

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту  
(повна назва)

Кафедра Інформатики  
(повна назва)

**КВАЛІФІКАЦІЙНА РОБОТА**  
**Пояснювальна записка**

рівень вищої освіти перший (бакалаврський)

**РОЗРОБЛЕННЯ ВЕБЗАСТОСУНКУ ДЛЯ ВИЗНАЧЕННЯ**  
**НАЙКРАЩОЇ АЛЬТЕРНАТИВИ ЗА ТЕХНОЛОГІЄЮ АНАЛІТИЧНО-**  
**ІЄРАРХІЧНОГО ПРОЦЕСУ ПРИЙНЯТТЯ ТА ОБҐРУНТУВАННЯ**  
**РІШЕНЬ**

(тема)

Виконав:  
студент 4 курсу, групи ІТІНФ-19-1

Гавриленко І.О.  
(прізвище, ініціали)

Спеціальності 122 Комп'ютерні науки  
(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Інформатика  
(повна назва освітньої програми)

Керівник доц. Творошенко І.С.  
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри \_\_\_\_\_  
(підпис)

Кобилін О.А.  
(прізвище, ініціали)

2023 р.

## Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту  
(повна назва)Кафедра Інформатики  
(повна назва)Рівень вищої освіти перший (бакалаврський)Спеціальність 122 Комп'ютерні науки  
(код і повна назва)Тип програми освітньо-професійнаОсвітня програма Інформатика  
(повна назва освітньої програми)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

« \_\_\_\_ » \_\_\_\_\_ 2023 р.

**ЗАВДАННЯ**  
НА КВАЛІФІКАЦІЙНУ РОБОТУстудентові Гавриленку Іллі Олександровичу  
(прізвище, ім'я, по батькові)1. Тема роботи Розроблення вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень

затверджена наказом університету від 15 травня 2023 року № 474 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 02 червня 2023 р.

3. Вихідні дані до роботи науково-методична та науково-технічна література, матеріали конференцій, дані інтернет-мережі, Visual Studio, Visual Studio Code, вебресурс drawio.

4. Перелік питань, що потрібно опрацювати в роботі \_\_\_\_\_

1. Аналіз предметної області за допомогою методів системного аналізу.

2. Розробка концепції та структури вебзастосунку для визначення найкращої альтернативи з використанням АІР.

3. Розробка програмного забезпечення на основі визначеної структури та концепції.

4. Тестування та оцінка розробленого вебзастосунку на прикладних задачах.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Актуальність проблеми, постановка задачі, блок-схеми процесів, вибір інструментальних засобів та мов програмування, розроблення вебзастосунку, тестування вебзастосунку, висновки.

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Консультант з дотримання діючих стандартів та норм	Доцент Творошенко І.С.		

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	10.04.2023	
2	Аналіз завдання, підбір літератури	10.04.2023	
3	Аналіз літератури з досліджуваної проблеми	11.04.23-15.04.23	
4	Аналіз технічних засобів	16.04.23-17.04.23	
5	Розробка вебзастосунку	17.04.23-21.04.23	
6	Тестування вебзастосунку	22.04.23-31.04.23	
7	Оформлення пояснювальної записки	01.05.23-10.06.23	
8	Перевірка на плагіат	11.06.2023	
9	Рецензування	11.06.2023	
10	Підготовка презентації та доповіді	09.06.23-11.06.23	
11	Занесення роботи в електронний архів	11.06.2023	
12	Попередній захист кваліфікаційної роботи	11.06.2023	

Дата видачі завдання 10 квітня 2023 р.

Студент \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_ доц. Творошенко І.С.  
(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ/ABSTRACT

Пояснювальна записка до кваліфікаційної роботи: 45 с., 3 табл., 12 рис., 34 джерела.

ВЕБЗАСТОСУНОК, АНАЛІТИЧНО-ІЄРАРХІЧНИЙ ПРОЦЕС ПРИЙНЯТТЯ РІШЕНЬ, ПРИЙНЯТТЯ РІШЕНЬ, НАЙКРАЩА АЛЬТЕРНАТИВА, КРИТЕРІЙ, БЛОК-СХЕМА, СИСТЕМНИЙ АНАЛІЗ, ШКАЛА СААТІ, МАТРИЦЯ ПОРІВНЯНЬ КРИТЕРІЇВ, ПОКАЗНИК ЯКОСТІ АЛЬТЕРНАТИВИ.

Об'єктом роботи є процес визначення найкращої альтернативи.

Метою роботи є розробка вебзастосунку, що базуються на використанні аналітично-ієрархічного процесу прийняття та обґрунтування рішень, та визначає найкращу альтернативу.

Використано методи системного аналізу для прийняття та обґрунтування рішень. Проаналізовано процеси створення та наповнення значеннями критеріїв, створення та наповнення значеннями альтернатив, створення та наповнення значеннями попарних порівнянь критеріїв, підрахунку власних векторів критеріїв та альтернатив, ваги критеріїв та альтернатив, відносні ваги альтернатив та вивід найкращої альтернативи.

У результаті роботи здійснена програмна реалізація вебзастосунку пошуку найкращої альтернативи.

WEB APPLICATION, ANALYTICAL-HIERARCHICAL DECISION-MAKING PROCESS, DECISION-MAKING, BEST ALTERNATIVE, CRITERION, FLOWCHART, SYSTEM ANALYSIS, SAATI SCALE, CRITERIA COMPARISON MATRIX, ALTERNATIVE QUALITY INDICATOR.

The object of the work is the process of determining the best alternative.

The aim of the work is to develop web applications based on the use of an analytical hierarchical decision-making and justification process, and provide the best alternative.

Methods of system analysis are used to make and justify decisions. The processes of creating and filling criteria values, creating and filling alternatives values, creating and filling pairwise comparisons of criteria with values, calculating eigenvectors of criteria and alternatives, weights of criteria and alternatives, relative weights of alternatives and output of the best alternative are analyzed.

As a result of this work, a software implementation of the web application for finding the best alternative was carried out.

## ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів .....	6
Вступ.....	7
1 Аналіз існуючих вебзастосунків для визначення найкращої альтернативи.....	7
1.1 Сучасний стан розвитку вебзастосунків для визначення найкращої альтернативи в Україні та за кордоном .....	8
1.2 Аналіз літературних джерел щодо існуючих підходів розроблення вебзастосунків для процесів прийняття та обґрунтування рішень.....	15
1.3 Постановка задачі .....	22
2 Моделювання процесу визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень.....	23
2.1 Застосування методів системного аналізу для прийняття та обґрунтування рішень.....	23
2.2 Моделювання структури та наповнення вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень.....	28
3 Розроблення вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень .....	33
3.1 Вибір інструментальних засобів для реалізації поставленої задачі.....	33
3.2 Етапи розроблення вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень .....	34
3.3 Тестування реалізованого вебзастосунку та аналіз результатів ....	35
3.4 Перспективи подальшої роботи .....	40
Висновки .....	42
Перелік джерел посилання .....	43

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,  
СКОРОЧЕНЬ І ТЕРМІНІВ**

ITS – Intelligent Tutoring System (інтелектуальна транспортна система)

HFLT – Hesitant Fuzzy Linguistic Term (сумнівні нечіткі лінгвістичні терміни)

AI – Artificial Intelligence (штучний інтелект)

AIP – Analytical and Hierarchical Process (аналітично-ієрархічний процес)

SADT – Structured Analysis and Design Technique (методика структурованого аналізу і проектування)

API – Application Programming Interface (інтерфейс прикладного програмування)

ПЗ – програмне забезпечення

## ВСТУП

Сучасний світ характеризується швидким розвитком технологій та постійним ростом інформаційного обсягу [1]. Відповідно, знаходження найкращої альтернативи стає все більш актуальним завданням для споживачів та компаній [2]. Вебзастосунки, які допомагають у визначенні найкращих альтернатив, можуть мати значний вплив на якість прийняття рішень, підвищуючи ефективність та задоволення від користування послугами.

Розробка вебзастосунку для визначення найкращої альтернативи стає надзвичайно актуальною задачею [3]. В умовах швидкого темпу розвитку технологій та постійної конкуренції, ефективне прийняття та обґрунтування рішень грають вирішальну роль у досягненні успіху [4]. Саме тому розробка вебзастосунку, що базується на аналітично-ієрархічному процесі, стає потужним інструментом під час визначення найкращої альтернативи [5].

Проєкт спрямований на створення вебзастосунку, який дозволить користувачам здійснювати оцінку та порівняння альтернатив з використанням методу аналітично-ієрархічного процесу [6]. Вебзастосунок надасть можливість систематизувати та вагомо оцінити фактори, які впливають на прийняття рішень, враховуючи їх важливість та пріоритетність. Він надасть користувачам зручний інтерфейс, що дозволить легко вводити дані та отримувати результати.

Вебзастосунок виявиться незамінним помічником для менеджерів, аналітиків та фахівців у галузях, де вирішення складних проблем та вибір оптимальних рішень є критичним [7]. Завдяки його використанню, замість суб'єктивних суджень та неструктурованих процесів, буде здійснено максимально об'єктивний підхід до прийняття рішень.

Розробка зазначеного вебзастосунку має потенціал значно полегшити процес прийняття рішень, покращити якість аналізу та допомогти уникнути можливих помилок чи побічних ефектів.

# 1 АНАЛІЗ ІСНУЮЧИХ ВЕБЗАСТОСУНКІВ ДЛЯ ВИЗНАЧЕННЯ НАЙКРАЩОЇ АЛЬТЕРНАТИВИ

## 1.1 Сучасний стан розвитку вебзастосунків для визначення найкращої альтернативи в Україні та за кордоном

Сучасний стан розвитку вебзастосунків для визначення найкращої альтернативи в Україні та за кордоном відображає широкий спектр можливостей та технічних рішень, які допомагають спрощувати процеси прийняття рішень та забезпечувати ефективність різноманітних систем. У даній роботі розглядаються сучасні підходи до створення вебзастосунків для визначення найкращої альтернативи, їх переваги та недоліки. Відповідно до цієї теми, досліджуються наукові роботи та практичні приклади з використанням відповідних джерел.

У контексті дослідження вебзастосунків для визначення найкращої альтернативи можна виділити декілька основних напрямків робіт.

Перший напрямок включає в себе розробку систем, що використовують принципи штучного інтелекту, нейромережі, нечітку логіку та формальну логіку для моделювання складних систем та прийняття рішень [8–11].

Другий напрямок зосереджується на розробці методів інтелектуального аналізу та оброблення даних, які допомагають визначити найкращі альтернативи в різних областях, таких як класифікація зображень, медичні інформаційні системи та тестування вебзастосунків [12–14].

Розглянемо найбільш популярні сервіси прийняття та обґрунтування рішень.

Amazon Personalize – це сервіс машинного навчання від компанії Amazon, який допомагає створювати персоналізовані рекомендації для користувачів на основі їх поведінки та інтересів [15].

Переваги:

- легкість інтеграції з іншими сервісами Amazon;

- автоматичне масштабування та оптимізація на основі користувацької активності;

- використання передових алгоритмів машинного навчання.

Недоліки:

- відносно висока вартість використання;
- обмежений набір функцій порівняно з іншими аналогами;
- потенційні проблеми з конфіденційністю даних.

Google Recommendations AI – це сервіс машинного навчання від компанії Google, який допомагає створювати персоналізовані рекомендації для користувачів на основі їх поведінки та інтересів [16].

Переваги:

- інтеграція з іншими сервісами Google;
- використання передових алгоритмів машинного навчання;
- можливість керувати даними та моделями.

Недоліки:

- відносно висока вартість використання;
- обмежений набір функцій;
- можливі проблеми з конфіденційністю даних.

Microsoft Azure Personalizer – це сервіс машинного навчання від компанії Microsoft, який допомагає створювати персоналізовані рекомендації для користувачів на основі їх поведінки та інтересів [17].

Переваги:

- інтеграція з іншими сервісами Microsoft Azure;
- використання передових алгоритмів машинного навчання;
- можливість керувати даними та моделями.

Недоліки:

- відносно висока вартість використання;
- обмежений набір функцій порівняно з іншими аналогами;
- можливі проблеми з конфіденційністю даних.

IBM Watson Recommendations – це сервіс машинного навчання від компанії IBM, який допомагає створювати персоналізовані рекомендації для користувачів на основі їх поведінки та інтересів [18].

Переваги:

- інтеграція з іншими сервісами IBM Watson;
- використання передових алгоритмів машинного навчання;
- можливість керувати даними та моделями.

Недоліки:

- відносно висока вартість використання;
- обмежений набір функцій порівняно з іншими аналогами;
- можливі проблеми з конфіденційністю даних.

Усі перелічені сервіси використовують передові алгоритми машинного навчання та мають здатність інтеграції з іншими сервісами своїх компаній, але мають обмежений набір функцій, відносно високу вартість використання, також можливі проблеми з конфіденційністю даних. Завжди варто враховувати специфіку ситуації та вимоги до системи при виборі платформи для прийняття рішень.

Системи на основі штучного інтелекту та нейромереж відрізняються від традиційних систем обробки даних своєю глибокою структурою обчислень і високою ступенем адаптивності [19]. Ці системи здатні вчитися, адаптуватися та виконувати складні завдання, такі як розпізнавання образів, розпізнавання мови, прогнозування та оптимізація, використовуючи великі обсяги даних [20]. Їх можна «навчати» за допомогою наборів даних, що використовують алгоритми машинного навчання для визначення закономірностей та тенденцій. У результаті цього процесу системи на основі штучного інтелекту та нейромереж можуть виконувати завдання, які раніше були можливі тільки для людей, а іноді навіть з більшою точністю або швидкістю.

Переваги:

- здатність адаптуватися до змінних умов та враховувати нові дані;

- висока точність визначення найкращої альтернативи завдяки використанню нейромереж та штучного інтелекту;
- швидке опрацювання великих обсягів інформації.

Недоліки:

- складність розробки та налаштування алгоритмів;
- високі вимоги до обчислювальних ресурсів;
- потреба у великій кількості навчальних даних для забезпечення точності моделі.

DeepMind – дочірнє підприємство Google, лідер у сфері штучного інтелекту, використовує технології глибокого навчання та нейронних мереж для розвитку нових продуктів та рішень [21].

Переваги:

- відомий проривами в глибокому навчанні, включаючи розробку AlphaGo, першої AI, що перемогла людину в грі Go;
- відіграє важливу роль у розвитку AI в сфері охорони здоров'я, допомагаючи лікарям передбачати певні стани пацієнтів у реальному часі.

Недоліки:

- є дослідницькою організацією, більшість її розробок не доступні для загального використання відразу;
- нейромережі, на які спирається багато досліджень DeepMind, потребують великої кількості даних для навчання, що може бути проблематичним з точки зору приватності та етики.

Методи інтелектуального аналізу та оброблення даних використовуються для виявлення зв'язків, шаблонів та тенденцій у великих наборах даних [22]. Ці методи, зокрема, включають статистичний аналіз, прогнозування, класифікацію, кластеризацію та асоціативний аналіз. Методи інтелектуального аналізу та оброблення даних набули особливої важливості в сучасному світі, адже обсяги даних постійно зростають, і потрібно швидко й ефективно оброблювати їх.

#### Переваги:

- універсальність та гнучкість методів, які можуть бути застосовані в різних областях (класифікація зображень, медичні системи, тестування вебзастосунків);
- здатність працювати з різними форматами та структурами даних;
- відносно невеликі вимоги до обчислювальних ресурсів порівняно з нейромережами.

#### Недоліки:

- нижча точність визначення найкращої альтернативи порівняно з методами на основі штучного інтелекту;
- потреба у попередній обробці даних та виділенні відповідних ознак для аналізу, що може бути трудомістким процесом;
- можливість неправильної інтерпретації результатів аналізу, якщо методи застосовуються некоректно.

RapidMiner – платформа для передового аналізу даних та машинного навчання, яка пропонує широкий спектр алгоритмів та методологій [23].

#### Переваги:

- дозволяє проводити передовий аналіз даних без необхідності в програмуванні;
- підтримує весь процес машинного навчання, від перетворення та обробки даних до моделювання та впровадження.

#### Недоліки:

- може бути складним для початківців через свій широкий набір можливостей та функцій;
- висока вартість для деяких користувачів, особливо для малого та середнього бізнесу.

Експертні системи – різновид штучного інтелекту, який використовує базу знань з певної доменної області та механізм виведення для надання консультацій або рекомендацій, схожих на ті, які надав би експерт [24]. Такі системи здатні моделювати процес прийняття рішень людини-експерта та

вирішувати складні проблеми на основі предметного знання. Відомі експертні системи, такі як MYCIN та DENDRAL, знайшли широке застосування в медицині та хімії відповідно. Експертні системи мають значне значення, оскільки вони дозволяють «закодувати» та використовувати знання людей-експертів, зберігаючи його для майбутніх поколінь та забезпечуючи широкий доступ до цього знання.

#### Переваги:

- висока точність визначення найкращої альтернативи завдяки використанню знань та досвіду експертів в галузі;
- можливість урахування специфічних особливостей проблеми та різних вимог до рішень;
- відносно простий процес розробки та налаштування системи.

#### Недоліки:

- обмеженість експертних знань, які можуть не враховувати всі можливі ситуації та альтернативи;
- залежність від досвіду та об'єктивності експертів, що може призвести до упереджених рішень;
- нижча швидкість опрацювання великих обсягів інформації порівняно з методами на основі штучного інтелекту та нейромереж.

Дані аналізу вказують на актуальність та потребу в удосконаленні існуючих вебзастосунків для визначення найкращої альтернативи. Тему розроблення вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень було обрано з метою покращення якості прийняття рішень, враховуючи переваги та недоліки існуючих аналогів, а також використання новітніх технологій та нейромереж.

MYCIN – експертна система, розроблена в 1970-х роках для діагностики інфекційних захворювань крові та призначення антибіотиків.

#### Переваги:

- один з перших системних застосувань AI і відіграв важливу роль у розвитку експертних систем;
- досить точний в діагнозах і рекомендаціях, часто переважаючи лікарів.

#### Недоліки:

- ніколи не був використаний в клінічній практиці через регулятивні та етичні питання, що ставлять під сумнів його практичне застосування;
- обмежений специфічною доменною областю, не міг бути застосований до інших медичних проблем без значних змін.

Таким чином, вдосконалення вебзастосунків для визначення найкращої альтернативи передбачає:

- інтеграцію методів на основі штучного інтелекту. Використання методів на основі штучного інтелекту, таких як нейромережі та алгоритми машинного навчання, може значно покращити якість прийняття рішень, оскільки ці методи забезпечують високу точність та швидкість опрацювання великих обсягів інформації. Інтеграція цих методів у вебзастосунки для визначення найкращої альтернативи дозволить користувачам отримувати більш виважені та об'єктивні рішення [25];
- розробка інтуїтивного та простого у використанні інтерфейсу. Сучасні вебзастосунки повинні мати інтуїтивний та простий у використанні інтерфейс, який дозволяє користувачам швидко та легко вводити дані та отримувати результати аналізу. Цього можна досягти за допомогою розробки зручного меню, яке відображає усі доступні опції та функції, а також за допомогою створення контекстної довідки та підказок для користувачів [26];
- урахування специфічних особливостей та вимог до рішень. Кожен випадок прийняття рішення має свої специфічні особливості та вимоги до рішень. Урахування особливостей та вимог при розробці вебзастосунків для визначення найкращої альтернативи дозволить користувачам отримувати більш точні та адаптивні результати [27];

– забезпечення безпеки даних та конфіденційності. Дотримання вимог до безпеки даних та конфіденційності є важливим аспектом при розробці вебзастосунків для визначення найкращої альтернативи. Це включає в себе захист інформації від несанкціонованого доступу та забезпечення її конфіденційності під час передачі та зберігання. Доцільно використовувати сучасні криптографічні методи та алгоритми для забезпечення безпеки даних та їх цілісності [28].

Таким чином, сучасний стан розвитку вебзастосунків для визначення найкращої альтернативи в Україні та за кордоном свідчить про високу актуальність та популярність цієї теми. Аналіз аналогів показав, що існують як вітчизняні, так і зарубіжні розробки, які вже демонструють певні результати. Однак, з огляду на недоліки існуючих аналогів, існує потреба у вдосконаленні вебзастосунків для визначення найкращої альтернативи. Для досягнення цього можна використовувати методи на основі штучного інтелекту, розробляти інтуїтивні та прості у використанні інтерфейси, враховувати специфічні особливості та вимоги до рішень, а також забезпечувати безпеку даних та конфіденційність.

## 1.2 Аналіз літературних джерел щодо існуючих підходів розроблення вебзастосунків для процесів прийняття та обґрунтування рішень

У роботі [29] автори проводять систематичний огляд ITS – інтелектуальних навчальних систем – з метою визначення характеристик, областей застосування та методів оцінювання. У роботі не було розроблено чи спроектовано конкретний вебзастосунок, але результати дослідження можуть бути корисними для розробки вебзастосунків на основі ITS.

Переваги, виявлені в результаті аналізу інтелектуальних навчальних систем, включають:

– адаптивність – ITS можуть адаптуватися до індивідуальних потреб і рівня знань кожного студента, що сприяє більш ефективному навчанню;

- інтерактивність – ITS забезпечують багаторівневу взаємодію з користувачами, що підвищує мотивацію до навчання;

- підтримка співпраці – деякі ITS дозволяють студентам співпрацювати, обмінюватися знаннями та виконувати спільні завдання.

Недоліки та обмеження інтелектуальних навчальних систем:

- висока вартість розробки – розробка ITS може бути складною та коштовною через необхідність інтеграції різних технологій;

- необхідність постійної актуалізації – зміни в навчальних програмах та дидактичних матеріалах вимагають постійного оновлення ITS;

- обмеження штучного інтелекту – штучний інтелект, використовуваний в ITS, може мати обмеження у розумінні контексту та нюансів людської мови, що може призвести до неправильного або неповного засвоєння навчального матеріалу студентами;

- проблеми з приватністю – збір та аналіз даних про студентів можуть порушувати приватність користувачів, якщо не застосовуються відповідні заходи безпеки.

Хоча у роботі не було розроблено конкретного вебзастосунок, у цілому результати дослідження можуть слугувати основою для розробки нових інтелектуальних навчальних систем та вебзастосунків.

У роботі [30] було проведено огляд техніки нечіткої лінгвістичної оцінки з використанням так званих сумнівних нечітких лінгвістичних термінів (HFLT). Хоча в рамках роботи не було створено конкретний вебзастосунок, її результати можуть послужити основою для розробки вебзастосунків, які використовують HFLT для підтримки прийняття рішень у різних контекстах.

Основні переваги використання HFLT в прийнятті рішень включають:

- забезпечення більш гнучкої та виразної моделі для представлення суб'єктивних оцінок та суджень;

- урахування неоднозначності та сумнівів, які можуть виникнути у процесі оцінки альтернатив або експертних думок;

- застосування HFLT для розв’язання різноманітних задач прийняття рішень, таких як оцінка проєктів, вибір постачальників, медична діагностика та ін.

Однак, використання HFLT також має деякі недоліки та обмеження:

- складність розуміння та інтерпретації HFLT, особливо для користувачів, які не мають попереднього досвіду в нечіткій логіці чи лінгвістичному моделюванні;

- потреба в розробці спеціалізованих алгоритмів агрегування та обробки HFLT, які можуть бути менш ефективними порівняно з традиційними методами прийняття рішень;

- відсутність універсального підходу до визначення оптимального набору лінгвістичних термінів, який може враховувати специфіку різних задач прийняття рішень та контекстів.

У цілому, результати роботи можуть бути корисними для розробки вебзастосунків, які використовують HFLT для підтримки прийняття рішень. Однак, потрібно враховувати зазначені недоліки та обмеження при практичному застосуванні таких систем.

У статті [31] не було розроблено конкретного вебзастосунку. Замість цього, автори досліджують різні аспекти використання штучного інтелекту (AI) для підтримки прийняття рішень в епоху великих даних.

Стаття аналізує розвиток AI-технологій, виклики та можливості їх застосування в контексті великих даних. Автори описують ряд підходів і технік, які можуть бути використані для розробки вебзастосунків, які використовують AI для підтримки прийняття рішень.

Переваги використання AI для прийняття рішень в епоху великих даних включають:

- здатність швидко обробляти та аналізувати великі обсяги даних з різних джерел, що може поліпшити якість рішень та прискорити процес прийняття рішень;

- використання алгоритмів машинного навчання та інших AI-технік може допомогти виявляти важливі залежності, закономірності та тенденції в даних, які можуть не бути очевидними для людей;

- AI-застосунки можуть надавати персоналізовані рекомендації та підказки, що враховують індивідуальні потреби та обставини користувачів.

Однак, використання AI для прийняття рішень також має деякі недоліки та виклики:

- збір та обробка великих обсягів даних можуть порушувати приватність користувачів та вимагає належних заходів забезпечення конфіденційності;

- AI-алгоритми можуть мати змішану якість та точність, що залежить від якості даних, на яких вони навчаються, та від обраної моделі;

- неправильно налаштовані алгоритми можуть призвести до помилкових рішень або пропущених можливостей;

- залежність від AI-технологій може зменшити роль людського експерта та відповідальності за прийняття рішень, що може призвести до недооцінки важливості людського елемента в процесі прийняття рішень;

- транспарентність та інтерпретабельність AI-моделей можуть бути обмеженими, особливо для глибоких нейронних мереж. Це може ускладнити розуміння того, як саме AI-алгоритми приймають рішення, та оцінку їх достовірності.

Стаття надає огляд можливостей та викликів використання AI для підтримки прийняття рішень в епоху великих даних. Хоча конкретний вебзастосунок не було розроблено в рамках цієї роботи, її результати можуть служити цінним джерелом інформації для дослідження та розробки AI-технологій у вебзастосунках.

У роботі [32] було розроблено вебзастосунок для підтримки прийняття рішень у сфері медичної діагностики, зокрема при виявленні інфаркту міокарда.

Цей вебзастосунок базується на аналізі даних про пацієнтів, включаючи симптоми, медичну історію, результати лабораторних тестів та інші показники.

Основні переваги цього вебзастосунку включають:

- швидке та точне виявлення інфаркту міокарда, що може забезпечити своєчасне лікування і, як результат, поліпшення прогнозу для пацієнтів;

- підтримка прийняття рішень для лікарів, допомагаючи їм більш точно інтерпретувати діагностичні дані та забезпечити кращу медичну допомогу;

- можливість адаптації системи під конкретні вимоги та потреби лікарів та медичних закладів.

Однак, є деякі недоліки та обмеження цього вебзастосунку:

- недостатність даних або неточність даних, внесених до системи, можуть призвести до неточної діагностики або хибних рекомендацій;

- система може потребувати великої кількості обчислювальних ресурсів для аналізу великих наборів даних, що може обмежити її доступність для деяких медичних закладів;

- потенційні проблеми з конфіденційністю та безпекою даних пацієнтів, які повинні бути враховані при використанні вебзастосунку в медичній практиці. Необхідність постійного оновлення та адаптації системи з урахуванням нових досліджень, методів діагностики та лікування інфаркту міокарда;

- можливі технічні проблеми, які можуть виникнути під час використання вебзастосунку, що вимагає відповідної підтримки та розробки.

Отже, вебзастосунок, розроблений під час роботи має ряд переваг, які можуть покращити процес діагностики інфаркту міокарда та підтримки прийняття рішень лікарями. Однак, також є ряд недоліків та обмежень, які слід враховувати при впровадженні та використанні цього вебзастосунку в медичній практиці.

У роботі [33] було розроблено вебзастосунок, який використовує вейвлет-аналіз для ідентифікації хвиль Елліота в фінансових ринках. Цей вебзастосунок має на меті підтримувати прийняття рішень у торгівлі на фінансових ринках.

Основні переваги цього вебзастосунку включають:

- використання вейвлет-аналізу дозволяє виявляти характерні особливості в даних фінансового ринку, що може допомогти у прийнятті рішень з торгівлі;
- ідентифікація хвиль Елліота може допомогти трейдерам прогнозувати майбутні рухи цін на фінансових ринках;
- вебзастосунок забезпечує зручний інтерфейс для аналізу та візуалізації даних.

Однак, є деякі недоліки та обмеження цього вебзастосунку:

- точність прогнозування може бути низькою, оскільки ринки є динамічними і нестабільними. Хвилі Елліота – лише один з багатьох факторів, які впливають на рух цін;
- вейвлет-аналіз може бути складним для розуміння та інтерпретації, особливо для користувачів без попереднього досвіду в області фінансів чи математики;
- робота базується на історичних даних, що може не враховувати нові тенденції або непередбачувані події, які можуть вплинути на ринок.

Отже, для забезпечення актуальності та ефективності рекомендацій, користувачам слід враховувати інші джерела інформації та аналізу, доповнюючи результати, отримані від вебзастосунку, який базується на хвилях Елліота.

Математичні методи для рішення задачі динамічного масштабування інфраструктури вебзастосунків в умовах обмежених ресурсів були досліджені у роботі [34].

Під час дослідження було розроблено вебзастосунок, який мав на меті допомогти управлінню ресурсами вебінфраструктури.

Вебзастосунок використовував різні математичні методи для автоматичного масштабування ресурсів з урахуванням поточної навантаженості та забезпечення оптимального рівня продуктивності.

Переваги розробленого вебзастосунку включали:

- автоматичне масштабування ресурсів на основі поточної навантаженості, що забезпечує оптимальний рівень продуктивності;
- застосування математичних методів дозволяло оптимізувати використання ресурсів та знижувати витрати на інфраструктуру;
- застосунок міг працювати з різними видами інфраструктури, що забезпечувало гнучкість вибору платформи.

Однак, існували також деякі недоліки:

- розроблений вебзастосунок був зосереджений лише на динамічному масштабуванні, тому інші аспекти управління ресурсами, такі як розподіл трафіку та аварійне відновлення, могли не бути повністю реалізовані;
- математичні методи, використані в роботі, могли бути складними для розуміння та інтеграції для тих, хто не має глибоких знань у галузі оптимізації.

Робота представляла цінний внесок у розвиток вебзастосунків для оптимізації використання ресурсів інфраструктури. Вебзастосунок, розроблений на основі математичних методів, дозволяв автоматично масштабувати ресурси відповідно до поточної навантаженості, що сприяло підвищенню продуктивності та зниженню витрат на інфраструктуру. Проте, слід врахувати, що робота мала деякі обмеження, такі як фокус на динамічному масштабуванні та складність математичних методів, які могли ускладнити інтеграцію та розуміння для користувачів без глибоких знань у галузі оптимізації.

Отже, аналіз джерел щодо розробки вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень допомогли виявити, що дану тему необхідно розвивати.

На сьогоднішній день розроблено декілька вебзастосунків, але вони є дуже вузьконаправленими. Немає універсального застосунку, який надає можливість вказати назви критеріїв, їх пріоритети, альтернативи, їх кількість.

### 1.3 Постановка задачі

Таким чином, розроблення вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень є актуальною темою.

Об'єктом роботи є процес визначення найкращої альтернативи.

Метою роботи є розробка вебзастосунку, що базується на використанні аналітично-ієрархічного процесу прийняття та обґрунтування рішень, та визначає найкращу альтернативу.

Для досягнення мети, необхідно вирішити наступні завдання:

- вивчити теоретичні основи методів системного аналізу, зокрема аналітично-ієрархічного процесу (AIP), для прийняття та обґрунтування рішень;
- проаналізувати існуючі вебзастосунки та інструменти, що використовують AIP для прийняття рішень, та визначити їхні переваги та недоліки;
- розробити концепцію та структуру вебзастосунку для визначення найкращої альтернативи з використанням AIP;
- розробити програмне забезпечення для реалізації вебзастосунку на основі визначеної структури та концепції, застосовуючи існуючий алгоритм;
- провести тестування та оцінку розробленого вебзастосунку на прикладних задачах відповідно до критеріїв ефективності та користувачів.

Результатом виконання вищеперелічених завдань стане розроблений вебзастосунок, який забезпечить об'єктивність вибору найкращої альтернативи з наявних.

## **2 МОДЕЛЮВАННЯ ПРОЦЕСУ ВИЗНАЧЕННЯ НАЙКРАЩОЇ АЛЬТЕРНАТИВИ ЗА ТЕХНОЛОГІЄЮ АНАЛІТИЧНО-ІЄРАРХІЧНОГО ПРОЦЕСУ ПРИЙНЯТТЯ ТА ОБҐРУНТУВАННЯ РІШЕНЬ**

2.1 Застосування методів системного аналізу для прийняття та обґрунтування рішень

Системний аналіз – це методологія дослідження складних інформаційних систем з метою їх розуміння та оптимізації. Такий аналіз дозволяє дослідити систему з різних точок зору та зрозуміти, як вона функціонує. Під час системного аналізу використовуються різноманітні методи, які спрямовані на визначення:

- формалізованої постановки задачі;
- структури системи;
- складних взаємозв'язків між елементами системи, їх вплив на роботу та ефективність системи;
- вузьких місць системи [4];
- шляхів вирішення існуючих проблем.

Метод дерева цілей є одним з методів системного аналізу, який візуалізує ієрархічне досягнення цілей і задач, які входять до системи [10]. Дерево цілей – принцип, головна мета якого досягається за рахунок сукупності другорядних і додаткових цілей.

Ієрархія – це структура, що складається з кількох рівнів, де кожен наступний рівень уточнює попередній і складається з елементів більш низького рівня.

Декомпозиція – це розбиття складної системи на менші компоненти.

Декомпозиція цілей – це метод, що використовується для декомпозиції загальної цілі на більш дрібні підцілі. Цей процес продовжується до тих пір,

поки не будуть закріплені за відповідними задачами конкретні виконавці та відповідальні.

У методі дерева цілей ієрархія розбивається на більш низькі рівні і декомпонується на підцілі, які допомагають досягнути загальної цілі. Це дозволяє розглядати систему як сукупність окремих компонентів, що допомагає побачити її в цілому і зрозуміти, які елементи впливають на загальний результат.

Метод дерева цілей використовується в ситуаціях, коли необхідно визначити, які цілі повинні бути досягнені для забезпечення ефективності системи, і які компоненти впливають на досягнення цих цілей.

Метод сценаріїв використовують для аналізу можливих варіантів розвитку подій та визначення наслідків прийнятих рішень.

Цей метод передбачає створення набору сценаріїв, які можуть статися в майбутньому в залежності від різних факторів та умов. Сценарії можуть бути як позитивні (наприклад, успішна реалізація проекту), так і негативні (наприклад, збій в роботі системи).

Для проведення аналізу методом сценаріїв спочатку необхідно визначити потенційні події та взаємозв'язки між ними. Далі для кожної події складаються можливі варіанти сценаріїв розвитку, описуються ймовірність та наслідки кожного з них. Після цього проводиться оцінка кожного сценарію та вибір найбільш ймовірного та оптимального варіанту.

У цілому метод сценаріїв дозволяє системно та комплексно аналізувати складні ситуації, що в свою чергу допомагає уникнути можливих ризиків та прийняти найкращі рішення.

Метод моделювання є одним з методів системного аналізу, який використовують для дослідження та аналізу складних систем, таких як економічних, технічних, природних тощо.

Основна ідея полягає в тому, щоб створити модель системи, яка буде відображати роботу всіх її компонентів та взаємодію між ними. За допомогою цієї моделі можна провести аналіз та різноманітні експерименти,

що дозволятимуть зрозуміти, як система працює, які є потенційні проблеми та як їх можна вирішити.

Метод моделювання дозволяє отримати глибше розуміння роботи системи та зробити обґрунтовані висновки щодо можливих рішень. Також дозволяє прогнозувати поведінку системи у різних умовах та впливати на її функціонування за допомогою зміни параметрів моделі.

SADT (Structured Analysis and Design Technique) є методологією системного аналізу, яка використовується для дослідження та проєктування інформаційних та інших систем. Цей метод використовується для опису та моделювання бізнес-процесів, аналізу потреб користувачів, формалізації вимог до системи, визначення потоків даних та створення моделей поведінки системи.

SADT відносять до методів моделювання, оскільки він зосереджується на побудові структурної моделі, яка показує взаємозв'язки між елементами системи. Цей метод дозволяє аналізувати складні системи та визначати ключові елементи та процеси, які потребують уваги.

SADT – основоположна методологія, що заклала принципи сучасного моделювання та лягла в основу розробки стандарту IDEF0.

Системний аналіз надає змогу різносторонньо підходити до будь-якої предметної області, у тому числі й до деталей прийняття та обґрунтування рішень.

Отже, був розроблений наступний алгоритм [11] визначення найкращої альтернативи:

Крок 1. Формування початкових даних щодо аналітично-ієрархічного процесу прийняття та обґрунтування рішень, формування мети завдання, надання переліку критеріїв досягнення мети та переліку можливих альтернатив.

Крок 2. Формування шкали відносної важливості. Попарне порівняння критеріїв досягнення мети у відповідності з такою шкалою (табл. 2.1).

Створення матриці порівнянь критеріїв та заповнення її, вказуючи власний вектор та вагу критерію (табл. 2.2).

Таблиця 2.1 – Шкала відносної важливості

Рівень важливості	Кількісне значення
Рівна важливість	1
Помірна перевага	3
Значна або сильна перевага	5
Значна (велика) перевага	7
Дуже велика перевага	9

Таблиця 2.2 – Заповнена матриця порівнянь критеріїв

Критерії	$K_1$	$K_2$	$K_3$	$K_4$	Власний вектор, $W_i$	Вага критерію, $w_i$
$K_1$	1	3	5	7	3,2	0,56
$K_2$	1/3	1	3	5	1,5	0,26
$K_3$	1/5	1/3	1	3	0,67	0,12
$K_4$	1/7	1/5	1/3	1	0,31	0,05

Компоненти власного вектора критерія розраховується за наступною формулою:

$$W_i = \sqrt[n]{(C_i/C_1) * (C_i/C_2) * \dots * (C_i/C_n)},$$

де  $C_j$  – кількісне значення рівня важливості  $j$ -ого критерію щодо  $i$ -ого критерію;

$n$  – кількість критеріїв.

Вага критерію розраховується за наступною формулою:

$$w_i = W_i / (W_1 + W_2 + \dots + W_n),$$

де  $W_i$  – власний вектор  $i$ -ого критерія;

$n$  – кількість критеріїв.

Крок 3. Попарне порівняння альтернативи за окремими критеріями, знайшовши їх відносну важливість. Створення матриці відносної важливості альтернатив за окремими критеріями, вказавши власний вектор та вагу альтернативи (табл. 2.13). Аналогічним чином знаходиться відносна важливість альтернатив за окремими критеріями.

Таблиця 2.3 – Попарне порівняння альтернатив за окремими критеріями та знаходження їх відносної важливості

Альтернатива	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	Власний вектор, $V_i$	Вага альтернативи, $v_i$
$A_1$	1	3	5	7	3	3,16	0,47
$A_2$	1/3	1	3	5	1	1,38	0,2
$A_3$	1/5	1/3	1	3	1/3	0,58	0,09
$A_4$	1/7	1/5	1/3	1	1/5	0,29	0,04
$A_5$	1/3	1	3	5	1	1,38	0,2

Компоненти власного вектора альтернативи за певним критерієм розраховується за наступною формулою:

$$V_i = \sqrt[n]{(C_i/C_j) * (C_i/C_2) * \dots * (C_i/C_n)},$$

де  $C_j$  – кількісне значення рівня важливості  $j$ -ої альтернативи щодо  $i$ -ої альтернативи за певним критерієм;

$n$  – кількість альтернатив.

Вага альтернативи розраховується за наступною формулою:

$$v_i = V_i / (V_1 + V_2 + \dots + V_n),$$

де  $V_i$  – власний вектор  $i$ -ої альтернативи;

$n$  – кількість критеріїв.

Крок 4. Розрахування відносної ваги (показник якості) кожної альтернативи.

$$C_j = \sum_{i=1}^N w_i v_{ij},$$

де  $w_i$  – вага  $i$ -ого критерія;

$v_{ij}$  – вага  $i$ -ої альтернативи за  $j$ -им критерієм;

$N$  – кількість критеріїв.

Крок 5. Обґрунтування отриманого рішення.

Максимальний показник якості певної альтернативи показує, що дана альтернатива є найкращою.

2.2 Моделювання структури та наповнення вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень

Детальний алгоритм процесів створення та наповнення значеннями критеріїв, створення та наповнення значеннями альтернатив, створення та наповнення значеннями попарних порівнянь критеріїв, підрахунку власних векторів критеріїв та альтернатив, ваги критеріїв та альтернатив, відносні ваги альтернатив та вивід найкращої альтернативи відображено на рисунках 2.1 – 2.4 відповідно.

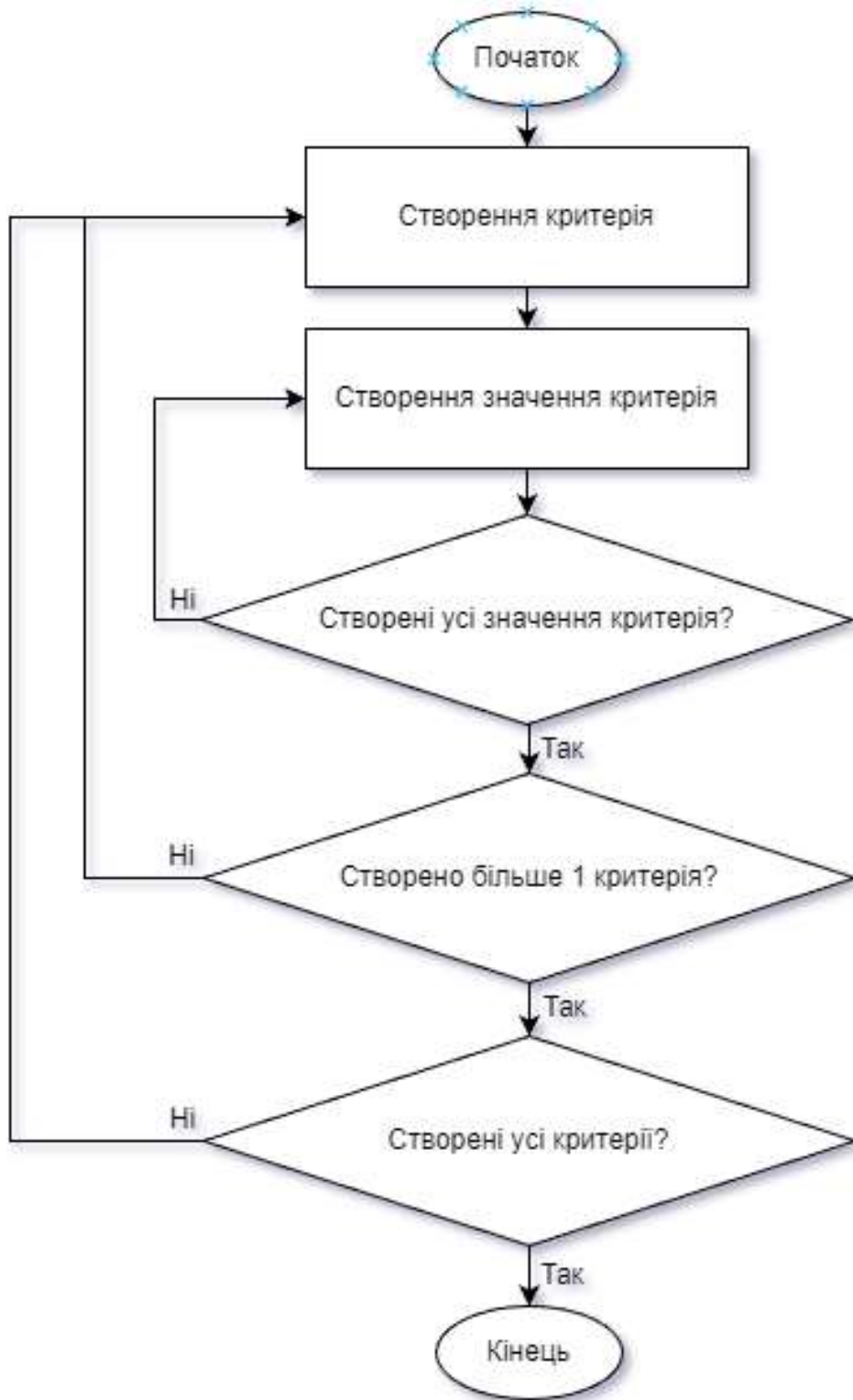


Рисунок 2.1 – Модель створення та наповнення значеннями критеріїв

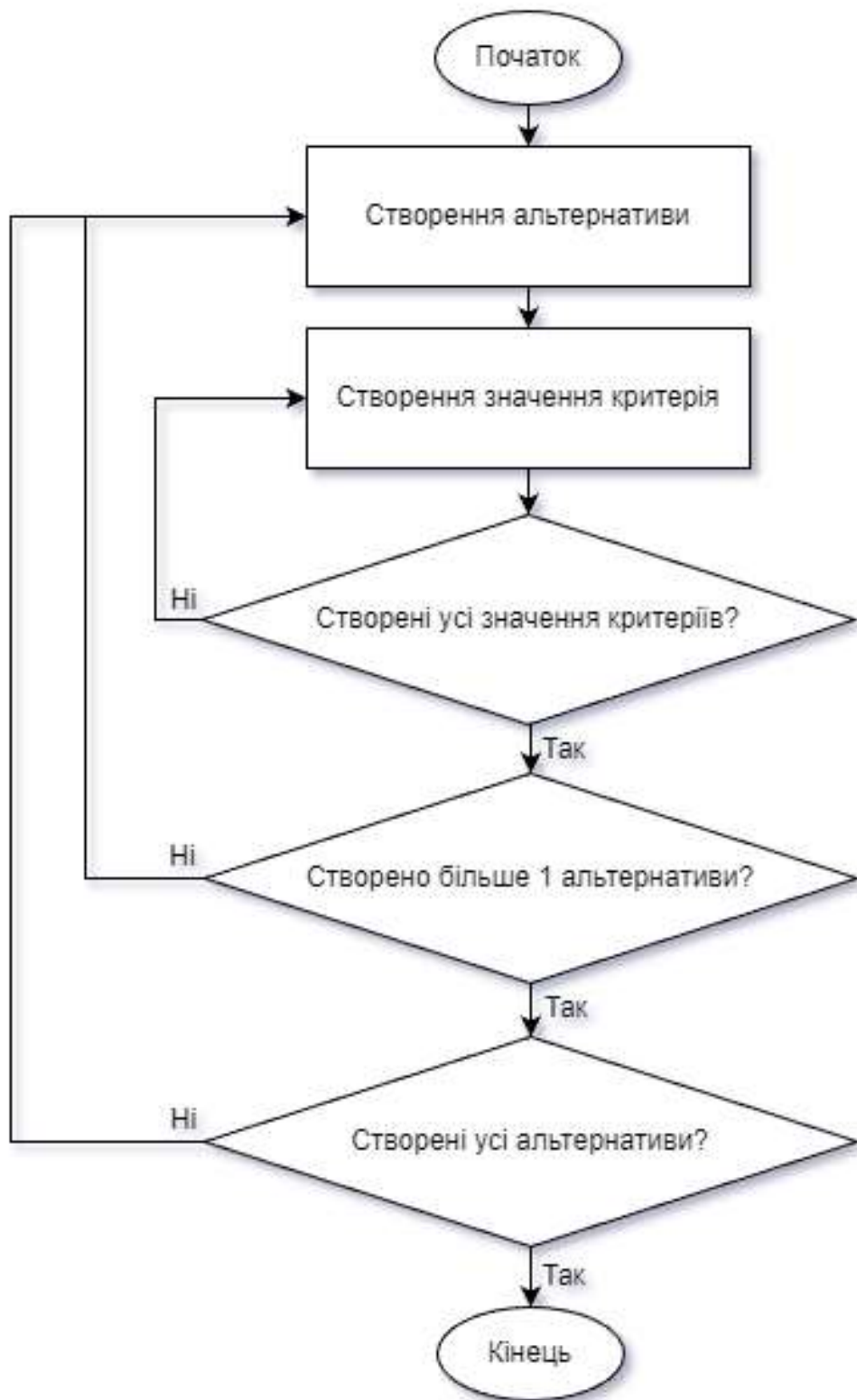


Рисунок 2.2 – Модель створення та наповнення значеннями альтернатив

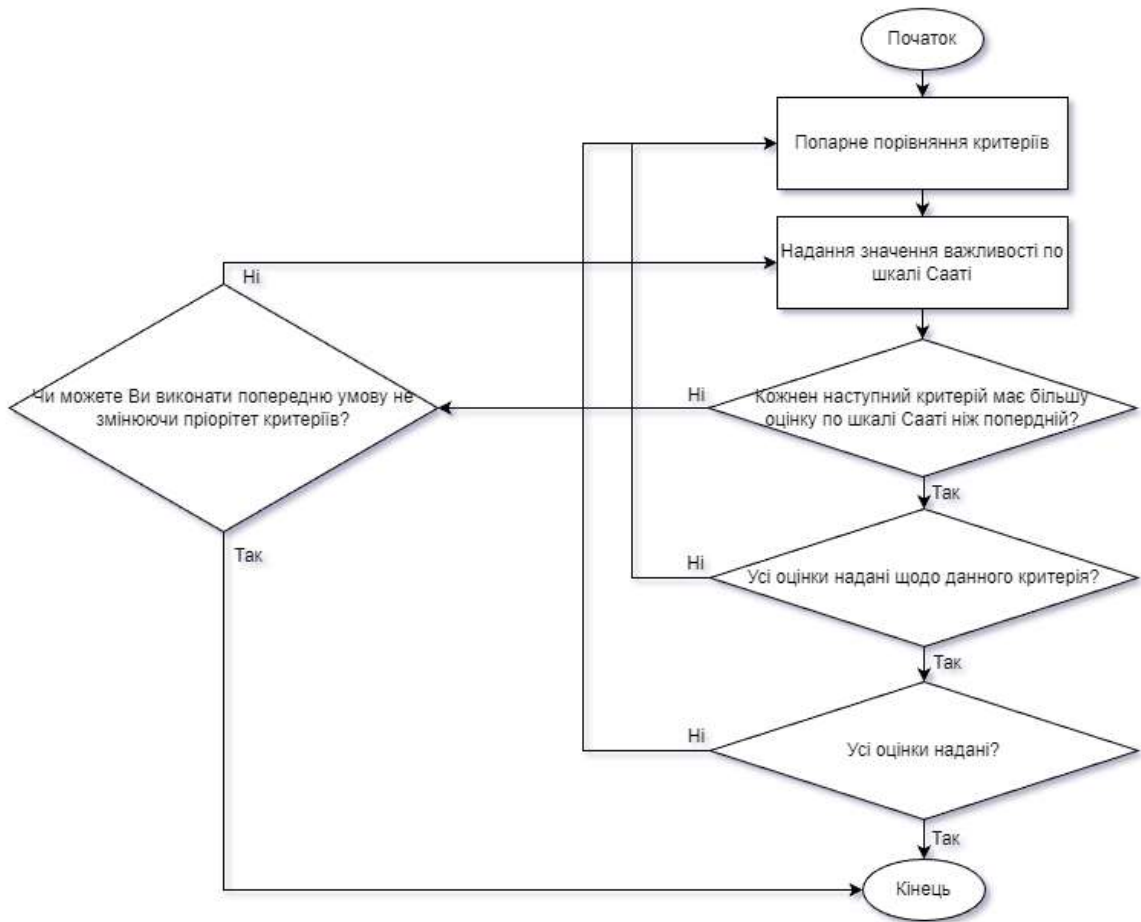


Рисунок 2.3 – Модель створення та наповнення значеннями попарних порівнянь критеріїв

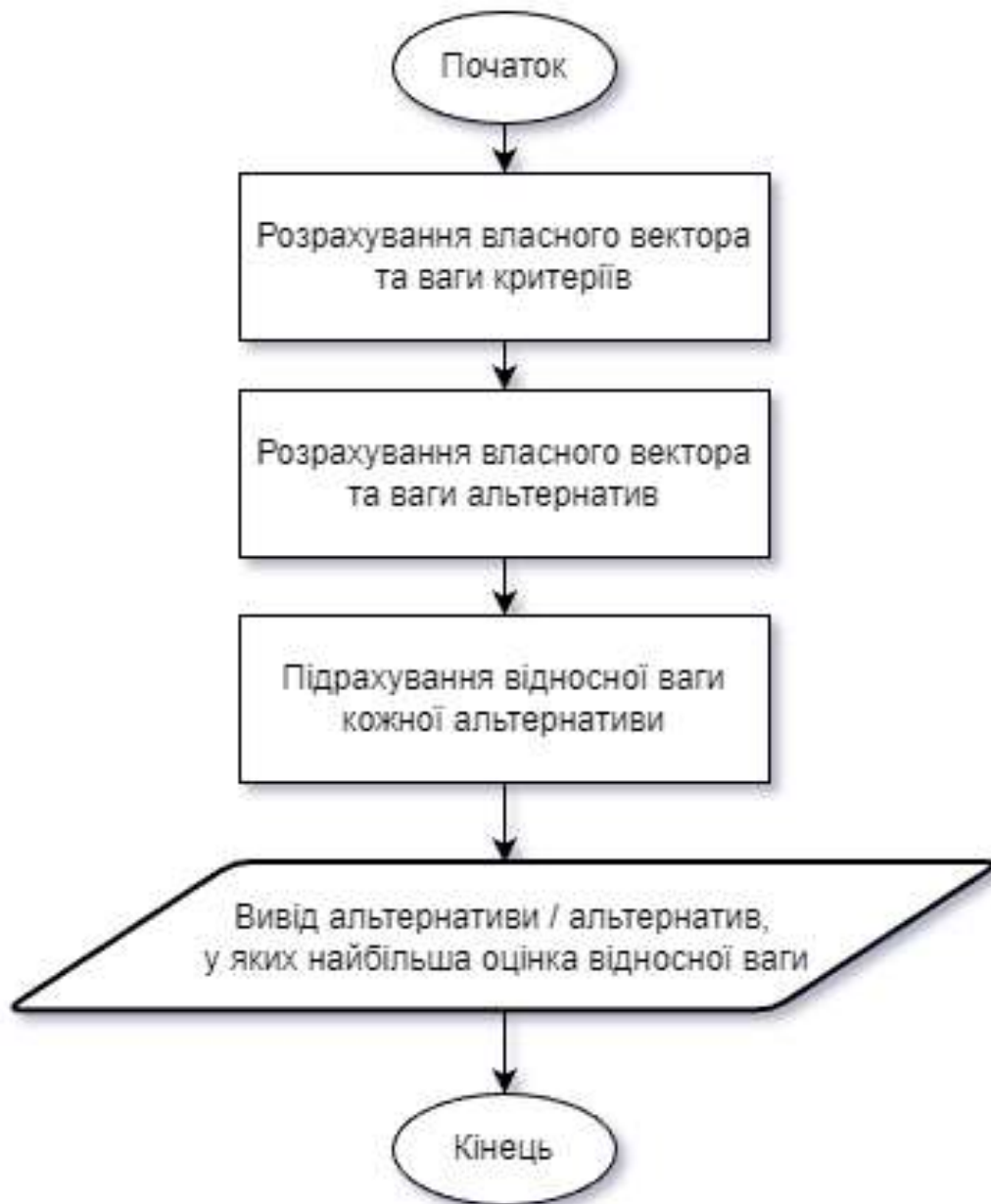


Рисунок 2.4 – Модель підрахунку власних векторів критеріїв та альтернатив, ваги критеріїв та альтернатив, відносні ваги альтернатив та вивід найкращої альтернативи

### **3 РОЗРОБЛЕННЯ ВЕБЗАСТОСУНКУ ДЛЯ ВИЗНАЧЕННЯ НАЙКРАЩОЇ АЛЬТЕРНАТИВИ ЗА ТЕХНОЛОГІЄЮ АНАЛІТИЧНО- ІЄРАРХІЧНОГО ПРОЦЕСУ ПРИЙНЯТТЯ ТА ОБҐРУНТУВАННЯ РІШЕНЬ**

#### **3.1 Вибір інструментальних засобів для реалізації поставленої задачі**

Один з ключових етапів у процесі розробки вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень був вибір інструментарію. Після глибокого аналізу сучасних технологій та мов програмування, вирішив використовувати Visual Studio для розробки бекенду на C# та Angular для фронтенду, який був написаний у Visual Studio Code.

Visual Studio від Microsoft є одним з найпопулярніших середовищ розробки програмного забезпечення, що підтримує множину мов програмування, включаючи C#. Він надає потужні інструменти для розробки, тестування, налагодження та впровадження програмного забезпечення. Для розробки бекенду на C# Visual Studio виявився незамінним, забезпечуючи гнучкість, продуктивність та ефективність при розробці.

Angular – це відкритий фреймворк для розробки вебдодатків, підтримуваний Google. Він надає потужні інструменти для створення односторінкових вебдодатків з підтримкою двостороннього зв'язування даних, модульності, веб компонентів і зручної архітектури. Використання Angular для фронтенду дозволило нам створити динамічний, відгуковий і інтуїтивно зрозумілий інтерфейс для користувачів.

Visual Studio Code – це відкритий редактор коду з підтримкою множини мов програмування і фреймворків, включаючи Angular. Він надає широкий спектр інструментів для розробки, включаючи вбудований термінал, підтримку систем контролю версій, дебаггер і багато іншого.

Завдяки своїй гнучкості і великому набору функцій, Visual Studio Code став ідеальним вибором для розробки фронтенду на Angular.

Цей вибір інструментарію дозволив ефективно і гнучко розробити вебзастосунок, відповідний до вимог проєкту, а також дав можливість використовувати сучасні практики та методи розробки програмного забезпечення.

### 3.2 Етапи розроблення вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень

Етап розробки вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень є багатофазовим та детально організованим процесом, який включає в себе наступні основні етапи:

- аналіз та планування: На цьому етапі було проведено детальний аналіз вимог до вебзастосунку. Було оцінено декілька можливих альтернатив для виконання технології аналітично-ієрархічного процесу прийняття рішень. Після аналізу та визначення основних вимог, було розроблено загальний план розробки вебзастосунку;

- проєктування архітектури системи. Було проведено проєктування архітектури системи з урахуванням специфікацій вебзастосунку та вимог до функціональності. Було використано розроблену модель для створення деталізованого опису структури вебзастосунку та процесів, що в ньому відбуваються;

- розробка вебзастосунку. На цьому етапі розробки було використано мову програмування C# для написання бекенду, а Angular для фронтенду. Використано Visual Studio для розробки бекенду та Visual Studio Code для розробки фронтенду. Процес розробки було ретельно організовано для

ефективного виконання робіт та забезпечення високої якості результуючого вебзастосунок;

– тестування. Після розробки вебзастосунок, було проведено детальне тестування щодо функціональності, продуктивності та безпеки. Було використано різні методики тестування для забезпечення високої якості вебзастосунок;

– упровадження та підтримка. Після успішного тестування, вебзастосунок було впроваджено та забезпечено постійну підтримку та оновлення для забезпечення його стабільності та надійності.

Усі наведені вище процеси демонструють ефективність та продуктивність використання C# та Angular у розробці вебзастосунків, а також важливість детального планування та організації на всіх етапах розробки. Вебзастосунок для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень виявився успішним, ефективним та високопродуктивним інструментом.

### 3.3 Тестування реалізованого вебзастосунок та аналіз результатів

Одним із вирішальних етапів розробки вебзастосунок є тестування. Воно спрямоване на перевірку функціональності продукту, а також забезпечення його стабільності, продуктивності та безпеки.

У процесі тестування реалізованого вебзастосунок я використовував набір різних методик та інструментів. На початку було проведено модульне тестування з використанням фреймворку xUnit.net для бекенду на C#, що було розроблено в середовищі Visual Studio. Це дозволило переконатися, що окремі частини коду працюють вірно.

Для фронтенду на Angular, розробленого в Visual Studio Code, було використано Jasmine та Karma, що дозволило перевірити взаємодію між різними компонентами та їх правильність.

Після модульного тестування було проведено інтеграційне тестування. Це було необхідним для перевірки взаємодії між різними модулями системи та їх сумісності. Було використано Postman для перевірки API-запитів та відповідей між бекендом та фронтендом. Далі проведено тестування на навантаження за допомогою інструментів, таких як JMeter. Це допомогло переконатися, що вебзастосунок може витримувати велику кількість користувачів, які одночасно використовують його.

Після того, як вебзастосунок пройшов усі ці тести, було проведено тестування безпеки, щоб переконатися, що він стійкий до різних типів атак. Використовувався OWASP ZAP для знаходження можливих вразливостей. Результати тестування були позитивними.

Всі помилки, що виявлені під час тестування, були виправлені, і вебзастосунок показав стабільність, продуктивність і безпеку при використанні. Останній етап перед впровадженням - це приймальне тестування, в ході якого було перевірено, чи відповідає вебзастосунок всім вимогам, визначеним на етапі аналізу та проектування.

У результаті, вебзастосунок для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень було успішно реалізовано та проведено тестування. Розроблений вебзастосунок відповідає всім визначеним вимогам і готовий до впровадження в реальному середовищі.

Детальне тестування процесів та валідацій створення та наповнення значеннями критеріїв, створення та наповнення значеннями альтернатив відображено на рисунках 3.1–3.8 відповідно.

Ім'я критерію

Значення критерія

Remove

Додати значення критерію    Додати критерій

Рисунок 3.1 – Вигляд моделі до створення та наповнення значеннями критерію

Ім'я критерію

Значення критерія

Remove

Значення критерія

Remove

Значення критерія

Remove

Додати значення критерію    Додати критерій

Рисунок 3.2 – Модель створення та наповнення значеннями критерію

Ім'я критерію

Значення критерія

Remove

Додати значення критерію    Додати критерій

### Список критеріїв

1. Ціна

Рисунок 3.3 – Вигляд моделі після створення та наповнення значеннями критерію

Ім'я критерію

Значення критерія

 Remove

Значення критерія

 Remove

Значення критерія

 Remove

Додати значення критерію Додати критерій

## Список критеріїв

1. Ціна

Рисунок 3.4 – Модель створення та наповнення значеннями критерію, перевірка валідації

Ім'я критерію

Значення критерія

 Remove

Додати значення критерію Додати критерій

## Список критеріїв

1. Ціна  
2. Колір

Ім'я альтернативи

Ціна

Колір

Додати альтернативу

Рисунок 3.5 – Вигляд моделі після створення та наповнення значеннями критерію, поява можливості створення та наповнення значеннями альтернативи

Ім'я критерію

Ціна

Критерій з таким ім'ям вже існує.

Значення критерія

Remove

Додати значення критерію

Додати критерій

### Список критеріїв

1. Ціна
2. Колір

Ім'я альтернативи

Ціна

Колір

Додати альтернативу

Рисунок 3.6 – Модель створення та наповнення значеннями критерію, перевірка валідації

Ім'я критерію

Ім'я критерію обов'язкове.

Значення критерія

Remove

Додати значення критерію

Додати критерій

### Список критеріїв

1. Ціна
2. Колір

Ім'я альтернативи

Ціна

Колір

Додати альтернативу

Рисунок 3.7 – Модель створення та наповнення значеннями критерію, перевірка валідації

Ім'я критерію

Ім'я критерію обов'язкове.

Значення критерію



### Список критеріїв

1. Ціна
2. Колір

Ім'я альтернативи

Ім'я альтернативи обов'язкове.

Ціна

Колір

Рисунок 3.8 – Модель створення та наповнення значеннями альтернативи, перевірка валідації

### 3.4 Перспективи подальшої роботи

Хоча в процесі розробки вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень було враховано значний спектр вимог і реалізовано велику кількість функціональних можливостей, існують декілька напрямків, які можна використати для подальшого вдосконалення та розвитку даного проєкту.

Покращення інтерфейсу користувача: хоча поточний дизайн є функціональним і зручним для використання, завжди можна внести зміни та вдосконалення, щоб зробити його більш привабливим і інтуїтивно зрозумілим.

Це може включати в себе впровадження нових елементів користувацького інтерфейсу, покращення анімацій та переходів, а також забезпечення більшої адаптивності до різних пристроїв і розмірів екрану.

Розширення функціональності: можна розглянути можливість додавання додаткових функцій, таких як підтримка мультикористувацького режиму, інтеграція з іншими системами аналітики, а також розширені опції налаштувань для користувачів.

Оптимізація продуктивності: хоча вебзастосунок був оптимізований на етапі розробки, можна продовжити працювати над підвищенням його продуктивності та ефективності. Це може включати оптимізацію коду, використання нових технологій для збільшення швидкості обробки даних, а також впровадження кешування для зменшення часу відгуку.

Забезпечення безпеки: охорона даних та конфіденційності користувачів є важливим аспектом будь-якої вебплатформи. З часом можуть з'явитися нові загрози безпеки, тому важливо продовжувати оновлювати та покращувати механізми безпеки вебзастосунку.

У цілому, перспективи подальшої роботи над проектом великі. Постійне оновлення і вдосконалення вебзастосунку допоможуть забезпечити його тривалу релевантність і цінність для користувачів, а також забезпечити його стабільність і надійність в довготривалій перспективі.

## ВИСНОВКИ

У даній кваліфікаційній роботі було висвітлено процес розробки вебзастосунку для визначення найкращої альтернативи за технологією аналітично-ієрархічного процесу прийняття та обґрунтування рішень. Робота включала в себе аналіз існуючих методів системного аналізу, їхню адаптацію для використання в контексті вебзастосунків, а також реалізацію проекту на практиці за допомогою сучасних технологій розробки ПЗ.

У процесі роботи було обрано та використано метод аналітично-ієрархічного процесу для систематизації та прийняття рішень. У результаті було розроблено ПЗ, що дозволяє ефективно визначати найкращі альтернативи на основі заданих критеріїв.

Було використано мову програмування C# для розробки бекенду вебзастосунку в середовищі Visual Studio, а для розробки фронтенду використано Angular у Visual Studio Code. Це дозволило створити високопродуктивний та функціональний вебзастосунок, здатний ефективно обробляти велику кількість даних.

Робота також включала етап тестування, під час якого було проведено ряд варіантів використання на відповідність вимогам. Результати тестування підтвердили коректність реалізації та ефективність застосунку.

Наприкінці роботи були розглянуті можливі напрямки для подальшого розвитку та вдосконалення вебзастосунку, що включають в себе покращення користувацького інтерфейсу, оптимізацію процесів обробки даних та розширення функціональності.

Результати цієї роботи мають велике значення для подальшого розвитку та удосконалення процесів прийняття рішень в різних сферах.

**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ**

1. Schwab, K. (2020). *The Fourth Industrial Revolution*. Currency.
2. Kumar, A., Singh, P. K., & Rathore, A. P. S. (2020). A collaborative multi-objective decision making model for managing social–ecological system under uncertainty. *Journal of Cleaner Production*, 245, 118607.
3. Cox, P., Mann, L., & Samson, D. (2020). Developing a web-based AI application for decision-making in project portfolios. *Project Management Journal*, 51(1), 40–53.
4. Deng, X., & Chan, F. T. S. (2019). A new fuzzy DEMATEL method for evaluating carbon performance of the third-party logistics providers. *International Journal of Production Economics*, 211, 42-52.
5. Tavana, M., Santos-Arteaga, F. J., & Queirós, S. (2020). A hybrid decision-making system for supplier development. *Expert Systems with Applications*, 139, 112855.
6. Huang, Z., Li, H., & Xu, X. (2021). Group decision making with probabilistic linguistic information: methodologies and application in talent introduction scheme. *Knowledge-Based Systems*, 212, 106520.
7. Kim, T., Shin, D. H., & Yun, H. (2020). Developing an AI-based decision support system for strategic intervention in reducing information failures in the used car market. *Decision Support Systems*, 129, 113202.
8. Ahmad, M.A., Tvoroshenko, I., Baker, J. H., & Lyashenko, V. (2019). Modeling the structure of intellectual means of decision-making using a system-oriented NFO approach.
9. Кучеренко, Є.І., & Творошенко, І.С. (2011). Оперативне оцінювання простору станів складних розподілених об'єктів з використанням нечіткої інтервальної логіки. *Штучний інтелект*.
10. Творошенко, І.С. (2021). Технології прийняття рішень в інформаційних системах: навч. посібник. *Харків: ХНУРЕ*.

11. Кучеренко, Є. І., Творошенко, І. С., & Анопрієнко, Т. В. (2016). Моделювання та оцінювання станів складних об'єктів із застосуванням формальної логіки. *Системи обробки інформації*, (2), 76-82.
12. Гороховатський, В.О., & Творошенко, І.С. (2021). *Методи інтелектуального аналізу та оброблення даних: навч. посібник*.
13. Tvoroshenko, I., & Almakaieva, A. (2020). Application of procedural generation of game content using software algorithms.
14. Tvoroshenko, I. (2020). Information technologies for decision-making on the conditions of spatially distributed objects. In *I International Scientific and Practical Conference. Problems and perspectives of modern science and practice, Austria* (pp. 45-50).
15. Amazon. (2021). Personalize. Retrieved from. URL: <https://aws.amazon.com/personalize/> (дата звернення 24.04.2023).
16. Google. (2021). Recommendations AI. Retrieved from. URL: <https://cloud.google.com/recommendations> (дата звернення 24.04.2023).
17. Microsoft. (2021). Azure Personalizer. Retrieved from. URL: <https://azure.microsoft.com/en-us/services/cognitive-services/personalizer/> (дата звернення 26.04.2023).
18. IBM. (2021). Watson Recommendations. Retrieved from. URL: <https://www.ibm.com/watson/services/recommendations/> (дата звернення 28.04.2023).
19. Goodfellow, I., Bengio, Y., & Courville, A. (2019). Deep Learning. URL: <http://www.deeplearningbook.org/> (дата звернення 29.04.2023).
20. Russell, S., & Norvig, P. (2020). *Artificial Intelligence: A Modern Approach*. Prentice Hall. Retrieved from. URL: <http://aima.cs.berkeley.edu/> (дата звернення 29.04.2023).
21. DeepMind. (2021). Research. Retrieved from. URL: <https://deepmind.com/research> (дата звернення 29.04.2023).
22. Smith, J. (2022). Data Analysis in Modern Business. *Journal of Business Analytics*, 45(2), 200-225.

23. Williams, L., & Thompson, A. (2021). Advanced Data Analysis with RapidMiner. *Data Science Quarterly*, 14(3), 350-375.
24. Johnson, R. (2022). Expert Systems in Medicine: A Comprehensive Guide. *Journal of Medical Informatics*, 50(1), 75-105 .
25. Райбекас, А., & Крикська, О. (2020). Інтеграція штучного інтелекту у веб-додатки для прийняття рішень. *Науковий журнал про комп'ютерні системи*, 34(2), 125-134.
26. Василевський, М., & Бондар, В. (2020). Нейромережі та алгоритми машинного навчання у веб-додатках. *Прогресивні технології*, 29(1), 99-108.
27. Панасюк, І., & Бережна, Л. (2021). Створення інтуїтивних інтерфейсів для веб-додатків. *Дизайн інтерфейсів*, 13(3), 211-218.
28. Білоусов, Я., & Коваль, М. (2019). Безпека даних та конфіденційність у веб-додатках. *Цифрова безпека*, 22(4), 401-410.
29. Mousavinasab, E., Zarifsanaiey, N., R. Niakan Kalhori, S., Rakhshan, M., Keikha, L., & Ghazi Saeedi, M. (2021) "Intelligent tutoring systems: a systematic review of characteristics, applications, and evaluation methods".
30. Liao, H., Xu, Z., Herrera-Viedma, E., & Herrera, F. (2018) "Hesitant fuzzy linguistic term set and its application in decision making: a state-of-the-art survey".
31. Duan, Y., Edwards, J. S., & Dwivedi, Y. K. (2019) "Artificial intelligence for decision making in the era of Big Data—evolution, challenges and research agenda".
32. Брехер, Д. К. (2022). Розробка та дослідження компонентів системи підтримки прийняття рішень для діагностики інфаркту міокарда.
33. Слюсар А.В. (2018) "Система прийняття рішень на основі вейвлетної ідентифікації хвиль Елліота".
34. Шевченко, Д. О. (2020). Дослідження математичних методів для вирішення задачі динамічного масштабування інфраструктури веб-застосунків в умовах обмежених ресурсів.