.

Спосіб Борда також не без недоліків. Зокрема, альтернатива Кондорсе, тобто альтернатива, яка краща будь-якої іншої при парному порівнянні альтернатив, може виявитися не вибраною в якості найкращої. Дійсно, нехай 5 експертів проранжували альтернативи a_1, a_2, a_3 наступним чином.

$$P_1 = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}, \quad P_2 = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}, \quad P_3 = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}, \quad P_4 = \begin{pmatrix} a_2 \\ a_3 \\ a_1 \end{pmatrix}, \quad P_5 = \begin{pmatrix} a_2 \\ a_3 \\ a_1 \end{pmatrix}.$$

За способом Борда найкращою альтернативою буде a_2 , оскільки $S_2 = 7$, $S_1 = 6$, $S_3 = 2$. Але найкращою альтернативою за правилом більшості голосів є a_1 , в цьому випадку більшість експертів проголосували за a_1 .

Метод ранжування і його різновиди [9] можуть застосовуватися в наступних випадках:

- коли необхідно впорядкувати будь-які об'єкти або явища в часі або просторі. Ця ситуація виникає, коли необхідно визначити не ступінь вираженості будь-якого їх якості, а лише взаємне просторове або тимчасове розташування цих об'єктів;

- при необхідності упорядкувати об'єкти відповідно до будь-яким якістю, але при цьому не потрібно проводити його точне вимірювання;

- якщо яка-небудь якість вимірюється, але в даний момент не може бути виміряна за причин практичного або теоретичного характеру.

При використанні методу ранжування на отримання кінцевого результату впливає правильний вибір його різновиди для конкретної ситуації. Точність і надійність процедури ранжирування в значній мірі залежать від кількості об'єктів. Чим більше таких об'єктів, тим нижче їх «розрізнення» з точки зору експерта, а, отже, тим менше надійно можна встановити ранг об'єкта, що є істотним недоліком методу.

До основних недоліків методу ранжування можна також віднести втрату інформації про оцінюваних об'єктах внаслідок упорядкування їх лише за взаємною розташуванню без врахування ступеня вираженості якого-небудь їх якості. Тому метод ранжування в «чистому вигляді» використовується не часто. Найчастіше він поєднується з іншими методами, що забезпечують більш чітке розходження між об'єктами. Одним з них є метод безпосередньої оцінки і деякі його різновиди (ранжування за порівнянною шкалою) [10].

Метод парних порівнянь. Метод парних порівнянь застосовується у випадках виявлення переваг для великого числа об'єктів і в випадках, коли відмінності між об'єктами настільки малі, що безпосередня оцінка або ранжирування не забезпечують їх розумного упорядкування.

Результати парного порівняння представляються у вигляді матриці парних порівнянь. Якщо експертні оцінки порівняння переваг при парному порівнянні довільної послідовності альтернатив задані у вигляді трьох функцій:

$$P_{ij} = 2, \text{ якщо } A_i > A_j;$$

$$P_{ij} = 0, \text{ якщо } A_i < A_j ;$$

$$P_{ij} = 1, \text{ якщо } A_i = A_j ,$$

то матриця парних порівнянь чотирьох ($n = 4$) альтернатив A_{ij} ($i = 1, \dots, n; j = 1, \dots, n$) буде мати вигляд:

i	$A(i=1)$	$A(i=2)$	$A(i=3)$	$A(i=4)$
j				
$A(j=1)$	1	2	0	0
$A(j=2)$	0	1	0	2
$A(j=3)$	2	2	1	2
$A(j=4)$	2	0	0	1

Рисунок 2.2 – Матриця парних порівнянь альтернатив з оцінками порівнянь

В кожен клітинку таблиці заноситься результат порівняння двох альтернатив, позначених в стовпці і в рядку. В клітинку ставиться одиничка, якщо альтернатива по рядку має більшу перевагу, ніж альтернатива, що відповідає стовпцю. Слід особливо звернути увагу на випадок, коли окремий респондент не може надати перевагу одній альтернативі перед іншою, тоді в клітинку вноситься 0,5.

Таким чином заповнюється вся таблиця. Після чого по кожному рядку підраховується число одиниць, тобто скільки разів альтернатива виявилася переважаючою над іншою.

Але в результаті використання методу парних порівнянь отримується метрична шкала, це доведено виходячи з того, що в динаміці переваги окремого респондента характерний нормальний закон розподілу. Це значить якщо опитувати респондента багато разів, то кожного разу він буде давати різні оцінки.

Метод парного порівняння без додаткової обробки і ряду обмежень не дає повного упорядкування об'єктів [9], що є його істотним недоліком.

Розглянуті вище три методи вирішення (безпосередньої оцінки, ранжування і парного порівняння) мають різні властивості, але призводять до близьких результатів.

Тому вибір методу «вторинної» обробки (ранжування або парні порівняння) пропонується проводити за кількістю оцінюваних об'єктів і невеликій відмінності між ними [11].

Оскільки поняття «невелика відмінність» не є чіткою фізичною величиною, а її визначення вимагає додаткових досліджень, то за критерій вибору пропонується брати «число оцінюваних об'єктів» (k). Якщо кількість ранжируваних об'єктів $n < k$ - метод ранжирування, а при $n > k$ - метод парних порівнянь. Слідуючи рекомендаціям [9], k не повинно бути більше 20, а найбільш надійна ця процедура, коли $k < 10$.

2.3 Розробка методу комплексної оцінки альтернатив в умовах критеріальної невизначеності

З огляду на вищевикладене, очевидно, що підвищити стійкість результату можливо за рахунок отримання комплексної оцінки за результатами рішення задачі вибору всіма можливими методами з представлених, якби був відомий спосіб обробки таких результатів [14].

В якості такого способу і пропонується використовувати Метод Комплексної Оцінки Альтернатив (МКОА), що забезпечує ефективне рішення за рахунок отримання стійкого результату за сукупністю методів.

Сутність запропонованого методу полягає в наступному.

Опис порівнюваних альтернатив, кожна з яких представлена вектором характеристик, задається вихідної матрицею $A (m \times n)$, де m - число порівнюваних варіантів альтернатив (рядків матриці A), n - число характеристик (стовпців матриці A).

Кожен елемент матриці A є значенням характеристики X_{ji} , $j = 1, m, i = 1, n$, тобто

X_{ji} , є значенням i -ї характеристики j -ї альтернативи. Матриця A має наступний вигляд:

Таблиця 2.1 – Вихідні дані в методі комплексної оцінки альтернатив

Альтернативи	Показники якості					
	C_1	C_2	-----	C_i	-----	C_n
V_1	X_{11}	X_{12}	-----	X_{1i}	-----	X_{1n}
V_2	X_{21}	X_{22}	-----	X_{2i}	-----	X_{2n}
-	-	-	-----	-	-----	-
V_j	X_{j1}	X_{j2}	-----	X_{ji}	-----	X_{jn}
-	-	-	-----	-	-----	-
V_m	X_{m1}	X_{m2}	-----	X_{mi}	-----	X_{mn}

Задається деяке нормуюче відображення множини характеристик альтернативи $\{X\}$:

$$\{X_{ij}\} \xrightarrow{F} \{K_{ij}\} (i = 1, n; j = 1, m),$$

де X_{ij} - i -та характеристика j -ї альтернативи;

K_{ij} - оцінка якості j -ї альтернативи j -ї властивості.

Результатом такого відображення F є матриця A' ($m \times n$), де m - число порівнюваних альтернатив (рядків матриці A'), n - число показників якості за різними властивостями альтернативи (стовпців матриці A').

Кожен елемент матриці A' є значенням показника якості альтернативи по $X_{ij}, j = 1, m = 1, n$, тобто K_{ij} , є значенням i -го показника j -го варіанта альтернативи.

Матриця A' має наступний вигляд:

Таблиця 2.2 – Показники якості альтернатив

Альтернативи	Показники якості					
	P_1	P_2	-----	P_i	-----	P_n
V_1	K_{11}	K_{12}	-----	K_{1i}	-----	K_{1n}
V_2	K_{21}	K_{22}	-----	K_{2i}	-----	K_{2n}
-	-	-	-----	-	-----	-
V_j	K_{j1}	K_{j2}	-----	K_{ji}	-----	K_{jn}
-	-	-	-----	-	-----	-
V_m	K_{m1}	K_{m2}	-----	K_{mi}	-----	K_{mn}

Для m рядків матриці A' при вирішенні завдання вибору кожним з допустимих методів визначається вектор дійсних невід'ємних чисел $(KO_{j1}, KO_{j2}, \dots, KO_{ji}, KO_{js})$, які складають комплексну оцінку відповідних альтернатив, $j = 1, m$, де m – число альтернатив. Отримуємо матрицю B :

Таблиця 2.3 – Комплексні оцінки якості альтернатив

Альтернативи	Показники якості					
	M	M_2	-----	M_i	-----	M_s
V_1	KO_{11}	KO_{12}	-----	KO_{1i}	-----	KO_{1s}
V_2	KO_{21}	KO_{22}	-----	KO_{2i}	-----	KO_{2i}
-	-	-	-----	-	-----	-
V_j	KO_{j1}	KO_{j2}	-----	KO_{ji}	-----	KO_{js}
-	-	-	-----	-	-----	-
V_m	K_{m1}	K_{m2}	-----	K_{mi}	-----	K_{ms}

Далі, вважаючи, що оцінки кожним з допустимих методів є оцінками «v-експертів», проводим «вторинну» обробку отриманої інформації або методом ранжування, або методом парних порівнянь, використовуючи як критерій вибору методу «кількість об'єктів» знаходимо результуючу оцінку

(k):

$n < k$ - метод ранжування;

$n > k$ - метод парних порівнянь, де $k = 10$.

2.4 Аналіз методів оцінки компетентності експертів

Оцінка компетентності експерта є однією з актуальних проблем експертного аналізу. Існуючі в даний час підходи до даної проблеми базуються або на суб'єктивних судженнях, результатах психологічних і соціальних досліджень [19], або на аналізі оцінок, виставлених експертами об'єктам експертизи. Так, наприклад, вважається в роботах Буркова Е.А. і ін. [20]. Згідно з цим підходом компетентність експерта слід оцінювати по тому, наскільки узгоджуються його оцінки з оцінками більшості [24]. Інакше кажучи, в основу даного методу покладено аксіома незсуненості [22]. Згідно цієї аксіоми компетентною є думка більшості, і, отже, найбільш компетентним слід вважати

того експерта, думка якого має мінімальне розходження з думкою більшості експертів. Дане положення стало підставою для розробки ряду методів аналізу і агрегування групових експертних оцінок. Як приклад можна запропонувати роботи П.І. Падерно і Е.А. Буркова [18, 20].

Однак даний підхід навряд чи можна назвати коректним. Некоректність полягає в наступному протиріччі: аксіома незсуненості вірна за умови, що більшість експертів мають високу компетенцією, а якщо це так, то рішення задачі оцінки експертів втрачає сенс. В результаті процедура, що базується на аксіомі незсуненості, ніяк не може бути віднесена до методу оцінки компетентності експертів. Вона є виявленням некомпетентних експертів на тлі більшості експертів з високою компетентністю.

В сучасних умовах, коли рівень освіти істотно впав, а вплив суб'єктивних факторів (зокрема віднесення людей до експертів з формальних чинників) зросла, процедури оцінки компетентності експертів, що базується на аксіомі незсуненості, будуть відігравати негативну роль. На тлі некомпетентної більшості експерт з високою кваліфікацією приречений на отримання низької оцінки компетентності.

Відомі методи взаємооцінки та самооцінки експертів [23]. Взаємооцінка здійснюється двома способами. У першому випадку, кожен кандидат в члени експертної комісії оцінює компетентність інших експертів [21]. У другому випадку оцінку якості запропонованих експертів здійснює аналітична група, якій доручено організацію і проведення експертизи.

При самооцінці визначення ступеня знайомства з предметом експертизи, компетентності і т. п. в деталізованому вигляді здійснюється самим експертом.

Взаємооцінка і самооцінка експертів може носити як якісний, так і кількісний характер.

Б.Г. Литвак пропонує для оцінки компетенції експертів використовувати метод оцінки несуперечності суджень експерта [6]. Даний метод ґрунтується на гіпотезі, що експерт часто вже не послідовний у своїх оцінках. Особливо

непослідовність експертів проявляється при використанні методу парних порівнянь.

Число суперечливих суджень може бути відносно мало або велике. Чим менше їх висловлює експерт, тим більш послідовний він у своїх оцінках. У монографії Б.Г. Литвака [23] метод оцінки несуперечності суджень експерта досить детально описується. Він цікавий, проте має скоріше академічну, ніж практичну цінність, його реалізація пов'язана з низкою проблем, зокрема для оцінок за цим методом необхідно мати великі архіви, що містять висновки оцінюваних експертів. На підставі їх аналізу та отриманих статистичних даних робляться оцінки. Крім того, метод складний для реалізації і не гарантує об'єктивних результатів, оскільки обробкою даних архівів повинні займатися ті ж експерти, рівень компетентності яких, як правило, невідомий.

Б.Г. Литвак згадує так званий документаційний метод оцінки якості експерта [23]. Суть його полягає в аналізі таких документальних даних, як число публікацій і посилань на роботи експерта, науковий ступінь, стаж, посада і т. д. У сучасних умовах даний метод виявляється неефективним в силу «засміченості» експертного співтовариства некомпетентними фахівцями з вченими ступенями, що в підсумку призвело до великої кількості (тисячі) об'єктів оцінювання. Для обробки справжнього обсягу інформації потрібні значні часові та фінансові ресурси.

Таким чином, проблема оцінки компетентності експертів пов'язана з проблемою вимірювання знань. В даний час найбільш відомі два способи таких вимірів: тестування і атестація.

Тестовий метод передбачає відбір експертів на підставі рішення ними тестових завдань, в яких відображена специфіка предмета експертизи. Як тест можуть розглядатися результати участі експерта в аналогічних експертизах, вважає Б.Г. Литвак. Об'єктивні оцінки компетенції експертів можна отримати шляхом аналізу їх наукових праць. Однак дані підходи пов'язані з певними труднощами, пов'язаними з обмеженнями для прийняття рішення часом,

складнощами з доступом до необхідної інформації і дефіцитом неформальних експертів.

Атестацію також не можна віднести до ефективного методу відбору, оскільки її результати, як правило, формальні і чутливі до загального наукового рівня організації.

Таким чином, розглянуті способи непридатні для вирішення поставленої проблеми.

Звідси випливає постановка наступного актуального завдання.

Нехай $\epsilon Q = Q_1 + Q_2$ експертів, де Q_1 - множина неформальних експертів; Q_2 - множина формальних експертів. Неформальні експерти - це експерти, що володіють науковим ім'ям і значущими науковими результатами; формальні експерти - це експерти, які відповідають формальним показникам, але не мають значущих наукових результатів.

Значний науковий результат - це нове знання, отримане в процесі фундаментальних або прикладних наукових досліджень та зафіксоване на носіях наукової інформації в формі звіту, наукової праці, наукової доповіді, наукового повідомлення про науково-дослідну роботу, монографічного дослідження, наукового відкриття і т. д. Таким чином, значний науковий результат - це науковий результат відомий і признаний широкою науковою громадськістю [19].

Потрібно з множини Q виділити підмножину Q_1 . Для вирішення поставленого завдання необхідно розробити безліч критеріїв для даного відбору. Ця задача не є тривіальною, оскільки $Q_1 \ll Q_2$. Крім того, існує безліч інших проблем, що ускладнюють вирішення завдання вибору і обґрунтування безлічі критеріїв. Так, зокрема, високі значення формальних показників, як, наприклад, індекс наукового цитування чи участь в міжнародних проектах як правило, не є об'єктивними показниками компетентності експерта в силу значного впливу на дані показники зовнішніх впливів [20].

Метод абсолютного авторитету. Суть його полягає в наступному. На множині Q визначається «абсолютний авторитет» - такі вчені, чий авторитет не

викликає сумнівів. До «абсолютного авторитету» звертаються з проханням назвати найбільш значущих експертів в його області знань.

До виявлених таким чином фахівців звертаються з подібним проханням.

Таким чином, число виявлених експертів зростає з геометричною прогресією, але процес завершується на другому - третьому кроці, оскільки кількість неформальних експертів суттєво обмежена. Дана процедура відома, під назвою «снігова куля».

Слід зазначити, що даний метод дуже чутливий до початкових умов, тобто залежить від якості «абсолютного авторитету»; якщо дане якість не відповідає встановленим вимогам, то наступний відбір експертів втрачає сенс.

Процедура не закінчується відбором експертів за запропонованим алгоритмом. Відібрати експертів - це не означає вирішити поставлене завдання, оскільки велика частина експертів з різних причин відмовиться співпрацювати (зайнятість, відсутність мотивації та ін.). Слід зазначити, що проблема мотивації є досить актуальною при організації діяльності експертної спільноти.

Частину експертів, яка висловлюватиме свою згоду співпрацювати, слід протестувати на їх можливість оперативно працювати, здатність виконувати взяті на себе зобов'язання.

Виявлення даних якостей експертів є визначальним, оскільки в умовах стрімко мінливих умов обстановки різко зросла необхідність оперативно і кваліфіковано приймати рішення. Це вимагається не тільки від експертів, але і від замовника і організаторів проведення експертизи [22].

Зазвичай на реалізацію перерахованих процедур йдуть роки, в результаті формується група експертів, яку можна називати спільнотою. Склад цієї групи стабільний і знаходиться в динамічній рівновазі, щороку хтось з різних причин її покидає і їх місця займають знову.

2.5 Висновок

У даному розділі визначено, якими будуть особливості розробленого програмного продукту, а саме використання композиції методів експертного оцінювання альтернатив та методу групового прийняття рішень.

Розглянуто і проаналізовано методи реалізації процесу прийняття рішень в умовах критеріальних невизначеності, серед яких обрано та застосовано комбінований алгоритм. Аналіз методів експертної оцінки показав, що метод ранжувань корисний, коли кількість альтернатив менша 10. Для випадку, коли кількість альтернатив більша 10 – доцільніше використовувати метод парних порівнянь.

Для оцінки компетентності експертів, серед методів самооцінки, методу аналізу узгодженості експертів та методу групової взаємооцінки обрано останій. Цей метод дозволяє розглядати експертів як альтернативи та за допомогою методу визначення важливості альтернатив провести оцінку компетентності експертів.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРИЙНЯТТЯ РІШЕНЬ В УМОВАХ КРИТЕРІАЛЬНОЇ НЕВИЗНАЧЕНОСТІ

3.1 Обґрунтування вибору програмно-апаратної платформи

У наш час не можна уявити людину, що не використовує інформаційні технології. Але якщо в двадцятому столітті передовими технологіями вважалися радіо, телебачення та телефонний зв'язок, то в двадцять першому столітті на перший план виходить Інтернет. Гігантські обсяги текстової, аудіо-відео інформації, новинні стрічки, блоги, соціальні мережі, ігри та багато чого, багато чого іншого. В даний час підключитися до Інтернету можна через супутники зв'язку, радіоканали, кабельне телебачення, телефон, стільниковий зв'язок, спеціальні оптико волоконні лінії або електропроводу. Всесвітня мережа стала невід'ємною частиною життя в розвинених і країнах, що розвиваються.

В кінці 1990-х років типова веб-сторінка була текстом з гіперпосиланнями і навігаційним меню з невеликими зображеннями - це була, фактично, гіпертекстова версія газетної сторінки. Мова HTML на той момент повністю задовольняв технічним потребам Інтернет-павутини. Настільні і веб-додатки радикально відрізнялися один від одного як за функціями, так і за зовнішнім виглядом [24].

Однак в процесі розвитку мережі з'явилася потреба в динамічному, інтерактивному вмісті веб-сторінок і складнішому форматуванні і оформленні, ніж міг надати HTML. Набуло поширення динамічне формування сторінок на стороні сервера за допомогою таких технологій, як CGI, Perl, ColdFusion, PHP, ASP. Для обробки простих інтерактивних сценаріїв на стороні користувача в браузері була додана підтримка JavaScript. Складна верстка і оформлення засобами HTML виявилися практично неможливими, і для опису зовнішнього вигляду став застосовуватися мова CSS. Відсутність клієнтського сховища даних компенсували за допомогою HTTP Cookie. Жодна з цих технологій не

передбачалася стандартом HTML, вони застосовувалися в якості вирішення практичних обмежень на тому чи іншому етапі.

Розробники, фактично, намагалися наблизити все більш складні вебсторінки (які тепер вже прийнято називати вебпріложеній) до рівня настільного програмного забезпечення за часом відгуку і швидкості роботи, функціональності і можливостям інтерактивної взаємодії, багатством оформлення. Але технологічні обмеження (в першу чергу - необхідність працювати через віддалені інтерпретатори) ускладнювали цю задачу.

Слід зазначити, що при розвитку вебчастини Інтернету постійно був і залишається актуальним питання сумісності сторінок з різними браузерами і платформами. У той час як базові стандарти (HTML, JavaScript, CSS) єдині, тонкощі їх реалізації та конкретні алгоритми рендеринга сторінок в різних системах можуть відрізнятися.

В якості вирішення проблеми сумісності веб-додатків, інтерактивності та задіяння ресурсів комп'ютера клієнта компанія Sun запропонувала аплети Java - програми, які виконуються у вікні браузера як частина веб-сторінки за допомогою віртуальної машини Java, доступною на більшості платформ. Java-аплети не отримали широкого розповсюдження, в першу чергу, через складність створення привабливого користувальницького інтерфейсу. З'явилися також орієнтовані на медіа-контент вбудовуються в веб-сторінки технології, такі як Shockwave, QuickTime і RealMedia. Найбільшого ж успіху вдалося досягти платформі Macromedia Flash, яка сьогодні встановлена на більш ніж 99% настільних комп'ютерів [25], і використовується на чверті веб-сайтів Інтернет [24]. Flash надає одночасно відмінний інструментарій для створення анімації і інтерфейсів (зокрема, за допомогою суміжній технології Flex), вбудована скриптова мова з розвиненими бібліотеками, можливості по роботі з аудіо і відео і універсальний плагін-плеєр для виконання Flash-програм. Компанія Microsoft має власну технологію Silverlight, у багато аналогічну Flash.

Для обміну даних між веб-сторінкою та веб-сервером традиційно необхідно було перезавантажувати сторінку повністю, що викликало суттєву

тимчасову затримку і повторний рендеринг. У 2005 році після зворотної інженерії веб-додатків компанії Google став широко відомий метод, який дозволяв в асинхронному режимі обмінюватися повідомленнями з браузером без перезавантаження сторінки - так званий Ajax. Такий підхід дозволив істотно наблизити інтерактивність інтерфейсу веб-додатків до настільним програмам. У той же час, підхід був заснований на використанні нестандартного розширення JavaScript, що згодом призвело до ускладнення цього спочатку простого мови, появи додаткових стандартів (JSON, W3C XMLHttpRequest) і спеціалізованих бібліотек (jQuery, Ext JS і т.п.). В результаті сьогодні JavaScript-програмування є окремою спеціальністю, досить складною в освоєнні.

В останні кілька років набули поширення мобільні операційні системи нового покоління - в першу чергу, iOS і Android. Основні програми цих систем мають властивості настільних додатків (виконуються локально і активно використовують апаратні ресурси), проте активно використовуються і веб-додатки, що запускаються як через вбудований браузер, так і нарівні зі звичайними додатками. За оцінками Morgan Stanley, до 2015 року веб буде частіше проглядатися на мобільних пристроях, ніж на персональних комп'ютерах і ноутбуках [26].

Слабка технічна реалізація Flash-плеєра і постійне виявлення вразливостей в різних його реалізаціях обмежило можливості розвитку цієї технології.

Одним з факторів, які не дозволили Flash стати домінуючою інтерактивною платформою Інтернет-павутини, стала відмова компанії Apple підтримувати її в браузерах пристроїв iPhone і iPad. У світлі останніх новин [26], можна констатувати спад інтересу до всіх вбудовуваним в веб-сторінки технологіям.

У HTML 5 передбачена підтримка ряду нових специфікацій, що розширюють можливості мови в області відображення 2D і 3D графіки і анімації, медіаконтенту, геолокації, мікроформатів, офлайнового режиму сторінок, drag-and-drop, підтримки файлової системи і інших. Очевидно, веб-додатки зробили ще один крок до рівня настільного програмного забезпечення [24].

Тут слід зазначити, що по суті, веб-програми та настільні додатки виконують одну і ту ж задачу: надати інформацію і функціональність на екран користувача. Головна відмінність полягає в тому, що велика частина логіки і даних веб-додатки розташовані на сервері. Це дозволяє, з одного боку, уникнути процесів встановлення та оновлення, а з іншого - робить користувача залежним від доступності сервера, вимагає його ідентифікації та роботи через програму-посередника, браузер. На прикладі Apple AppStore і MacStore можна бачити, що для настільного програмного забезпечення також можливо вдосконалити і спростити схему доставки функціональності користувачеві [23].

Як видно, на протязі довгих років HTML розвивався екстенсивним чином. На даний момент мова збагатилась великою кількістю побічних технологій з різною логікою і інструментарієм. Виникає питання - яким чином технологія початку 90-х протягом 20 років займає цін-тральних положення в Інтернет-павутині, суцільно наповненою інноваціями?

По-перше, дотримання єдиної технології обумовлено вимогою забезпечення сумісності. Веб-ринок досить сильно диверсифікований і єдиний діючий контролюючий орган W3C не має достатнього впливу і ресурсів для революційного впровадження нової, що відповідає сучасним реаліям Інтернет технології. Для порівняння, що належать окремим компаніям технології Flash і Silverlight активно розвивалися, втрачаючи від версії до версії повну зворотну сумісність на рівні коду (що компенсується збереженням зворотної сумісності на рівні віртуальної машини). Також складність технологій викликає інертність користувачів, фахівців і організацій, яким непросто буде перебудуватися на нові моделі.

По-друге, позначається дефіцит якісних універсальних альтернатив HTML. Інтернет-павутина - це, безперечно, найбільш розвинена реалізація ідеї гіпертексту основними платформами для клієнтських систем (Windows, Mac / OSX і Unix) і мобільними пристроями (iOS, Android), можна назвати тільки мови C / C ++. Певні можливості надають Adobe AIR (припинена підтримка Unix), Java (не підтримує на iOS), Microsoft .NET (працюють поза Windows тільки через

зв'язку Mono / MonoTouch), але здебільшого, єдине універсальне рішення на сьогодні - веб-додаток.

І, нарешті, найважливіша причина полягає в комерційному успіху Інтернет-павутини. Стрімко розвинулася Інтернет-індустрія привернула величезну кількість коштів, фахівців і користувачів.

Концентрація людських потоків і даних на більш успішних веб-сайтах привела до утворення найбільших корпорацій, таких як Google, Amazon, Yahoo!, eBay, Facebook. Сучасна структура веб-технологій, незважаючи на очевидні мінуси, влаштовує більшість великих гравців, які могли б сприяти інноваціям. Зокрема, відсутність вбудованої в Інтернет павутину моделі пошуку, забезпечує Google понад мільярд переглядів сторінок щодня, а реклама веб-сайтів приносить 96% прибутку [26]. Інші компанії використовують концентрацію інформації в своїх руках, обумовлену клієнт-серверної архітектурою веб-сайтів, в комерційних цілях; так, наприклад, соціальна мережа Facebook використовує особисті дані користувачів для таргетування реклами мережі Microsoft.

Централізацію можна назвати таким, що очолює трендом всієї ІТ-індустрії останньої пари років; незважаючи на експоненціальне зростання продуктивної потужності і дискового простору клієнтських станцій, «хмарні» технології впроваджуються повсюдно, приносячи, з одного боку, зручність використання з будь-якої точки планети, а з іншого - проблеми з безпекою даних і навантаження на мережеву інфраструктуру [27].

Всесвітня павутина (World Wide Web, Інтернет) - розподілена система, що надає доступ до пов'язаних між собою документів, розташованих на різних комп'ютерах, підключених до Інтернету.

Всесвітня павутина ґрунтується на архітектурі клієнт-сервер. І на сервері, і на клієнті має бути встановлено додаткове програмне забезпечення - Веб-сервер і Веб-оглядач відповідно. Це програмне забезпечення підтримує стандартний протокол передачі гіпертекстів (HTTP), а також ряд інших протоколів, зокрема, протокол передачі файлів (FTP). Архітектура клієнт-сервер означає, що взаємодія користувача комп'ютера з Інтернетом відбувається наступним чином:

- користувач вводить адресу (URI або URL) веб-документа, який він хоче переглянути;
- Веб-оглядач формує відповідний HTTP-запит до Веб-сервера;
- Веб-сервер знаходить в Мережі запитаний документ і передає його оглядачеві як відповідь на запит (в кінцевому результаті, веб-сервер знаходить відповідний файл на локальному жорсткому диску і відправляє його по мережі);
- оглядач інтерпретує отриманий документ і відображає його користувачеві (відображає гіпертекст).

Для правильної і надійної роботи мереж були розроблені так звані протоколи. Найвідоміші з них: IPX / SPX і TCP / IP. Про останній і піде мова в даній роботі, тому що стек протоколів TCP / IP стає все популярнішим у світі.

До мережі на базі протоколу TCP / IP приєднано обчислювальні системи різних типів. Набір протоколів, з яких складається TCP / IP, спочатку був розроблений для забезпечення обміну даними між комп'ютерними системами різних типів так, ніби робота велася б в одній системі. TCP / IP був створений в рамках проекту під керівництвом одного з агентств Міністерства оборони США, що займається передовими дослідницькими розробками.

Ранні версії протоколу TCP / IP набули популярності з багатьох причин. Виділимо три основні [28].

По-перше, агентство DARPA надало субсидію на включення даного набору протоколів в систему UNIX університету в Берклі. Після того як TCP / IP був представлений на комерційному ринку, в розмові про нього завжди згадувалась також і UNIX. Операційна система UNIX та протокол TCP / IP стали звичними для більшості великих університетів, де вони використовувалися на робочих станціях в інженерних і дослідницьких системах.

По-друге, в 1983 році всі пропозиції уряду Сполучених Штатів, які мали відношення до комп'ютерних мереж, передбачали використання протоколу TCP/IP. Це був той самий рік, коли мережу ARPAnet перевели на протокол TCP/IP. У той час перехід стався в лічені дні – інтернет був ще невеликий.

По-третє, був розроблений графічний інтерфейс для зручного доступу до ресурсів мережі. Протокол TCP / IP (або його застосування) було важко використовувати тим, хто не мав досвіду роботи з ним. Пошук інформації в інтернеті був складним завданням. До появи програми перегляду - так званого браузера (browser) - доступ до застосунків, які використовували TCP / IP, здійснювався з командного рядка за допомогою декількох базових програм. Вони дозволяли викликати віддалену систему і використовувати її в якості терміналу, надсилати файли, а також надсилати і отримувати пошту. Деякі виробники створили для подібних додатків графічні інтерфейси, які, однак, були не зручними і комерційного успіху не мали. Браузер приховав всі складності протоколу TCP / IP і його додатків, дозволивши працювати з графікою так само просто, як з текстом, клацнувши мишею по виділеному тексту або по зображенню, можна було переміститися в будь-яку точку мережі інтернет (з урахуванням вимог безпеки). Браузер також зробив більш легким доступ до інформації.

Незабаром всі оцінили можливість цього протоколу забезпечувати мережеву взаємодію різнорідних комп'ютерних систем. Це не вимагало суттєвого оновлення апаратної частини мейнфреймів, міні персональних комп'ютерів. Оскільки протокол просто встановлюється на існуючі комп'ютерні пристрої, TCP / IP став вельми популярним мережевим протоколом і це положення зберігається і сьогодні.

Серед інших основних мережевих моделей виділяють OSI. Вона також як є багаторівневою, проте, якщо модель tcp ip розділена на 4 рівні, то OSI - на 7. Вони обидві мають ідентичні рівні, на які покладено однакові завдання.

Кожен з рівнів моделі ISO OSI називається стеком, а протоколи, які відповідають цим рівням - стеком протоколів. Кожен рівень стека залежить від рівня, що знаходиться нижче нього. Це означає, що кожен рівень надає будь-які служби вище стоячого рівня. При відправленні даних з одного комп'ютера на інший дані проходять всі рівні зверху вниз на комп'ютері-відправника і від низу до верху на комп'ютері-одержувачі.

У табл. 3.1 показано, як протоколи TCP / IP вписуються в модель OSI [18]:

Таблиця 3.1 – Протоколи TCP/IP в моделі OSI

7	Прикладний	HTTP, SMTP, SNMP, FTP, Telnet, scp, SMB,NFS, RTSP, BGP
6	Представницький	XDR, ASN.1, AFP
5	Сеансний	TLS, SSL, ISO 8327 / CCITT X.225, RPC, NetBIOS, ASP
4	Транспортний	TCP, UDP, RTP, SCTP, SPX, ATP, DCCP, GRE
3	Мережевий	IP, ICMP, IGMP, CLNP, OSPF, RIP, IPX, DDP
2	Канальний	Ethernet, Token ring, PPP, HDLC, X.25, Frame relay, ISDN, ATM, MPLS, Wi-Fi, ARP, RARP
1	Фізичний	Електро-провідники, радіозв'язок, оптоволоконні мережі

На рисунку 3.1 показано як рівні моделі TCP / IP можна порівняти з рівнями моделі OSI [25, 26].

OSI		TCP/IP
7 6	WWW, Gopher, WAIS, SNMP, FTP telnet, SMTP, TFTP	I
5 4	TCP, UDP	II
3	IP, ICMP, RIP, OSPF, ARP	III
2 1	Не регламентується Ethernet, Token Ring, FDDI, X.25, SLIP, PPP	IV

Рисунок – 3.1. Порівняння моделі OSI та стека TCP/IP

Найнижчий (рівень IV, можна умовно його назвати "Канальний") Канальний рівень вирішує наступні завдання:

- організація логічного з'єднання між взаємодіючими вузлами;
- узгодження в рамках з'єднання швидкостей передавача і приймача інформації;
- забезпечення надійної передачі, пошук і виправлення помилок.

На цьому рівні відбувається формування фреймів - власних протокольних одиниць. Фіксуючи межі кадра можна забезпечити надійну передачу даних і виявити помилки. Функція корекції помилок не є обов'язковою, тому в деяких реалізаціях цього рівня вона може бути відсутньою [27].

Наступний рівень (рівень III, можна умовно його назвати "Мережевий") - утворює єдину мережу і виконує наступні функції:

- передача даних з однієї мережі в іншу;
- визначення маршрутів і комутації;
- відстеження неполадок мережі;
- напрямок пакетів в потрібну мережу;
- присвоєння унікальної адреси мережевого вузла;
- просування даних.

Вищеописані завдання реалізуються за допомогою протоколів і маршрутизаторів.

Маршрутизатори мають кілька мережевих інтерфейсів для з'єднання мереж. Вони реалізуються програмно на базі універсального комп'ютера, а також на базі спеціалізованих платформ.

Протоколи цього рівня tcp/ip діляться на два класи: зі встановленням з'єднання і без встановлення з'єднання. Перші встановлюють маршрут передачі даних від джерела до одержувача, перш ніж відправити пакет, після чого розривають зв'язок. Друга група відправляє дані, які містять адресу відправника і одержувача в кожному пакеті. Проміжні мережеві пристрої, ґрунтуючись на цих даних, визначають маршрут прямування. Так пакет даних буде передаватися

від одного пристрою до іншого, поки не дійде до одержувача. Такі протоколи не можуть гарантувати, що інформація дійде до одержувача в тому вигляді, в якому вона була відправлена адресатом. У разі, якщо пакети підуть різними маршрутами, транспортні протоколи почнуть відновлення порядку даних.

Наступний рівень (рівень II, можна умовно його назвати "Транспортний") називається основним. Під час передачі даних може статися втрата пакета або спотворення інформації. Ряд додатків має власні засоби обробки помилок, інші - вважають за краще взаємодіяти тільки через безпечне з'єднання.

Протоколи tcp/ip цього рівня забезпечують надійність інформаційного зв'язку між двома кінцевими вузлами. На них покладені такі функції [25]:

TCP - регулювання навантаження на мережу, скорочення очікування отримання даних, передача даних в певній послідовності, відновлення інформації, втраченої в процесі передачі.

UDP - забезпечення надійного зв'язку у випадках, коли обсяг переданих даних невеликий. На відміну від TCP використовується для передачі даних на кілька комп'ютерів.

Виділяють 5 класів транспортного сервісу, які відрізняються за такими параметрами: терміновість, можливість відновлення перерваного з'єднання, наявність засобів мультиплексування, виявлення спотворень і інших помилок, пошук втрачених дублів.

Вибір того чи іншого класу залежить від того, як сам додаток вирішує завдання надійності і від того, наскільки надійна сама передача даних. Припустимо, що якість каналів зв'язку висока і ймовірність появи помилок мінімальна, тоді немає необхідності використовувати найбільш розвинений сервіс цього рівня [28].

Верхній рівень (рівень I, можна умовно його назвати "Прикладний"). Являє собою набір різних протоколів, які дозволяють користувачам мережі використовувати файли, принтери або гіпертекстові веб-сторінки, здійснювати спільну роботу. На цьому рівні мережі tcp/ip працює більшість мережевих додатків.

Виділяють багато різних протоколів, найбільш відомі серед них:

- HTTP - для роботи з інтернет браузерами;
- FTP - для передачі файлів, завантаження сторінок;
- SMTP - надсилання та отримання пошти;
- SNMP - протокол управління мережевими пристроями;
- RIP - протокол збору маршрутної інформації.

3.2 Обґрунтування вибору мови програмування

Стандартна мова розмітки HTML дозволяє легко створювати статичні Web-сторінки. Користувач не може змінювати їх вміст, не може взаємодіяти з ними. Для того щоб зробити сторінку справжньому інтерактивною, потрібен ще одну мову, що виконується в контексті браузера, - скриптова мова.

Дослідження роботи додатків інтернету показали, що для виконання певних дій користувача немає необхідності постійно звертатися до сервера - ці дії можна реалізувати на стороні клієнта, якби він дозволяв якимось чином їх запрограмувати. Так з'явився вбудований в програму перегляду Web-сторінок (браузер) мову JavaScript, який розширив можливості мови розмітки HTML, надаючи розробнику можливість вбудовувати в документ HTML код програми, що виконується на клієнті [25].

Скриптова мова використовується для створення інтерактивних сторінок. Зазвичай він не містить всіх можливостей справжніх мов програмування, таких, наприклад, як робота з файлами або управління графікою. Створені за допомогою скриптових мов програми не можуть виконуватися самостійно - вони працюють тільки в контексті браузера, підтримує виконання скриптових програм. створювані на скриптових мовах програми, називаються сценаріями або скриптами, включаються до складу Web-сторінок і розпізнаються і обробляються браузером окремо від решти HTML - коду.

Об'єктно-орієнтована скриптова мова програмування JavaScript (спочатку названа LiveScript його творцем, Бренданом Ваше, і розгорнута в складі браузера Netscape Navigator) була вперше представлена публіці в 1995 році [26].

Мова програмування JavaScript - об'єктно-орієнтована мова розробки вбудованих додатків, що виконуються як на стороні клієнта, так і на стороні сервера.

Для додавання JavaScript-коду на сторінку, можна використовувати теги `<script>` `</script>`.

Область застосування JavaScript дуже широка:

- в клієнтській частини веб-додатків;
- в AJAX;
- в технології Comet;
- в браузерних операційних системах;
- для створення невеликих програм, що розміщуються в закладки браузера (Букмарклет);
- додатки, написані на JavaScript, можуть виконуватися на серверах, що використовують Java 6 і пізніших версій, що використовується для побудови серверних додатків, що дозволяють обробляти JavaScript на стороні сервера;
- в якості мови розробки мобільних додатків (на платформі Mojo SDK в Palm webOS);
- для реалізації віджетів, так і для реалізації движків віджетів (Apple_Dashboard, Microsoft Gadgets, Google Desktop Gadgets, Klipfolio Dashboard);
- для написання прикладного ПО (57% вихідного коду Mozilla Firefox написано на JavaScript);
- як скриптової мови доступу до об'єктів додатків;
- в офісних додатках для автоматизації рутинних дій, написання макросів, організації доступу з боку веб-служб;
- в Excel Services 2010 Додати два нових інтерфейсу програмування додатків: REST API і JavaScript Object Model (JSOM).

Для забезпечення високого рівня абстракції і досягнення прийнятної ступеня крос-браузерності при розробці веб-додатків використовуються бібліотеки JavaScript. Вони являють собою набір багаторазово використовуваних об'єктів і функцій. Серед відомих JavaScript бібліотек можна відзначити Adobe life, Dojo Toolkit, Extjs, jQuery, Mootools, Prototype, Qooxdoo.

Netscape винайшла JavaScript, і JavaScript був вперше використаний в браузерах фірми Netscape. Але Netscape працює спільно з ECMA (European Computer Manufacturers Association / Європейська Асоціація Виробників Комп'ютерів) для створення стандартизованого міжнародного мови програмування на базі ядра JavaScript. ECMA є асоціацією міжнародних стандартів в галузі інформації і систем комунікації. Стандартизована версія JavaScript, звана «ECMAScript», поводитьсь абсолютно однаково в усіх додатках, що підтримують цей стандарт. Компанії можуть використовувати цей відкритий стандартна мова для розробки своїх реалізацій JavaScript. Перша версія стандарту ECMA задокументована в специфікації ECMA-262 [27].

Таблиця 3.2 – Версії Javasscript та ECMA

Версія JavaScript	Співвідношення з версією ECMA
JavaScript 1.1	ECMA-262, Edition 1 ґрунтується на JavaScript 1.1.
JavaScript 1.2	<p>ECMA-262 не була завершена, когда з'явилась версія JavaScript 1.2.</p> <p>JavaScript 1.2 не повністю сумісна з ECMA-262, Edition 1 з наступних причин:</p> <ul style="list-style-type: none"> • Netscape розробила в JavaScript 1.2 додаткові можливості, які не увійшли в ECMA-262. • ECMA-262 добавила дві нові можливості: інтернаціоналізація через Unicode та універсальна поведінка на всіх платформах. Деякі можливості JavaScript 1.2, такі как об'єкт Date, були платформозалежними і поводити себе по різному на різних платформах.
JavaScript 1.3	<p>JavaScript 1.3 повністю сумісна з ECMA-262, Edition 1.</p> <p>JavaScript 1.3 усунув невідповідність між JavaScript 1.2 и ECMA-262, зберігши всі додаткові можливості JavaScript 1.2, за винятком == та !=, які були змінені для забезпечення відповідності з ECMA-262.</p>
JavaScript 1.4	<p>JavaScript 1.4 повністю сумісний з ECMA-262, Edition 1.</p> <p>Третя версія специфікації ECMA була не закінчена, коли JavaScript 1.4 з'явився на світ.</p>
JavaScript 1.5	JavaScript 1.5 повністю сумісний з ECMA-262, Edition 3.

3.3 Обґрунтування вибору фреймворку для програмної реалізації

Розробка якісного веб-додатка залишається непростим завданням. Найчастіше це пов'язано з тим, що розробники використовують застарілі системи розробки, від яких не можуть відмовитися, тому що вони є звичними і надійними. Технологія SPA [29] дозволяє використовувати сучасні засоби розробки веб-сайтів.

SPA (Single Page Application) - це веб-додаток або веб-сайт, в якому використовується тільки один HTML-документ в якості оболонки для всіх веб-сторінок. Цей HTML-документ організовує взаємодію з користувачем через динамічно завантажувані HTML-коди, CSS-стилі і JavaScript-код за допомогою технології AJAX. В результаті при оновленні даних веб-сторінка не перезавантажується повністю, і, як наслідок, веб-програми працюють швидше і стають зручніше [27].

На даний момент кілька великих компаній надають різні засоби і технології в області SPA. До числа найбільших і популярних відносяться компанії Google і Facebook, які є розробниками таких SPA-продуктів, як Angular і React.

React - це JavaScript-бібліотека з відкритим вихідним кодом для розробки призначених для користувача інтерфейсів. Дана бібліотека розробляється і підтримується компанією Facebook і співтовариством окремих розробників і корпорацій. Бібліотека React може бути успішно використана для розробки односторінкових і мобільних додатків.

Бібліотека React створена розробником програмного забезпечення Джорданом Валку і вперше була згадана в новинній стрічці Facebook в 2011 році. Вихідний код був представлений в 2013 році на конференції «JSConf US» [30].

В основу розробки бібліотеки покладено компонентний підхід, що дозволяє створювати інкапсульовані компоненти, які мають власний стан, і об'єднувати ці компоненти в складні інтерфейси. Оскільки логіка самих компонентів реалізована на мові програмування JavaScript, можна передавати дані по всьому додатком, не прив'язуючись до інтерфейсу DOM.

Однією з ключових особливостей бібліотеки є можливість використання препроцесора JSX, з синтаксисом, схожим з мовою розмітки HTML, який перетворюється в код мови програмування JavaScript. Препроцесор JSX дозволяє описувати структуру інтерфейсу. Як правило, при використанні бібліотеки React, для побудови компонентів використовують саме цей препроцесор, але можна використовувати так само і чистий JavaScript код.

До особливостей бібліотеки також відносяться можливість використання односпрямованої передачі даних. Властивості можуть передаватися від батьківських компонентів до дочірнім. Таким чином, компоненти отримують властивості як безлічі постійних значень, завдяки цьому, виключається можливість змінювати властивості компонента при передачі даних, але залишається можливість зміни властивостей через функції зворотного виклику. Такий механізм називають «властивості вниз, події наверх» [30].

Бібліотека React працює з використанням віртуального інтерфейсу DOM (Document Object Model). У пам'яті створюється кеш-структура, яка дозволяє обчислювати різниця попереднього і поточного станів інтерфейсу. Таким чином, у розробника з'являється можливість працювати зі сторінкою, в припущенні, що вона повністю оновлюється, але які компоненти оновлювати, вирішує сама бібліотека. У свою чергу, компоненти мають життєвий цикл, для кожного циклу реалізовані свої методи, які дозволяють розробникам запускати код на різних стадіях життєвого циклу компонента. Наведемо приклад:

- метод `shouldComponentUpdate` дозволяє скасувати перерисовку компонента, якщо вона не потрібна;
- метод `componentDidMount`;
- метод `render`

Головною метою бібліотеки є надання високої швидкості доступу, простоти і масштабованості. Також вона дозволяє використовувати інші бібліотеки для розробки призначених для користувача інтерфейсів, такі, як бібліотека для управління станом додатки Redux або бібліотеки для маніпулювання даними GraphQL.

Angular - це платформа для розробки веб-додатків, написана мовою програмування TypeScript. Розроблено та підтримується командою з компанії Google, а також спільнотою розробників з різних компаній. Створювалася як нова версія фреймворку AngularJS, однак, нова версія була повністю переписана мовою програмування TypeScript і володіє іншою архітектурою, через що не має зворотної сумісності з попередньою версією [33].

Платформа не тільки надає інструменти, але і дизайни шаблону, що дозволяє створювати обслуговується і розширюваний проект.

Правильне дотримання архітектури додатку, яка надається при створенні, дозволяє уникнути плутанини в класах і методах [32].

Платформа Angular використовує розмітку HTML для визначення призначеного для користувача інтерфейсу додатку. Розмітка HTML є більш інтуїтивним і менш складним засобом, ніж процедурне визначення інтерфейсу в мові програмування JavaScript, що дозволяє залучити більше розробників.

Маніпуляція з об'єктами інтерфейсу DOM для представлення даних також є частиною поведінки платформи. Код маніпулювання інтерфейсом DOM повинен бути всередині директив, а не в поданні. Це значною мірою знижує, зокрема, через те, що платформа бачить уявлення як сторінку з кодом HTML з заповненими даними. Абстрагуючись від маніпуляцій інтерфейсу DOM, розробники можуть зосередитися на поданні призначеного для користувача інтерфейсу без цих відволікаючих чинників.

У платформі Angular реалізовані контролери - прості функції, які мають тільки одне завдання - маніпулювати областю дії. Наприклад, з'являється можливість використовувати платформу, щоб попередньо заповнити дані в області з сервера або реалізувати перевірку бізнес-логіки. На відміну від інших середовищ, контролери не є об'єктами і не є нащадками інших об'єктів.

Модульне тестування, реалізоване в платформі, є її винятковою особливістю. Модульне тестування істотно перевершує традиційний спосіб, тому що дозволяє створити окрему тестову сторінку, в якій перевіряє працездатність компонента [30].

Платформа Angular і бібліотека React є безкоштовними, і обидві користуються популярністю серед програмістів в області розробки веб-сайтів і веб-додатків. Розглянемо переваги і недоліки кожної з платформ.

- При розробці веб-сайту або веб-додатки найнеобхіднішим властивістю використовуваного інструменту є його самодостатність. Бібліотека React є основою для розробки користувальницького інтерфейсу, але потребує додаткових бібліотек, в той час як платформа Angular є повноцінним фреймворком, який зазвичай не вимагає додаткових бібліотек.

- Поріг входження є головним фактором при виборі інструменту у початківців розробників. Бібліотека React є досить простий у використанні: немає впровадження залежностей, немає класичних шаблонів, немає занадто складних функцій. Бібліотека досить проста для освоєння, якщо розробник вже знайомий з мовою програмування JavaScript.

- Документація і підтримка також сильно впливає на вибір інструменту у розробників. Працюючи з бібліотекою React, ви повинні бути постійним учнем, так як структура часто оновлюється. У той час, як спільнота розробників намагається якомога швидше просувати останню документацію, йти в ногу з усіма змінами не так просто. Платформа Angular викликає великий скептицизм, частково через непопулярність першої версії платформи Angular 1.0. Розробники звикли відхиляти фреймворк як занадто складний, так як на його вивчення йшло багато часу. Проте, ця структура розроблена компанією Google, що говорить на користь вибору платформи Angular. [32]

- Структура бібліотеки React надає розробникам свободу вибору. Не існує «єдино правильної структури» для програми, розробленої за допомогою бібліотеки React. У той же час, структура платформи Angular є фіксованою і складною, більш придатною для досвідчених розробників.

- Найбільша перевага платформи Angular полягає в тому, що, на відміну від бібліотеки React, він підтримує впровадження залежностей. Тому платформа Angular дозволяє мати різні життєві цикли для різних компонентів.

- Бібліотека React використовує віртуальну об'єктну модель документів, яка дозволяє легко реалізовувати незначні зміни даних в одному елементі без оновлення структури всього дерева. У той час, як платформа Angular використовує реальну об'єктну модель документів та інкрементальне дерево, що звільняє від необхідності використовувати велику кількість пам'яті для зберігання копії дерева.

Розробка веб-сайтів і веб-додатків за допомогою технології SPA на сьогоднішній день дуже популярна і стрімко розвивається, що мотивує розробників створювати і розвивати інструменти, які працюють на технології SPA. Бібліотека React і платформа Angular користуються великою популярністю серед розробників. Вони обидві гарні в області SPA, і у кожної є відмінні риси. Але для програмної реалізації інформаційної технології доцільніше використати Angular через його самодостатність.

3.4 Розробка структури інформаційної технології

В першу чергу потрібно звернути увагу на те, що побудова інформаційної системи являє собою комплекс окремих секторів – модулів (підсистем). Модуль – це компонент цілого, що визначається за певною властивістю. Саме тому структура будь-якої інформаційної системи (ІС) може бути представленою сукупністю підсистем [34].

Спроекуємо інформаційну технологію прийняття рішень в умовах критеріальної невизначеності відповідно до задач, які повинна вирішувати дана інформаційна технологія. Розроблену архітектуру інформаційної технології прийняття рішень в умовах критеріальної невизначеності представлено на рисунку 3.2.

Вхідні дані користувачів представлені записами у базі даних.

Далі модуль порівняння альтернатив використовує ці дані для формування результату. Модуль містить в собі підмодуль оцінки альтернатив окремим

експертом та модуль виведення результату з урахуванням коефіцієнтів компетентності експертів.

Дана інформаційна технологія надає дані у форматі API для формування відсортованого списку альтернатив. Для цього створений серверний модуль, який взаємодіє з модулем формування відфільтрованого списку альтернатив. Також серверний модуль відповідає за валідацію вхідних значень від клієнтів інформаційної технології [35].

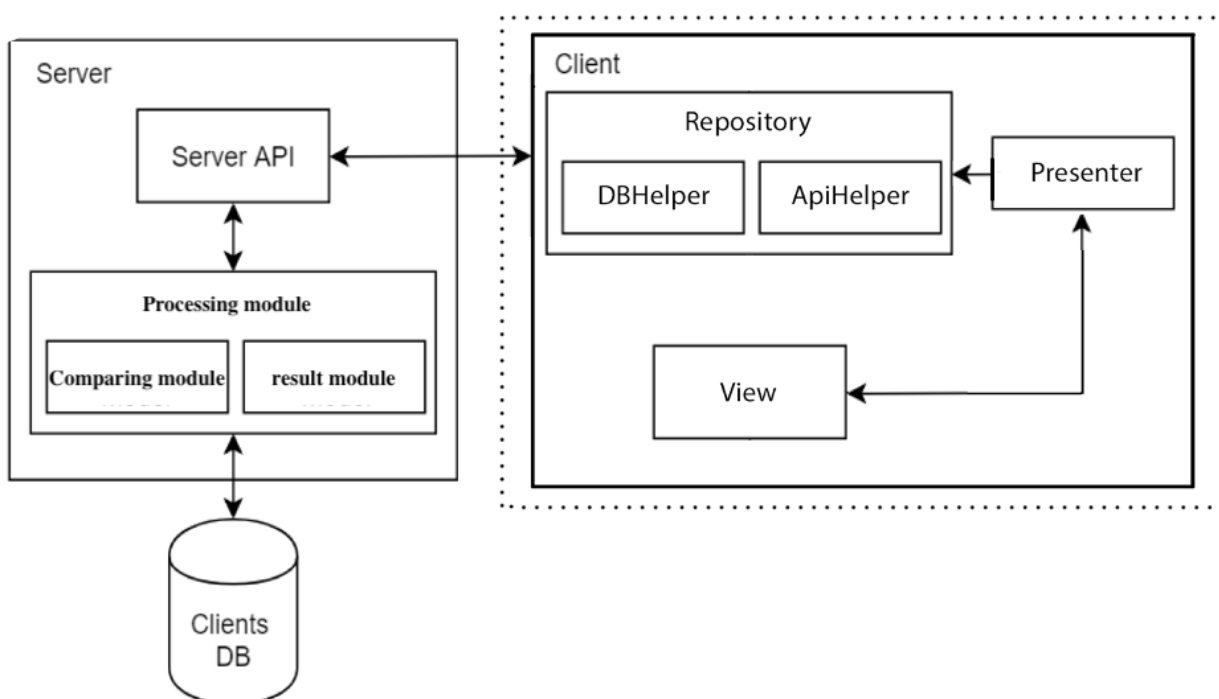


Рисунок 3.2 – Архітектура інформаційної технології прогнозування відтоку клієнтів

Клієнт інформаційної технології представлений додатком Angular. Надає інтерфейс для вводу інформації про експертів та альтернативи, забезпечує взаємодію з сервером за допомогою HTTP запитів.

3.5 Розробка алгоритму роботи інформаційної технології

Розглянемо алгоритм функціонування програмного модуля від його включення до моменту завершення його роботи доцільно розбити на два підмодулі: формування списку альтернатив і експертів та виведення результату урахуванням коефіцієнтів компетентності експертів. Відповідно кожен з даних підмодулів відповідає за свою область та обмінюється даними з іншим [36, 37].

Спершу розглянемо алгоритм роботи підмодуля формування списку альтернатив та списку експертів і визначення їх компетентності.

1. Початок роботи алгоритму
2. Авторизація ОПР
3. Перевірка чи натиснуто кнопку «Додати альтернативу». Якщо натиснуто, то до списку альтернатив, які потрібно порівняти додається ще одна.
4. Перевірка чи натиснуто кнопку «Далі». Якщо натиснуто, то відбувається перехід до формування списку експертів.
5. Перевірка чи натиснуто кнопку «Додати експерта». Якщо натиснуто, то до списку експертів, компетентність яких потрібно оцінити додається ще один.
6. Перевірка чи натиснуто кнопку «Далі». Якщо натиснуто, то відбувається перехід до очікування результату.
7. Перевірка чи натиснуто кнопку «Результат». Якщо натиснуто, то відбувається виведення результатів роботи.
8. Кінець роботи алгоритму

На рисунках 3.3 та 3.4 показані схеми алгоритмів роботи цих підмодулів. Щодо підмодуля порівняння альтернатив, визначення компетентності експертів та виведення результатів, то він має наступний вигляд.

1. Початок роботи алгоритму.
2. Отримання списку всіх експертів, які може порівнювати користувач.
3. Визначення компетентності експертів.
4. Отримання списку альтернатив, які вніс ОПР.

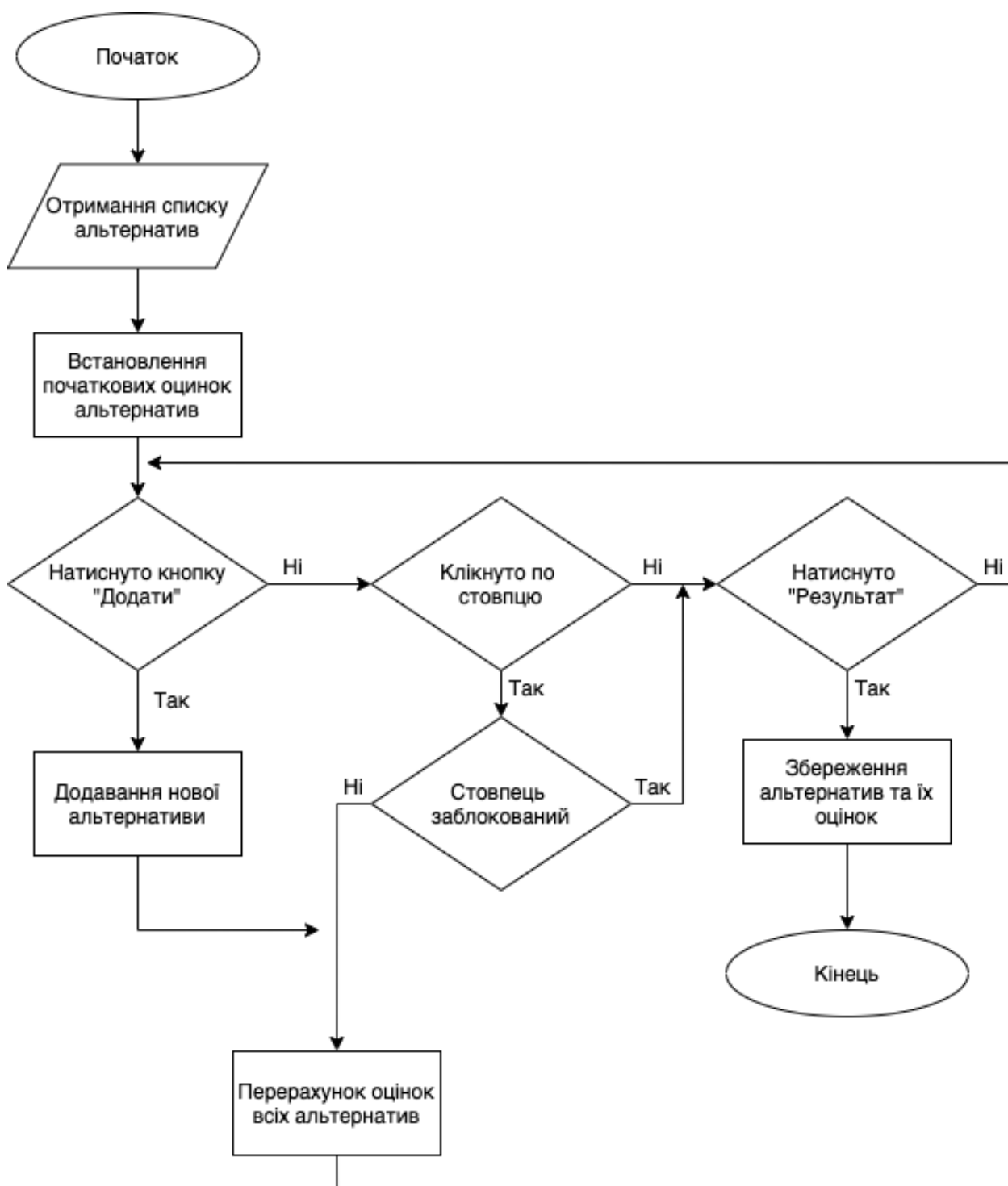


Рисунок 3.3 – Алгоритм інформаційної технології прийняття рішень в умовах критеріальної невизначеності

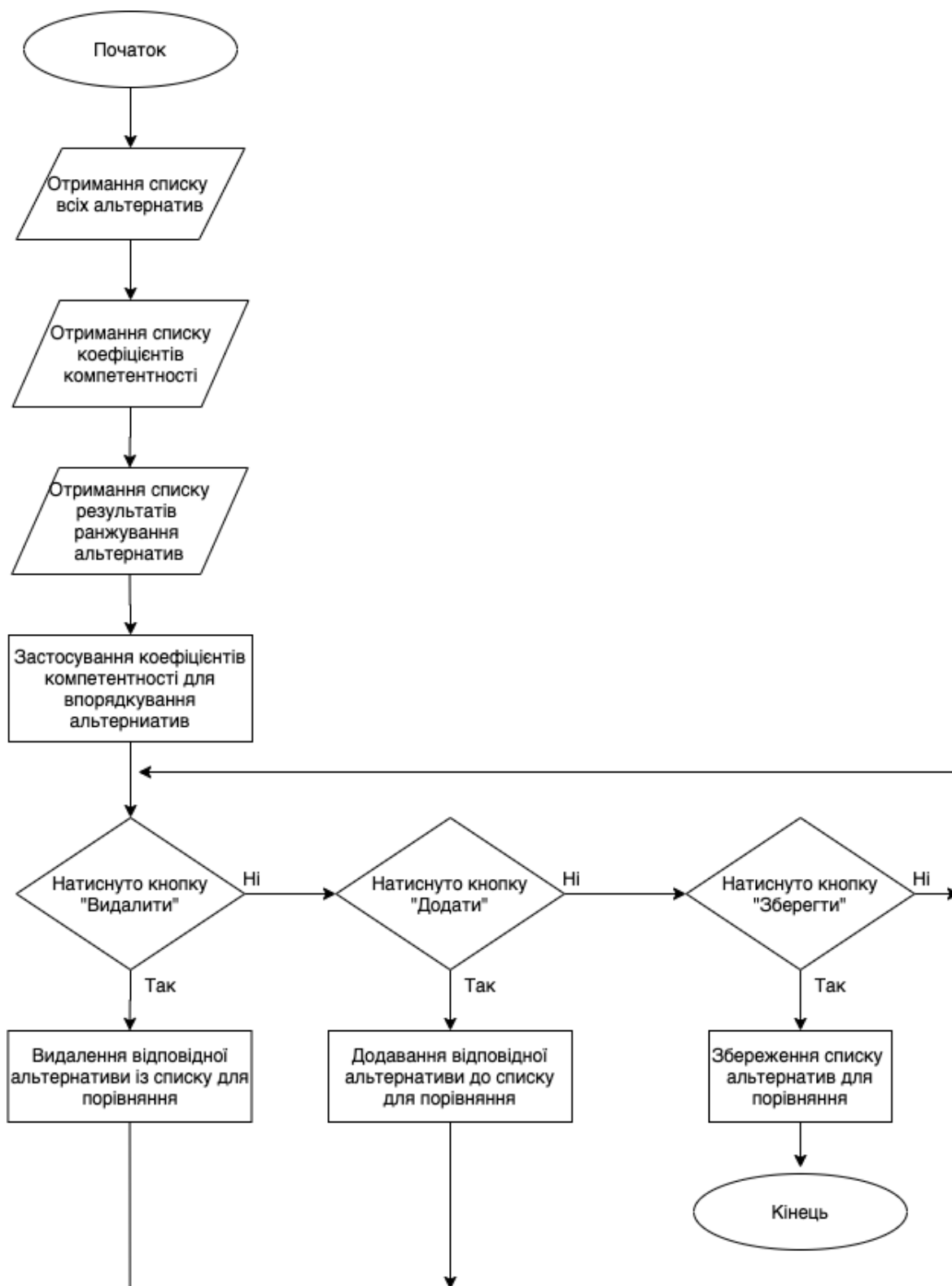


Рисунок 3.4 – Алгоритм роботи підмодуля формування результату виконання

5. Порівняння альтернатив.
6. Застосування коефіцієнтів компетентності експертів для визначення с поміж всіх можливих альтернатив тих, які можливо цікаві користувачу використовуючи данні отримані від попередніх користувачів.
7. Перевірка чи натиснуто кнопку «Видалити». Якщо натиснуто, то із списку запропонованих користувачу альтернатив видаляється та, на кнопку якої натиснуто.
8. Перевірка чи натиснуто кнопку «Додати». Якщо натиснуто, то до списку рекомендованих альтернатив додається та, на кнопку якої натиснуто.
9. Перевірка чи натиснуто кнопку «Зберегти». Якщо натиснуто, то відбувається збереження відфільтрованого списку альтернатив.
10. Кінець роботи алгоритму.

3.6 Аналіз результатів тестування інформаційної технології

Прийняти рішення – це лише половина справи. Друга половина – оцінити, наскільки воно було правильним, вірним і ефективним. Важливо це з тієї причини, що оцінка дозволяє зрозуміти, наскільки грамотними були вжиті заходи, чи приведуть вони до успіху в майбутньому, і взагалі, чи варто на них розраховувати.

Кінцевим продуктом аналізу прийнятого рішення завжди є результат. На основі нього можна судити, чи досягнута мета, які були задіяні для її досягнення ресурси, скільки було витрачено сил і часу, що вийшло в результаті.

Для визначення ефективності прийнятих рішень було обрано коефіцієнт рівня досягнення цілей, що розраховується за формулою [38]:

$$K_{\text{дос}} = \frac{V_c \cdot R_{c1} + V_c \cdot R_{c2} + \dots + V_c \cdot R_{cn}}{n},$$

де n – кількість цілей, одиниці;

V_c – вагомість досягнутих цілей, частка;

R_c – рівень досягнення цілей, %.

Було проведено ряд експериментів з використанням програм-аналогів та розроблюваної технології. Результати дослідження коефіцієнта рівня досягнення цілей представлено в таблиці 3.3.

Таблиця 3.3 – Результати дослідження коефіцієнта рівня досягнення цілей

Програмний засіб	Вагомість досягнутих цілей	Рівні досягнутих цілей	Кількість цілей	Коефіцієнт рівня досягнення цілей
PriEsT	0,81	82%	3	60,94%
	0,67	83%		
	0,76	80%		
MindDecider	0,75	78%	3	50,58%
	0,63	74%		
	0,62	74%		
Розроблюваний	0,82	85%	3	66,27%
	0,77	86%		
	0,74	85%		

Отже, в результаті аналізу тестування програмних засобів PriEsT, MindDecider та розробленої інформаційної технології прийняття рішень в умовах критеріальної невизначеності виявлено, що, використання розробленої інформаційної технології критеріальної невизначеності збільшує ефективність прийняття рішень в середньому на 10.5%.

3.7 Висновок

В даному розділі було описано проектування структури програмного забезпечення та програмну реалізацію, включаючи обґрунтування вибору платформи та мови програмування, розробку алгоритмів роботи програми, тестування та аналіз результатів роботи програми.

Програмне забезпечення спроектовано в об'єктно-орієнтованій парадигмі. Обґрунтовано вибір фреймворку Angular, базуючись на таких основних перевагах: широкий функціонал, висока продуктивність розробки, велика кількість стандартних бібліотек та вільна ліцензія використання – BSD, вичерпна документація, переносимість.

Проведено вибір мови програмування. Для програмної реалізації обрано мову програмування JavaScript, оскільки саме на ній виконується розробка веб-додатків, вона має непогану швидкодію, реалізує парадигму ООП та має широкий вибір бібліотек для реалізації переносимого графічного інтерфейсу.

Розроблено алгоритм роботи програмного забезпечення. Алгоритми детально описано та наведено їхні схеми. Проведено тестування програмної реалізації та аналіз результатів роботи. Тестування показало, що розроблювана технологія покращує ефективність прийняття рішень в середньому на 10.5%.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Оцінювання комерційного потенціалу розробки

Метою проведення технологічного аудиту є оцінювання комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності [39].

Результатом виконання магістерської кваліфікаційної роботи є інформаційна технологія прийняття рішень в умовах критеріальної невизначеності. Для проведення технологічного аудиту залучено трьох незалежних експертів: Колодний Володимир Володимирович (к.т.н., доцент каф. комп'ютерних наук ВНТУ) Колесницький Олег Костянтинович (к.т.н., доцент каф. комп'ютерних наук ВНТУ), та Озеранський Сергій Володимирович (к.т.н., доцент каф. комп'ютерних наук ВНТУ).

Оцінювання комерційного потенціалу буде здійснене за критеріями, що наведені в таблиці 4.1.

Таблиця 4.1 - Критерії оцінювання комерційного потенціалу розробки
бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів

Продовження таблиці 4.1

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-тер.	0	1	2	3	4
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
5	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві

Продовження таблиці 4.1

11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання комерційного потенціалу експертами розробки зведено в таблицю 4.2.

Таблиця 4.2 - Результати оцінювання комерційного потенціалу розробки

Критерії	Прізвище, ініціали, посада експерта		
	1 – Колодний	2 – Колесницький	3 – Озеранський
	Бали, виставлені експертами:		
1	4	3	4
Ринкові переваги (недоліки):			
2	3	3	3
3	5	4	4
4	4	3	4
5	5	4	5
Ринкові перспективи			
5	4	3	3
7	3	4	3
Практична здійсненність			
8	4	3	4
9	3	3	3
10	5	5	5
11	5	4	5
12	4	5	5
Сума балів	СБ ₁ =40	СБ ₂ =37	СБ ₃ =39
Середньоарифметична сума балів $\overline{СБ}$	47		

За даними таблиці 4.2 можна зробити висновок, щодо рівня комерційного потенціалу розробки. Зважимо на результат й порівняємо його з рівнями комерційного потенціалу розробки, що представлено в таблиці 4.3.

Таблиця 4.3 – Рівні комерційного потенціалу розробки

Середньоарифметична сума балів $\overline{СБ}$, розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0 – 10	Низький
11 – 20	Ниже середнього
21 – 30	Середній
31 – 40	Вище середнього
41 – 48	Високий

З таблиці зрозуміло, що рівень комерційного потенціалу розробки становить 47 балів, така оцінка відповідає рівню «високий».

Нова розробка являє собою інформаційну технологію прийняття рішень в умовах критеріальної невизначеності, що дозволяє визначити, яка з альтернатив є кращою в контексті прийняття конкретного рішення.

Розглянемо основні системи-конкуренти, які теж дають змогу оцінювати та впорядковувати альтернативи при прийнятті рішень.

- MindDecider призначений для використання як в наукових дослідженнях, так і для прийняття рішень при управлінні проектами, окремими завданнями і управління часом. У своїх проектах користувач може використовувати вже підготовлені об'єкти з їх властивостями з різних предметних областей (хімічні елементи, фізичні постійні, математичні константи, акції фондових бірж та ін.) Або ж створювати свої. Серед переваг наявний функціонал аналізу ефективності роботи, основними недоліками можна назвати призначення лише до одної операційної системи Windows.

PriEsT реалізує процес аналітичної ієрархії (АНР), який широко використовувався в багатьох галузях, таких як охорона здоров'я, транспорт,

телекомунікації та прийняття політичних рішень. Два типи проблем, що розглядаються PriEsT, – це проблеми ранжування та бюджетування [6].

Завдяки відкритому коду, цей інструмент також підходить для дослідницької спільноти. Наприклад, комерційні інструменти не пропонують всіх доступних методів ранжування, і зазвичай приховують технічні деталі реалізації своїх продуктів. PriEsT навпаки має відкритий код, а також дозволяє вносити зміни.

У таблиці 4.4 наведені основні технічні показники аналога і розроблюваного програмного продукту

Таблиця 4.4 - Основні технічні показники аналога і нового програмного продукту

Показники (приклад)	Аналог	Нова розробка	Відношення параметрів нової розробки до параметрів аналога
Функціональність	Середня	Середня	1
Надійність	Висока	Висока	1
Сумісність	Висока	Середня	0,5
Економія ресурсів і часу	Середня	Висока	1,5
Простота використання	Середня	Висока	1,5

За функціональністю нова розробка перевищує аналог оскільки, аналог хоча і надає можливість порівнювати альтернативи, але поступається в точності ранжування, та не враховує індивідуальні характеристики кожного експерта.

З точки зору зручність використання продуктів аналогів – обидва продукти мають доволі зручний інтерфейс користувача, проте вони потребують додаткового часу для його опанування .

Комбінування різних алгоритмів та методів у аналога відсутнє. У нова розробка використовує в своїй основі комбінацію різних методів та алгоритмів оцінювання та впорядковування альтернатив.

Продуктивність даних програмних засобів знаходиться на майже однаковому рівні. Проте за рахунок застосування в процесі розробки великої кількості нових, сучасних методик розроблювана система допоможе приймати рішення набагато швидше та краще.

Реалізація інтформаційної технології прийняття рішень в умовах критеріальної невизначеності буде найефективнішою у вигляді тимчасової підписки на користування функціоналом нової розробки.

Нова розробка може використовуватися у будь якій сфері, яка передбачає оцінювання альтернатив при прийнятті рішень, а тому безпосереднім споживачем можна вважати менеджерів великих підприємств, та власників підприємств середнього та малого бізнесу.

Пропонований продукт повинен вирішити проблему високої вартості аналізу альтернатив при прийнятті рішень на українському ринку, шляхом доступної ціни на аналогічну послугу, та вищу зручність використання у порівнянні з аналогами.

Даний продукт буде реалізовуватися на Українському ринку у сфері підтримки прийняття рішень.

На даний продукт очікується високий попит, адже кожен власник компанії, для ефективного управління якою потрібно приймати рішення, бажає мати ефективний інструмент для прийняття рішень.

Основним напрямом подальшого впровадження є впровадження функціоналу роботи інтелектуальної системи аналізу альтернатив, та створення можливості використання даної системи без залучення експертів.

Програмний продукт планується реалізовувати шляхом ліцензійної угоди. Розповсюджуватиметься по типу Trial – умовно-безкоштовне програмне забезпечення з повним функціоналом, але з часом використання 30 днів, але з подальшою підпискою на місяць, рік, чи без терміну дії. Програму не потрібно буде інсталювати на комп'ютер чи смартфон. Користувачу достатньо буде перейти на сайт програмного продукту.

Розробка потребує залучення незначних фінансових ресурсів, а більшість матеріалів для її реалізації відомі і давно використовуються. На ринку праці є велика кількість фахівців в області веб-розробки, що спростить задачу пошуку розробників, регламентні обмеження відсутні і немає необхідності отримання дозвільних документів.

З метою комерціалізації розробки необхідно провести пошук потенційних партнерів та інвесторів.

4.2 Прогнозування витрат на виконання наукової роботи та впровадження результатів

Здійснимо прогнозування витрат на виконання науково-дослідної, дослідно-конструкторської та конструкторсько-технологічної роботи для розробки програмного забезпечення.

Виконаємо розрахунок витрат з урахуванням того, що розробкою займався один розробник програмного забезпечення.

Основна заробітна розробника-дослідника Z_o :

$$Z_o = \frac{M}{T_p} \cdot t \text{ [грн]}, \quad (4.1)$$

де M – місячний посадовий оклад конкретного розробника;

T_p – кількість робочих днів в місяці; $T_p = 22$ дні;

t – кількість днів роботи розробника.

Розрахунки заробітних плат для керівника і програміста наведені в таблиці 4.5.

Таблиця 4.5 – Розрахунки основної заробітної плати

Працівник	Оклад М, грн.	Оплата за робочий день, грн.	Число днів роботи, t	Витрати на оплату праці, грн.
Науковий керівник	12000	545,45	10	5454,5
Інженер-програміст	10000	454,54	50	22727
Всього				28181,5

Додаткова заробітна плата Z_d розробника розраховується як 12% від основної заробітної плати:

$$Z_d = 0,12 \cdot 28181,5 = 3381,78 \text{ (грн.)}$$

Нарахування на заробітну плату $H_{зп}$ розробника становить:

$$H_{зп} = (Z_o + Z_d) \cdot \frac{\beta}{100} \text{ [грн]}, \quad (4.2)$$

де Z_o – основна заробітна плата розробника;

Z_d – додаткова заробітна плата розробника;

β – ставка єдиного внеску на загальнообов'язкове державне соціальне страхування – 22%.

$$H_{зп} = (28181,5 + 3381,78) \cdot 0,22 = 6943,92 \text{ (грн.)}$$

У спрощеному вигляді амортизаційні відрахування розраховуємо за формулою:

$$A = \frac{Ц \cdot H_a}{100} * \frac{T}{12}, \quad (4.3)$$

де $Ц$ – балансова вартість обладнання, грн;

H_a – річна норма амортизаційних відрахувань % (для програмного

забезпечення 25%);

T – Термін використання (T = 3 міс.).

Таблиця 4.6 – Розрахунок амортизаційних відрахувань

Найменування програмного забезпечення	Балансова вартість, грн.	Норма амортизації, %	Термін використання, міс.	Величина амортизаційних відрахувань, грн
Персональний комп'ютер	20000	25	2	833,33
Всього				833,33

Матеріали, що використовуються при розробці продукту внесено до таблиці 4.7.

Таблиця 4.7 – Матеріали, що використовуються при виготовленні даного продукту

Найменування матеріалу	Ціна за одиницю, грн.	Витрачено, шт.	Вартість витраченого матеріалу, грн
Папір (упаковка)	120,00	1	120,00
Канцтовари	6,00	2	12,00
Карта пам'яті	300	1	300,00
Всього			432,00

Витрати на електроенергію V_e розраховуються за допомогою формули:

$$V_e = V \cdot P \cdot \Phi \cdot K_p \text{ [грн]}, \quad (4.4)$$

де V – вартість 1кВт електроенергії;

P – установлена потужність обладнання, кВт;

Φ – фактична кількість годин роботи комп'ютера при створенні програмного продукту, годин;

$K_{\text{п}}$ – коефіцієнт використання потужності.

Отже, витрати на енергію становлять:

$$B_e = 1,68 \cdot 0,6 \cdot 480 \cdot 0,6 = 290,30 \text{ (грн)}.$$

Витрати за доступ до Інтернет можна розрахувати за формулою:

$$B_{\text{ді}} = C_{\text{ді}} \cdot T \text{ [грн]}, \quad (4.5)$$

де $C_{\text{ді}}$ – це ціна доступу за місяць;

T – кількість місяців використання доступу до мережі.

$$B_{\text{ді}} = 200 \cdot 2 = 400,00 \text{ (грн)}.$$

Інші витрати $B_{\text{ін}}$ охоплюють: витрати на управління організацією, оплату службових відряджень, витрати на утримання, ремонт та експлуатацію основних засобів, витрати на опалення, освітлення, водопостачання, охорону праці тощо. Інші витрати $I_{\text{в}}$ можна прийняти як 100% від суми основної заробітної плати розробників:

$$B_{\text{ін}} = 1 \cdot 28181,50 = 28181,50 \text{ (грн)}.$$

Сума всіх попередніх статей витрат зумовлює витрати на виконання даної частини (розділу, етапу) роботи – B .

$$B = Z_o + Z_d + H_{\text{зп}} + A + B_{\text{мат}} + B_e + B_{\text{ін}} \text{ [грн]},$$

$$B = 28181,50 + 3381,78 + 6943,92 + 833,33 + 432 + 290,30 + 390 + 28181,5 = 68634,33 \text{ (грн)}.$$

2-й етап. Проведемо розрахунок загальних витрат на виконання та впровадження результатів даної роботи. Загальна вартість всієї наукової роботи визначається за $V_{\text{заг}}$ формулою:

$$V_{\text{заг}} = \frac{V}{\alpha} \text{ [грн]}, \quad (4.6)$$

де α – частка витрат, які безпосередньо здійснює виконавець даного етапу роботи, у відн. одиницях.

Оскільки у роботі було задіяно лише одну людину, якою виконується вся робота, то α становить 1. Підставивши дані у формулу, отримуємо:

$$V_{\text{заг}} = 68634,33 \text{ (грн)}.$$

3-й етап. Прогнозування загальних витрат на виконання та впровадження результатів виконаної роботи. Прогнозування загальних витрат ЗВ на виконання та впровадження результатів виконаної роботи здійснюється за формулою:

$$\text{ЗВ} = \frac{V_{\text{заг}}}{\beta} \text{ [грн]}, \quad (4.7)$$

де β – коефіцієнт, який характеризує етап (стадію) виконання даної роботи.

Отож, якщо розробка знаходиться:

- на стадії науково-дослідних робіт, то $\beta \approx 0,1$;
- на стадії технічного проектування, то $\beta \approx 0,2$;
- на стадії розробки конструкторської документації, то $\beta \approx 0,3$;
- на стадії розробки технологій, то $\beta \approx 0,4$;
- на стадії розробки дослідного зразка, то $\beta \approx 0,5$;
- на стадії розробки промислового зразка, $\beta \approx 0,7$;
- на стадії впровадження, то $\beta \approx 0,9$.

Отже, підставимо дані в формулу й отримаємо результат:

$$ЗВ = \frac{68634,33}{0,9} \approx 76260,00 \text{ (грн).}$$

4.3 Прогнозування комерційних ефектів від реалізації результатів розробки

В умовах ринку узагальнюючим позитивним результатом, що його отримує підприємство від впровадження результатів тієї чи іншої розробки, є збільшення чистого прибутку підприємства. Зростання чистого прибутку забезпечить підприємству надходження додаткових коштів, які дозволять покращити фінансові результати діяльності.

Виконання даної наукової роботи та впровадження її результатів складає приблизно 1 рік. Позитивні результати від впровадження розробки очікуються вже в перші місяці після впровадження.

Проведемо детальніше прогнозування позитивних результатів та кількісне їх оцінювання по роках.

Обчислимо збільшення чистого прибутку підприємства $\Delta\Pi_i$ для кожного із років, протягом яких очікується отримання позитивних результатів від впровадження розробки, розраховується за формулою:

$$\Delta\Pi_i = \sum_1^n (\Delta\Pi_{\text{я}} \cdot N + \Pi_{\text{я}} \cdot \Delta N)_i \text{ [грн]}, \quad (4.8)$$

де $\Delta\Pi_{\text{я}}$ – покращення основного якісного показника від впровадження результатів розробки у даному році;

N – основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

ΔN – покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки;

$П_я$ – основний якісний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

n – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки.

Припустимо, що внаслідок впровадження результатів наукової розробки чистий прибуток підприємства збільшиться на 100 грн., а кількість одиниць реалізованої послуги збільшиться: протягом першого року – на 300 од., протягом другого року – ще на 200 од., протягом третього року – ще на 100 од.

Орієнтовно: реалізація продукції до впровадження результатів наукової розробки складала 1 шт., а прибуток, який отримувало підприємство на одиницю продукції до впровадження результатів наукової розробки був 500 грн.

Спрогнозуємо збільшення чистого прибутку підприємства від впровадження результатів наукової розробки у кожному році відносно базового прибутку.

Збільшення чистого прибутку підприємства $\Delta П_1$ протягом першого року складе:

$$\Delta П_1 = 500 \cdot 1 + (500 + 100) \cdot 300 = 180500,00 \text{ (грн).}$$

Збільшення чистого прибутку підприємства $\Delta П_2$ протягом другого року:

$$\Delta П_2 = 500 \cdot 1 + (500 + 100) \cdot (300 + 200) = 300500,00 \text{ (грн)}$$

Збільшення чистого прибутку підприємства $\Delta П_3$ протягом третього року становитиме:

$$\Delta П_3 = 500 \cdot 1 + (500 + 100) \cdot (300 + 200 + 100) = 360500,00 \text{ (грн).}$$

Отже, розрахунки показують, що комерційний ефект від впровадження розробки виражається у значному збільшенні чистого прибутку підприємства.

4.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності

Щоб оцінити доцільність фінансування проекту, необхідно провести розрахунки ефективності вкладених інвестицій. Основними показниками є абсолютна і відносна ефективність вкладених інвестицій та термін їх окупності.

На першому етапі розрахуємо теперішню вартість інвестицій PV , що вкладаються в наукову розробку. Такою вартістю ми можемо вважати прогнозовану величину загальних витрат ZB на виконання та впровадження результатів науково дослідної роботи. Тобто будемо вважати, що $ZB = PV = 76260,00$ грн.

На другому етапі розраховуємо очікуване збільшення прибутку $\Delta\Pi_i$, яке отримає підприємець від впровадження результатів наукової розробки, для кожного із років, починаючи з першого року впровадження. Таке збільшення прибутку було розраховане раніше. Результати вкладених у наукову розробку інвестицій можна побачити по тому, що у першому році підприємство отримає збільшення чистого прибутку на 180500,00 грн. відносно базового року, у другому році – збільшення чистого прибутку на 300500,00 грн (відносно базового року), у третьому році – збільшення чистого прибутку на 360500,00 грн (відносно базового року).

На третьому етапі, для спрощення подальших розрахунків, будемо вісь часу, на яку наносимо всі платежі (інвестиції та прибутки), що мають місце під час виконання науково-дослідної роботи та впровадження її результатів.

Платежі показуються у ті терміни, коли вони здійснюються. Тоді рисунок, що характеризує рух платежів (інвестицій та додаткових прибутків) буде мати вигляд, наведений на рис. 4.1.

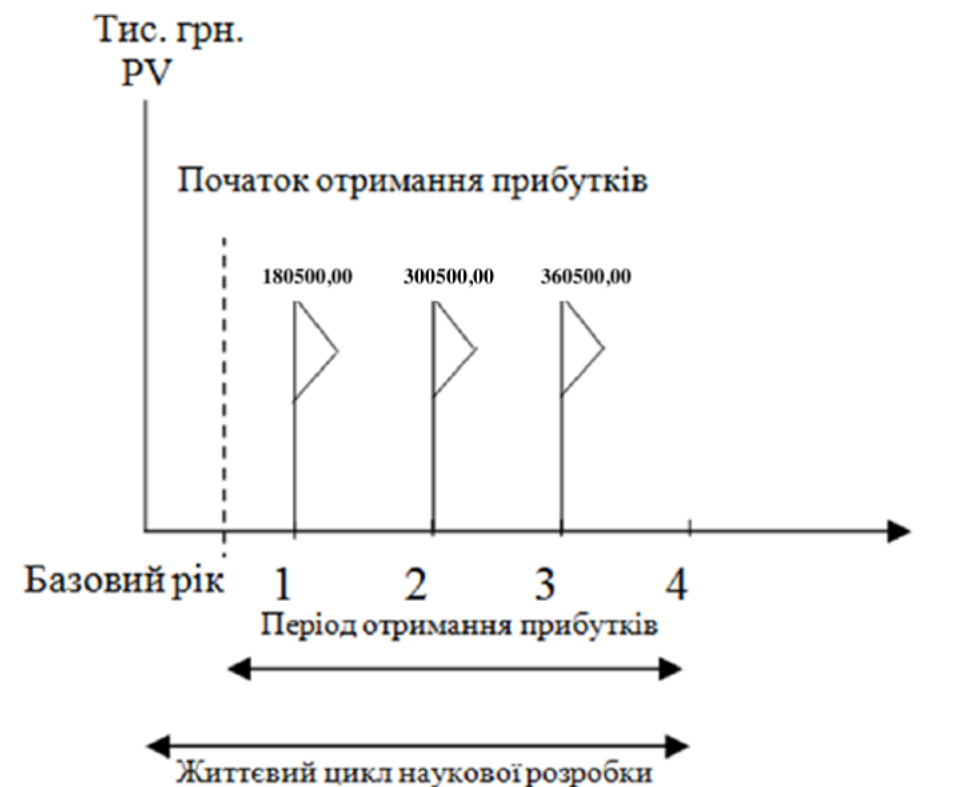


Рисунок 4.1 – Вісь часу з фіксацією платежів, що мають місце під час розробки та впровадження результатів НДДКР

На четвертому етапі розраховуємо абсолютну ефективність вкладених інвестицій E_{abc} за формулою:

$$E_{abc} = (ПП - PV) [\text{грн}], \quad (4.9)$$

де ПП – приведена вартість всіх чистих прибутків, що їх отримає підприємство від реалізації результатів наукової розробки, грн, розраховано за формулою 4.10;

PV – теперішня вартість інвестицій $PV = 3B = 76260,00$ (грн).

$$ПП = \sum_1^T \frac{\Delta\Pi_i}{(1+\tau)^t} [\text{грн}], \quad (4.10)$$

де $\Delta\Pi_i$ - збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДДКР, грн;

t – період часу, протягом якого виявляються результати впровадженої НДДКР, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні; для України цей показник знаходиться на рівні 0,1;

t – період часу (в роках) від моменту отримання чистого прибутку до точки „0”.

Проведемо розрахунки.

$$\text{ПП} = \frac{180500}{(1+0,1)^0} + \frac{300500}{(1+0,1)^2} + \frac{360500}{(1+0,1)^3} = 616236,97 \text{ (грн)}$$

Тоді ,

$$E_{abc} = 616236,97 - 76260,00 = 539976,97 \text{ (грн)}.$$

Оскільки $E_{abc} > 0$, то результат від проведення наукових досліджень та їх впровадження принесе прибуток, але це також ще не свідчить про те, що інвестор буде зацікавлений у фінансуванні даного проекту.

На п'ятому етапі розрахуємо відносну (щорічну) ефективність вкладених у наукову розробку інвестицій E_v . Для цього використаємо формулу:

$$E_v = \sqrt[T_{ж}]{1 + \frac{E_{abc}}{PV}} - 1 \quad (4.11)$$

де E_{abc} – абсолютна ефективність вкладених інвестицій, грн;

PV теперішня вартість інвестицій $PV = 3B$, грн;

$T_{ж}$ – життєвий цикл наукової розробки, роки.

$$E_v = \sqrt[3]{1 + \frac{539976,97}{76260,00}} - 1 = 1, \text{ або } 100\%.$$

Порівняємо E_v з мінімальною (бар'єрною) ставкою дисконтування τ_{\min} , яка визначає ту мінімальну дохідність, нижче за яку інвестиції вкладатися не будуть. У загальному вигляді мінімальна (бар'єрна) ставка дисконтування визначається за формулою:

$$\tau = d + f, \quad (4.12)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2020 році в Україні $d = 0,2$;

f – показник, що характеризує ризикованість вкладень; зазвичай, величина $f = (0,05 \dots 0,1)$.

$$\tau = 0,2 + 0,1 = 0,3$$

Оскільки $E_v = 100\% > \tau_{\min} = 0,3 = 30\%$, то інвестор може бути зацікавлений у фінансуванні даної наукової розробки.

Розрахуємо термін окупності вкладених у реалізацію проєкту інвестицій.

Термін окупності вкладених у реалізацію проєкту інвестицій $T_{ок}$ можна розрахувати за формулою:

$$T_{ок} = 1 / E_v \text{ [роки]}, \quad (4.13)$$

$$T_{ок} = 1 / 1 = 1 \text{ (рік)}.$$

Термін окупності вкладених у реалізацію проєкту інвестицій становить 1 рік. Аскільки $T_{ок} < 3 \dots 5$ років, що свідчить, що фінансування даної наукової розробки є доцільним.

Отже, розрахунок ефективності вкладених інвестицій та періоду їх окупності показав, що фінансування розробки є доцільним.

4.5 Висновок

В даному розділі було виконано оцінювання комерційного потенціалу розробки клієнтської частини інформаційної технології прийняття рішень в умовах критеріальної невизначеності.

Проведено визначення рівня комерційного потенціалу розробки із залученням трьох незалежних експертів. Аналіз показав, що рівень комерційного потенціалу розробки є вищим середнього.

Аналіз комерційного потенціалу розробки показав, що розроблена інформаційна система за своїми показниками випереджає аналогічні інформаційні системи і має велику перспективність. Продукт має кращі функціональні показники, а тому є досить конкурентоспроможним на ринку.

Згідно із розрахунками кошторисів на виконання науково-дослідної, дослідно-конструкторської та конструкторсько-технологічної роботи загальні витрати на розробку складають 76260,00 грн.

Було розраховано абсолютну ефективність вкладених інвестицій. Ефективність склала суму 539976,97 грн що свідчить про отримання вигоди від залучення коштів у розробку.

Щорічна ефективність вкладених в наукову розробку інвестицій складає 100 %, цей показник є вищим за мінімальну бар'єрну ставку дисконтування, яка складає 30%.

Термін повернення вкладених у проект інвестицій становить 1 рік, це додатково свідчить про доцільність фінансування нової розробки.

ВИСНОВКИ

У даній магістерській кваліфікаційній роботі проведено дослідження присвячене проблемі прийняття рішень в умовах критеріальної невизначеності та розроблено інформаційну технологію прийняття рішень в умовах критеріальної невизначеності. Проаналізувавши предметну область визначено, що прийняття рішень відіграє важливу роль у бізнесі, маркетингу, наукових дослідженнях та загалом житті кожної людини. Визначено основні проблеми, що виникають під час прийняття рішень, технології, за допомогою яких їх можна усунути та вимоги до інформаційної технології прийняття рішень в умовах критеріальної невизначеності.

Також під час виконання магістерської кваліфікаційної роботи досліджено та описано основні етапи, з яких складається процес прийняття рішень, визначено значення, вплив та важливість етапів.

Проаналізовано методи визначення коефіцієнта компетентності експерта. Серед методів самооцінки, методу аналізу узгодженості експертів та методу групової взаємооцінки обрано останій. Це й метод дозволяє розглядати експертів як альтернативи та за допомогою методу визначення важливості альтернативи провести оцінку компетентності експертів. Аналіз методів експертної оцінки показав, що метод ранжувань корисний, коли кількість альтернатив менша 10. Для випадку, коли кількість альтернатив більша 10 – доцільніше використовувати метод парних порівнянь.

Виконано проектування та розробку програмного забезпечення інформаційної технології. Опираючись на поставлені задачі, визначено структуру програмного продукту, розроблено та описано алгоритм роботи системи. Обґрунтовано вибір мови програмування JavaScript, основними перевагами якої є: кросплатформеність і широкий вибір фреймворків. Проведено тестування розробленої інформаційної системи, воно показало, що система працює відповідно до поставлених завдань. При порівнянні розробленого продукту з аналогами, розроблюваний програмний продукт показав кращі

результати, а саме покращення ефективності прийняття рішень в середньому на 10,5%, що підтверджує досягнення мети магістерської кваліфікаційної роботи.

Наукова новизна результатів дослідження полягає в запропонованій комплексній оцінці та впорядкуванні альтернатив в результаті діяльності експертів, що на відміну від існуючих, враховує думку кожного експерта з урахуванням його компетентності.

Прогнозування та оцінювання економічного потенціалу розробки, витрат на виконання науково-дослідної, дослідно-конструкторської та конструкторсько-технологічної роботи, комерційні ефекти від реалізації результатів розробки, ефективності вкладених інвестицій та періоду їх окупності показало великий потенціал для принесення прибутку інвесторам.

Результати досліджень пройшли експериментальне випробовування і впроваджені у ТОВ «АСТА.МОБІ».

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. В. В. Колодний, В. О. Мартишев: Візуалізація визначення вагових коефіцієнтів важливості критеріїв/ Тези доповідей Регіональної НПК ФІТКІ, ВНТУ – 2019 – Електронний репозитарій ВНТУ.
2. В. О. Мартишев, В. В. Колодний, «Комп’ютерна програма “Візуальний квантифікатор важливості критеріїв (ВКВК)” Свідоцтво про реєстрацію авторського права на твір № 79232. Дата реєстрації 22.05.2018.
3. О.І. Кушлик-Дивульська, Б.Р. Кушлик. Основи теорії прийняття рішень. – Київ, НТУУ «КПІ», 2014. – 3с.
4. Т.Саати Принятие решений, Метод анализа иерархий, Перевод с английского Р.Г. Вачнадзе, Москва "Радио и связь" 1993
5. Турунтаев Л.П. Теория принятия решений: Учебное пособие. - Томск: Томский межвузовский центр дистанционного образования, 2007. – 197с.
6. Теорія прийняття рішень – КПІ [Електронний ресурс]. – Режим доступу: URL: <http://mses.kpi.ua/knigi/zmist/TPR.pdf>. - Назва з екрану.
7. Documentation – Mind Decider – [Електронний ресурс]. – Режим доступу: <http://www.mindecider.com/ru/Products.MDPro.htm>
8. Documentation – PriEst – [Електронний ресурс]. – Режим доступу: <https://sourceforge.net/p/priority/wiki/Home/>
9. Учебно-методичний посібник: Методи експертних оцінок. III «Стратегія» м. Жовті Води, 1997.
10. Евланов Л.Г., Кутузов В.А. Экспертные оценки в управлении. М.: Экономика, 1996 133 с.
11. Добров Г.М., Ершов Ю.В., Левин Е.И., Смирнов Л.П. Экспертные оценки в научно-техническом прогнозировании. Киев: Наукова думка, 1994. 263с.
12. Данелян, Т. Я. Экономические информационные системы (ЭИС) предприятий и организаций / Т.Я. Данелян. - М.: Юнити-Дана, 2015. - 284 с.

13. Добров Г.М., Єршов Ю.В., Левін Є.І., Смир новий Л. П. Експертні оцінки на науково-технічному прогнози рованії. Київ: Наукова думка, 1974. 263
14. 11. Евланов Л.Г. Прийняття рішень на умовах невизначеності. М.: ИУНХ, 1976. 196 з.
15. 12. Евланов Л.Г., Кутузов В.А. Експертні оцінки на управлінні. М.: Економіка, 1978. 133 з.
16. 13. Карданская М. Прийняття управлінського рішення. М.: ЮНИТИ, 1999. 407с.
17. Бурков Е.А. Методы и алгоритмы анализа и агрегирования групповых экспертных оценок. Автореф. дис. на соискание ученой степени канд. техн. наук. Санкт-Петербург, 2011.
18. Бурков Е.А. Определение компетентности экспертов на основе поставленных ими оценок // Известия СПбГЭТУ «ЛЭТИ». 2009. № 4.
19. Бурков Е.А., Карпачевский А.В., Падерно П.И. Оценка компетентности экспертов на основе результативности их участия в экспертизах // Известия СПбГЭТУ «ЛЭТИ». 2011. № 10.
20. Бухарин С.Н., Цыганов В.В. Методы и технологии информационных войн. М.: Академический проект, 2007.
21. Дюбуа Д., Прад А. Теория возможностей. М.: Радио и связь, 1990.
22. Литвак Б.Г. Экспертная информация. Методы получения и анализа. М., 2009.
23. Миркин Б.Г. Проблема группового выбора. М.: Наука, 1974.
24. Романюк О. Н. Веб-дизайн і комп'ютерна графіка. Навч. посібник / О. Н. Романюк, Д. І. Кательніков, О. П. Косовець. – В.: ВНТУ, 2007. – 147 с.
25. Байдачный С. С. Маленко Д. А. ASP .NET 2. 0: Секреты создания Web-приложений СОЛОН-ПРЕСС. 2007. 5-98003-295-9.
26. Крис Джамса. Эффективный самоучитель по креативному Web-дизайну / Крис Джамса, Конрад Кинг, Энди Андерсон. - ДиаСофтЮП, 2005 г. - 672с.

27. Беллиньясо М. Разработка Web-приложений в среде ASP .NET 2.0. . Перевод с англ. / М. Беллиньясо – М.: Вильямс, 2007. – 640 с.
28. Стотлемайер Д. Тестирование Web-приложений / Д. Стотлемайер – М.: Кудиц-образ, 2003. – 240 с.
29. Грекул, В. И. Управление внедрением информационных систем / В.И. Грекул, Г.Н. Денищенко, Н.Л. Коровкина. - Москва: РГГУ, 2014. - 224 с.
30. Angular – [Электронный ресурс]. – Режим доступа: <https://angular.io/>
31. React – [Электронный ресурс]. – Режим доступа: <https://react.org/>
32. Vue.js – [Электронный ресурс]. – Режим доступа: <https://vuejs.org/>
33. Мак-Дональд, Мэтью Silverlight 5 с примерами на TypeScript для профессионалов . / Мэтью Мак-Дональд. - М.: Вильямс, 2013. - 848 с.
34. Методи структурного аналізу і проектування [Электронный ресурс]. – Режим доступа: <http://or-rsv.narod.ru/SADT/SADT.htm>
35. Липаев В.В. Проектирование программных средств: Учебное пособие для вузов. / Липаев В.В.– М.: Высшая школа, 2006.
36. Майер-Шенбергер В., Кукьер К. Большие данные. Революция, которая изменит то, как мы живем, работаем и мыслим: пер. с англ./под ред. И. Гайдюк, М.: Манн, Иванов и Фербер, 2014. 221 с.
37. Маклаков С.В. BPWin и ERWin. CASE-средства разработки информационных систем / Маклаков С.В – М.: ДИАЛОГ-МИФИ, 2015.
38. Борщ В. І., Оцінювання якості управлінського рішення як результату управлінської діяльності / В. І. Борщ // Ринкова економіка: сучасна теорія і практика управління. - 2014. - Т. 1, вип. 2. - С. 151-158.
39. Методичні вказівки до виконання студентами-магістрантами економічної частини магістерських кваліфікаційних робіт / Уклад. В. О. Козловський – Вінниця: ВНТУ, 2012. – 22 с.