

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ**

**Інститут (факультет) автоматизації і комп'ютерних систем імені проф. І.В. Ельперіна  
Кафедра інформаційних технологій, штучного інтелекту і кібербезпеки**

**«До захисту в ЕК»**  
Директор інституту(декан факультету)  
\_\_\_\_\_ Андрій ФОРСЮК  
(підпис) (ім'я та прізвище)

«08» грудня 2025р.

**«До захисту допущено»**  
Завідувач кафедри  
\_\_\_\_\_ Сергій ГРИБКОВ  
(підпис) (ім'я та прізвище)

«08» грудня 2025р.

**КВАЛІФІКАЦІЙНА РОБОТА  
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ МАГІСТРА**

зі спеціальності 122 Комп'ютерні науки

(код і назва спеціальності)

освітньо-професійної програми Управління інформацією та аналітика даних  
на тему: Інтелектуальна система підбору електромобілів для клієнтів автосалону

Виконав: здобувач 2 курсу, групи КН-2-2М

\_\_\_\_\_ Ковальчук Назар Васильович  
(прізвище, ім'я та по батькові повністю) (підпис)

Керівник \_\_\_\_\_ Мошенський Андрій Олександрович  
(прізвище, ім'я та по батькові повністю) (підпис)

Консультанти \_\_\_\_\_  
(ім'я та прізвище) (підпис)

\_\_\_\_\_ (ім'я та прізвище) (підпис)

\_\_\_\_\_ (ім'я та прізвище) (підпис)

Рецензент \_\_\_\_\_  
(ім'я та прізвище) (підпис)

*Я як здобувач(ка) Національного університету харчових технологій розумію і підтримую політику університету з академічної доброчесності. Я не надавав(-ла) і не одержував(-ла) недозволеної допомоги під час підготовки цієї роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело*

Здобувач \_\_\_\_\_  
(підпис)

Київ - 2025р.

# НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) Автоматизації і комп'ютерних систем

Кафедра інформаційних технологій, штучного інтелекту і кібербезпеки

Освітній ступінь магістр

Спеціальність 122 Комп'ютерні науки

(код і назва)

**Освітньо-професійна програма** Управління інформацією та аналітика даних

**ЗАТВЕРДЖУЮ**

Завідувач кафедри інформаційних технологій, штучного інтелекту і кібербезпеки

**Сергій ГРИБКОВ**

«05» листопада 2025 року

## ЗАВДАННЯ

### НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Ковальчука Назара Васильовича

(прізвище, ім'я, по батькові)

1. Тема роботи Інтелектуальна система підбору електромобілів для клієнтів автосалону

керівник роботи Мошенський Андрій Олександрович доцент, кан тех наук

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від «05» листопада 2025 року №906-кв

2. Строк подання здобувачем роботи: 01 грудня 2025 року

3. Вихідні дані до роботи:

1. Табличний файл із вхідними даними (формат CSV).

2. Перелік критеріїв оцінювання та їх типи (виграшні/витратні).

3. Методики багатокритеріального аналізу (WSM, TOPSIS).

4. Програмні інструменти для реалізації DSS-системи (Python, Streamlit).

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити):

1. Аналіз предметної області та особливостей вибору електромобілів.

2. Огляд наявних інформаційних систем і методів для рекомендацій у автомобільній сфері.

3. Формалізація задачі підбору електромобіля та визначення ключових критеріїв.

4. Математичне моделювання задачі багатокритеріального аналізу альтернатив.

5. Аналіз і порівняння методів оцінювання (WSM, TOPSIS).

6. Обґрунтування вибору технологій, інструментів і програмних рішень.
7. Проектування архітектури DSS-системи.
8. Реалізація алгоритмів багатокритеріального оцінювання.
9. Тестування, апробація системи та аналіз результатів.
5. Перелік графічного матеріалу:
  1. Візуалізація структури вхідного набору даних.
  2. Скріншоти інтерфейсу DSS-системи
6. Консультанти розділів роботи:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1.	Мошенський А.О., доцент, кандидат технічних наук	10.11.2025	18.11.2025
2.	Мошенський А.О., доцент, кандидат технічних наук	18.11.2025	20.11.2025
3.	Мошенський А.О., доцент, кандидат технічних наук	20.11.2025	24.11.2025

7. Дата видачі завдання: 01 жовтня 2025 року

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Видача завдання	01.11.2025	Виконав
2	Виконання пошуку матеріалів	04.11.2025	Виконав
3	Оформлення розділу 1	11.11.2025	Виконав
4	Оформлення розділу 2	14.11.2025	Виконав
5	Розроблення системи	17.11.2025	Виконав
6	Оформлення розділу 3	20.11.2025	Виконав
7	Оформлення автореферату	24.11.2025	Виконав
8	Оформлення презентації	26.11.2025	Виконав

Здобувач

\_\_\_\_\_

(підпис)

Назар КОВАЛЬЧУК

\_\_\_\_\_

(прізвище та ініціали)

Керівник роботи

\_\_\_\_\_

(підпис)

Андрій МОШЕНСЬКИЙ

\_\_\_\_\_

(прізвище та ініціали)

## АНОТАЦІЯ

Кваліфікаційна робота магістра на тему «Інтелектуальна система підбору електромобілів для клієнтів автосалону» складається зі вступу, трьох розділів, висновків, списку використаних джерел і додатків.

У роботі досліджено методи підтримки прийняття рішень та багатокритеріального оцінювання альтернатив, зокрема WSM і TOPSIS. Розроблено інтелектуальну DSS-систему, що автоматизує підбір електромобіля за технічними й економічними параметрами клієнта автосалону.

Система реалізована у вигляді вебзастосунку на основі Streamlit з використанням бібліотек Pandas, NumPy, Plotly та Scikit-learn. Вона підтримує налаштування ваг критеріїв і містить модуль пояснюваності рішень, який візуалізує внесок кожного фактора у загальний рейтинг.

Практичне значення роботи полягає у можливості використання системи для персоналізованого підбору електромобілів, скорочення часу вибору та підвищення ефективності продажів у автосалонах.

**Ключові слова:** ІНТЕЛЕКТУАЛЬНА СИСТЕМА, DSS, ЕЛЕКТРОМОБІЛЬ, РЕКОМЕНДАЦІЙНА СИСТЕМА, WSM, TOPSIS, STREAMLIT, АНАЛІТИКА ДАНИХ.

## SUMMARY

The master's qualification work titled "Intelligent System for Selecting Electric Vehicles for Car Dealership Clients" consists of an introduction, three chapters, conclusions, a list of references, and appendices.

The work investigates decision support and multi-criteria evaluation methods, particularly WSM and TOPSIS. An intelligent decision support system (DSS) was developed to automate the selection of electric vehicles based on customers' technical and economic preferences.

The system is implemented as a web application using Streamlit, with Pandas, NumPy, Plotly, and Scikit-learn libraries. It allows flexible weighting of criteria and includes an explainability module visualizing each factor's contribution to the final score.

Practical significance: the developed system can be used for personalized vehicle selection, reducing decision-making time and improving the efficiency of sales in car dealerships.

**Keywords:** INTELLIGENT SYSTEM, DSS, ELECTRIC VEHICLE, RECOMMENDER SYSTEM, WSM, TOPSIS, STREAMLIT, DATA ANALYTICS.

## ЗМІСТ

<u>ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ</u> .....	9
<u>ВСТУП</u> .....	10
<u>РОЗДІЛ 1. Дослідження предметної області та постановка задачі</u> .....	14
<u>1.1 Загальна характеристика предметної області</u> .....	14
<u>1.2. Аналіз існуючих інформаційних систем у сфері підбору електромобілів</u>	17
<u>1.3. Функціональне моделювання та аналіз існуючих процесів підбору електромобілів</u> .....	20
<u>1.4. Виявлені задачі та проблеми під час аналізу процесів підбору електромобілів</u> .....	24
<u>1.5. Аналітичний огляд літератури</u> .....	27
<u>1.6. Постановка завдання на дослідження</u> .....	30
<u>1.7. Висновки до розділу 1</u> .....	34
<u>РОЗДІЛ 2. Дослідження та обґрунтування технологій, методів, алгоритмів</u> .....	36
<u>2.1. Моделювання основних задач</u> .....	36
<u>2.2. Огляд та аналіз існуючих рішень для розв’язання виявлених задач</u> .....	39
<u>2.3. Обґрунтування вибору технологій, методів, алгоритмів</u> .....	42
<u>2.4. Висновки до розділу 2</u> .....	46
<u>РОЗДІЛ 3. Розроблення та апробація аналітичної моделі</u> .....	48
<u>3.1. Архітектура та загальна структура DSS-системи</u> .....	48
<u>3.2. Розроблення модулів завантаження, обробки та нормалізації даних</u> .....	51
<u>3.3. Реалізація алгоритмів багатокритеріального оцінювання альтернатив</u> .....	55
<u>3.4. Реалізація інтерфейсу та візуалізації результатів</u> .....	60
<u>3.5 Висновки до розділу 3</u> .....	63
<u>ВИСНОВКИ</u> .....	65

<u>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ</u> .....	68
<u>ДОДАТКИ</u> .....	72
<u>Додаток А. Структура вхідного набору даних</u> .....	72
<u>Додаток Б. Перелік програмних модулів DSS-системи</u> .....	73
<u>Додаток В. Інтерфейс інтелектуальної DSS-системи підбору електромобілів</u> .....	110

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ**

<b>Скорочення</b>	<b>Розшифрування</b>
<b>DSS</b>	Decision Support System – система підтримки прийняття рішень
<b>EV</b>	Electric Vehicle – електромобіль
<b>WSM</b>	Weighted Sum Model – модель зваженої суми
<b>TOPSIS</b>	Technique for Order Preference by Similarity to Ideal Solution – метод подібності до ідеальної точки
<b>MCDM</b>	Multi-Criteria Decision Making – багатокритеріальне прийняття рішень

## ВСТУП

**Актуальність теми.** Стрімкий розвиток ринку електромобілів та цифрових технологій впливає на трансформацію підходів до вибору транспортних засобів, що зумовлює зростання попиту на інтелектуальні системи підтримки прийняття рішень. Урахування великої кількості параметрів, таких як запас ходу, час заряджання, енергоефективність, вартість володіння та технічні аспекти, потребує застосування методів багатокритеріального аналізу, серед яких вагова сума (WSM), метод аналізу ієрархій (AHP) та метод TOPSIS. Сучасні тенденції цифрової аналітики, що включають автоматизацію прийняття рішень і використання великих масивів даних, підтверджують актуальність розроблення DSS-рішень для автосалонів та споживачів. Урахування об'єктивних критеріїв, технологічних параметрів, ринкових даних та індивідуальних уподобань користувача робить проблему побудови інтелектуальної системи підбору електромобілів важливою й практично значущою.

**Зв'язок роботи з науковими програмами, планами, темами.** Кваліфікаційну роботу виконано відповідно до плану науково-дослідних робіт кафедри інформаційних технологій, штучного інтелекту і кібербезпеки Національного університету харчових технологій та пов'язано з тематичними НДР кафедри: НДР «Дослідження та використання сучасних інформаційних технологій для виконання функцій та завдань виробничого і організаційного управління підприємств харчової галузі» № ДР 0120U105386, 2020–2025 рр.; НДР «Дослідження та використання засобів штучного інтелекту для розв'язання управлінських завдань в харчовій промисловості» № ДР 0125U003887, 2025–2030 рр. Тематика кваліфікаційної роботи узгоджується з науковими напрямками кафедри у сфері інтелектуальних систем, багатокритеріального аналізу та інформаційних технологій підтримки прийняття рішень.

**Мета дослідження.** Розробити інтелектуальну систему підтримки прийняття рішень для підбору електромобілів, яка забезпечує

багатокритеріальний аналіз, оцінювання та рекомендації моделей з урахуванням індивідуальних уподобань користувача.

**Завдання дослідження:**

- проаналізувати предметну область підбору електромобілів та визначити ключові критерії вибору;
- дослідити сучасні методи багатокритеріальної оцінки альтернатив, включаючи WSM, АНР та TOPSIS;
- сформуванати математичну модель оцінювання електромобілів;
- розробити архітектуру DSS-системи;
- реалізувати програмну систему з підтримкою завантаження даних, фільтрації, оцінювання та пояснення рішень;
- провести тестування модулів і перевірити коректність роботи алгоритмів;
- створити інтерфейс користувача та забезпечити візуалізацію результатів;
- підготувати експериментальні матеріали та пояснення прийнятих рішень.

**Об'єкт дослідження.** Процес прийняття рішень при виборі електромобілів клієнтами автосалону.

**Предмет дослідження.** Методи, моделі та програмні засоби інтелектуального підбору електромобілів із застосуванням багатокритеріальних алгоритмів аналізу альтернатив.

**Методи дослідження:**

- Аналіз і синтез – для вивчення наукових джерел з інтелектуальних систем підтримки прийняття рішень, дослідження моделей рекомендаційних систем та методів багатокритеріальної оцінки альтернатив, включаючи WSM і TOPSIS, а також для формування цілісного бачення предметної області.

- Індукція та дедукція – для формування логічних припущень щодо впливу технічних, економічних і користувацьких критеріїв на результат підбору електромобіля та обґрунтування вибору алгоритмів оцінювання.
- Класифікація та узагальнення – для систематизації характеристик електромобілів (енергоспоживання, динаміка, зарядна потужність, тип кузова тощо), структурування даних і визначення основних критеріїв відбору.
- Статистичні методи аналізу – для перевірки якості даних, оцінки розподілів значень технічних і економічних показників, виявлення кореляцій між параметрами (ціна–пробіг, потужність–ефективність) та визначення інформативності характеристик.
- Методи оптимізації та математичного моделювання – для побудови моделі багатокритеріальної оцінки альтернатив, визначення цільової функції та вагових коефіцієнтів, реалізації алгоритмів WSM і TOPSIS у модулі DSS.
- Методи машинного навчання – для автоматизації вибору вагових коефіцієнтів, виявлення закономірностей у наборі технічних характеристик та підвищення точності рекомендацій.
- Методи нормалізації даних – для приведення різнорідних параметрів до єдиного масштабу, усунення пропусків і забезпечення коректності обчислень у процесі оцінювання альтернатив.
- Візуалізація даних – для графічного представлення результатів аналізу, побудови діаграм внеску критеріїв, порівняння альтернативних моделей та підвищення інтерпретованості рекомендацій.
- Моделювання та експеримент – для розроблення, тестування та верифікації DSS-системи на реальних даних електромобілів 2025 року, проведення експериментів із різними ваговими сценаріями та оцінки ефективності роботи модулів системи.

**Наукова новизна одержаних результатів.** Наукова новизна полягає в розробленні комбінованої моделі багатокритеріального оцінювання електромобілів, що поєднує WSM, АНР і TOPSIS для покращення точності й обґрунтованості вибору. Запропонована система забезпечує автоматизацію аналізу технічних параметрів, адаптацію вагових коефіцієнтів та інтерактивне пояснення прийнятого рішення.

**Практичне значення одержаних результатів.** Результати дослідження можуть бути використані автосалонами, онлайн-платформами, консалтинговими компаніями та приватними користувачами для об'єктивного вибору електромобіля. Система здатна скорочувати час прийняття рішення, підвищувати точність підбору та забезпечувати прозорість оцінювання.

**Особистий внесок здобувача.** Проведено аналіз предметної області, сформовано набір критеріїв, зібрано та підготовлено вхідні дані, розроблено програмну систему, реалізовано алгоритми WSM, АНР та TOPSIS, виконано тестування, побудовано інтерфейс користувача та сформовано експериментальні результати.

**Структура та обсяг роботи.** Кваліфікаційна робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків. Повний обсяг роботи становить 113 сторінок, включаючи 10 рисунків та 3 додатки. Список використаних джерел містить 30 найменування.

## РОЗДІЛ 1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

### 1.1 Загальна характеристика предметної області

Ринок електромобілів упродовж останнього десятиліття демонструє стале зростання, що зумовлено низкою глобальних технологічних, економічних і соціальних тенденцій[1]. Його розвиток відбувається в контексті переорієнтації світової транспортної системи на екологічно безпечні та енергоефективні рішення, а також у рамках загального тренду декарбонізації економіки[2,3,4]. Посилення державних екологічних стандартів, зростання вимог до зниження рівня викидів, активний розвиток відновлюваних джерел енергії та удосконалення акумуляторних технологій формують умови, за яких електромобільність стає ключовим напрямом модернізації транспортної галузі.

Електромобілі вже давно перестали бути нішевим продуктом, призначеним для ентузіастів або обмеженого кола споживачів. Завдяки підвищенню доступності, поліпшенню технічних показників і розширенню модельного ряду вони перетворилися на конкурентоспроможну альтернативу автомобілям із двигунами внутрішнього згоряння. Сучасні моделі характеризуються зростаючим запасом ходу, оптимізованим енергоспоживанням і здатністю підтримувати високі швидкості заряджання. Водночас зменшуються витрати на сервісне обслуговування, оскільки електротяга передбачає значно меншу кількість зношуваних вузлів. Сукупність цих чинників підсилює економічну доцільність придбання електромобілів і сприяє трансформації структури попиту на світових автомобільних ринках.

Предметна область підбору електромобіля характеризується складністю, багатофакторністю та значним обсягом параметрів, які необхідно враховувати під час прийняття рішення. У процесі вибору зазвичай аналізуються технічні, економічні та експлуатаційні характеристики транспортного засобу. До технічних параметрів належать запас ходу, ємність і тип тягової батареї, енергетична ефективність силової установки, динамічні можливості, конструктивні

особливості приводу, пропускну здатність зарядного інтерфейсу та час заряджання до практично корисного рівня. Економічні характеристики охоплюють не лише початкову вартість автомобіля, а й загальну вартість володіння, яка включає витрати на заряджання, амортизацію, технічне обслуговування та страхування. Експлуатаційні показники пов'язані з рівнем комфорту, надійністю, місткістю салону та багажного відділення, ступенем інтегрованості інформаційних і телематичних технологій, а також загальним користувацьким досвідом.

Важливе місце у структурі предметної області займає зарядна інфраструктура[5]. Рівень її розвитку безпосередньо впливає на комфорт і передбачуваність користування електромобілем. У країнах із розгалуженими мережами швидкісних станцій постійного струму, домашніх та міських пунктів заряджання електромобілів сприймається як повноцінна альтернатива традиційним транспортним засобам. Натомість у регіонах із недостатньою інфраструктурною підтримкою відчутним стає так зване «зарядне занепокоєння», яке знижує готовність споживачів переходити на електротранспорт. Саме тому під час порівняння різних моделей важливо враховувати не лише їхні технічні та економічні характеристики, а й доступність зарядних станцій відповідного стандарту.

Розмаїття автомобільних моделей є ще одним чинником, що ускладнює процес вибору. Сьогодні на ринку представлені електромобілі різних класів — від компактних міських хетчбеків до великих позашляховиків і преміальних спортивних моделей. Кожен сегмент має власні технічні та експлуатаційні особливості, а також орієнтується на різні групи споживачів. У таких умовах визначення оптимальної моделі потребує комплексного підходу, який враховує сукупність критеріїв, що інколи знаходяться у суперечності між собою. Наприклад, збільшення ємності акумулятора позитивно впливає на запас ходу, але водночас підвищує масу автомобіля та його вартість, що може погіршувати динамічні або економічні показники. Така багатовимірність предметної області

створює передумови для використання інструментів, здатних систематизувати дані та забезпечити об'єктивну оцінку альтернатив.

У цьому контексті важливу роль відіграють системи підтримки прийняття рішень (DSS). Інтелектуальні DSS-системи застосовують методи обробки даних, фільтрації параметрів, нормування показників та алгоритми багатокритеріального оцінювання для визначення найкращої альтернативи відповідно до потреб користувача[6]. На відміну від традиційних підходів, що передбачають ручне порівняння характеристик, DSS-системи дозволяють враховувати значно більшу кількість параметрів і забезпечувати прозорий, стандартизований процес оцінювання. Використання математичних моделей, зокрема методів WSM[7], TOPSIS[8] та інших підходів багатокритеріального аналізу, підвищує об'єктивність результатів і дає змогу оцінювати електромобілі не лише за окремими характеристиками, а й за їхнім комплексним впливом на загальну відповідність критеріям вибору.

Сучасні DSS-рішення також виконують функцію зменшення інформаційного навантаження на користувача. Оскільки обсяг доступної інформації про характеристики електромобілів постійно збільшується, а кількість моделей щороку зростає, виникає потреба у системах, здатних агрегувати ці дані, інтерпретувати їх і формувати рекомендації на основі заданих вимог. Застосування таких систем підвищує точність і швидкість прийняття рішення, а також зменшує ризик вибору нераціонального варіанта через неповний або помилковий аналіз.

Таким чином, предметна область підбору електромобілів формується на перетині кількох ключових компонентів: технічних можливостей сучасного електротранспорту, економічних аспектів володіння, рівня розвитку зарядної інфраструктури та методологічних підходів до багатокритеріального аналізу. Її складна структура та висока варіативність параметрів обґрунтовують необхідність застосування інтелектуальних інформаційних технологій, що дозволяють здійснювати комплексну оцінку альтернатив та автоматизувати процес прийняття рішень. Такий підхід забезпечує більшу точність, об'єктивність і ефективність

порівняння моделей, що є особливо важливим в умовах стрімкого розвитку електромобільності та постійного оновлення ринку.

## **1.2. Аналіз існуючих інформаційних систем у сфері підбору електромобілів**

Розвиток цифрових платформ, які забезпечують пошук, порівняння та попередній аналіз електромобілів, є закономірним результатом зростання ринку електротранспорту та збільшення складності самих моделей. З огляду на велику кількість параметрів, що визначають технічні, економічні та експлуатаційні властивості електромобіля, виникла потреба у появі спеціалізованих інформаційних систем, здатних структурувати ці дані та подавати їх у зручному форматі. На сьогодні на ринку представлено широкий спектр таких платформ, які охоплюють як глобальні каталоги та порівняльні онлайн-ресурси, так і комерційні конфігуратори, мобільні додатки та локальні національні сервіси. Незважаючи на їхню кількість, рівень функціональної повноти та аналітичних можливостей цих рішень є нерівномірним, а їхня ефективність обмежена специфікою реалізації.

Більшість існуючих платформ орієнтовані на базове подання технічних характеристик без глибоких аналітичних інструментів[9,10]. Такі системи здебільшого виконують функції електронного довідника або фільтра за кількома ключовими параметрами, серед яких найчастіше фігурують ціна, запас ходу, тип кузова, рік випуску та потужність зарядки. Структура подання інформації є статичною та не передбачає механізмів інтегрованого оцінювання, що значно обмежує корисність таких платформ у процесі прийняття рішення. Користувач отримує перелік автомобілів, відсортований за одним обраним параметром, але не має можливості одночасно враховувати десяток чи більше критеріїв, що є характерним для реального процесу вибору електромобіля.

Однією з фундаментальних проблем більшості платформ є відсутність комплексності. Системи, які функціонують на основі простих фільтрів, не

застосовують математичні методи багатокритеріальної оптимізації, що унеможлиблює отримання інтегрального оцінювання моделі. За таких умов процес вибору електромобіля повністю переноситься на користувача, який змушений самотійно зіставляти різні характеристики, зважувати їх за важливістю та узагальнювати результати. Це значно підвищує когнітивне навантаження та спричиняє ризик суб'єктивних помилок, адже людина обмежена у здатності одночасно аналізувати великі обсяги різнорідних числових даних. Більш того, фільтрація за одним параметром майже ніколи не дає користувачу «оптимальну» відповідь, оскільки електромобіль є складним технічним об'єктом, а його якість визначається комбінацією параметрів, які нерідко мають суперечливий характер.

Ще одним суттєвим недоліком існуючих платформ є відсутність персоналізації. Переважна більшість систем подає дані у вигляді незалежних характеристик і дозволяє виконувати лише просте сортування. Однак жодна з базових систем не надає можливості користувачу налаштовувати ваги критеріїв відповідно до власних пріоритетів. Вибір електромобіля є індивідуальним процесом, що залежить від численних зовнішніх обставин — стилю водіння, характеру маршрутів, кліматичних умов, бюджету, вимог до зарядної інфраструктури та навіть особистих уподобань користувача щодо бренду. Відсутність можливості враховувати пріоритетність критеріїв призводить до того, що рекомендації системи стають усередненими та малоефективними, оскільки вони не відображають реальних потреб конкретної особи.

Проблема прозорості та пояснюваності рішень також є поширеною серед існуючих платформ. Навіть ті системи, які пропонують рейтинг моделей або формують певний підсумковий індикатор, як правило, не надають детального пояснення щодо того, як було сформовано результат. Користувач не отримує інформації про внесок кожного показника, не може відстежити логіку розрахунків і не має можливості оцінити коректність рекомендації. Така непрозорість суттєво

знижує довіру до результатів і суперечить сучасним тенденціям розвитку систем штучного інтелекту, у яких пояснюваність стає важливою складовою оцінювання якості алгоритмічних рішень.

Вагомим обмежувальним чинником є якість вихідних даних, якими оперують платформи. Частина ресурсів використовує відкриті джерела, що не завжди містять актуальну інформацію про нові моделі електромобілів або оновлення модельних лінійок. Немало сайтів оновлюють свої бази з великим запізненням, через що користувач отримує застарілі технічні характеристики. Інколи дані подаються у неоднорідному форматі, що ускладнює подальшу аналітичну обробку та порівняння. Нерідко відсутні важливі параметри, такі як потужність швидкісного заряджання, ефективність споживання енергії, тип хімії батареї, реальна місткість акумулятора або температурний вплив на запас ходу. Внаслідок цього можливості системи прийняття рішення стають обмеженими через низьку якість вхідної інформації.

Окремої уваги заслуговують комерційні конфігуратори автомобільних брендів. Хоча вони пропонують високу деталізацію інформації щодо певної моделі та дозволяють моделювати різні комплектації, їхня головна мета полягає у просуванні продукції конкретного виробника. Це призводить до обмеженої порівняльної здатності таких платформ та їхньої повної неспроможності здійснювати незалежний аналіз ринку. Вони надають глибоку, але однобічну інформацію, яка не може бути використана для вибору між різними брендами.

Натомість у зарубіжних інформаційно-аналітичних системах спостерігається тенденція до інтеграції методів багатокритеріального аналізу. Деякі DSS-рішення, орієнтовані на комерційні автосалони або корпоративних клієнтів, уже реалізують моделі нормування показників, застосовують методи MCDM[11,12,13] та пропонують користувачеві складові рейтинги з поясненням логіки обчислень. У таких системах комбінуються методи фільтрації, кластеризації, моделювання споживчих сценаріїв і прогнозування загальної

вартості володіння. На практиці це дозволяє отримувати більш точні та персоналізовані рекомендації. Однак більшість із цих платформ має закриту комерційну модель і не доступна для широкого використання, а їх інтеграція у локальні ринки є обмеженою.

У контексті українського ринку ситуація є ще складнішою. Національні інформаційні ресурси, які містять цілісні бази електромобілів з можливістю аналітичного порівняння, практично відсутні. Існуючі локальні платформи здебільшого дублюють функціональні можливості простих каталогів, що не виходять за рамки базової фільтрації. Вони не використовують математичні моделі багатокритеріального оцінювання, не забезпечують прозорості рішень та не пропонують персоналізованих рекомендацій. Розрив між потребами користувачів та функціональністю доступних платформ стає дедалі відчутнішим у зв'язку зі стрімким зростанням ринку електромобілів в Україні.

Узагальнюючи викладене, можна стверджувати, що нинішній стан розвитку інформаційних систем у сфері підбору електромобілів не повною мірою відповідає вимогам комплексного аналізу. Більшість наявних платформ надає лише локальні функціональні можливості, не інтегруючи їх у єдину багатокритеріальну модель оцінювання. Відсутність персоналізації, низька якість вихідних даних, обмежена глибина аналітики та брак пояснюваності рішень значно знижують ефективність таких систем. Усе це формує об'єктивну потребу у створенні DSS-системи нового покоління, яка поєднуватиме структурований набір актуальних даних, математичні та алгоритмічні методи аналізу, прозорість прийняття рішення та можливість адаптації під індивідуальні вимоги користувача.

### **1.3. Функціональне моделювання та аналіз існуючих процесів підбору електромобілів**

Процес підбору електромобіля є багатокроковою, комплексною та інтелектуально насиченою діяльністю, яка охоплює низку взаємопов'язаних дій користувача, що виконуються у контексті неоднорідної та часто неповної інформації. Незважаючи на наявність великої кількості цифрових ресурсів, котрі пропонують технічні дані про електромобілі, реальна практика вибору залишається фрагментарною та проливає світло на відсутність цілісного підходу до прийняття рішення. Більшість рішень приймається користувачами в умовах інформаційної асиметрії та відсутності формалізованого механізму оцінювання, що створює передумови для суб'єктивних помилок, зниження точності вибору та збільшення часу, необхідного для прийняття остаточного рішення.

На початковому етапі користувач, як правило, здійснює пошук доступної інформації про моделі електромобілів. Цей етап характеризується високим ступенем розпорошеності даних, оскільки вони розміщені на різноманітних вебресурсах, включно з оглядовими порталами, тематичними блогами, офіційними сайтами автовиробників, комерційними каталогами, інтернет-магазинами, маркетплейсами та численними сторонніми джерелами. Через різноманітність форматів подання інформації, відсутність єдиних стандартів її структурування та різну ступінь актуальності користувач змушений витратити значну кількість часу на ручне порівняння параметрів, їх уточнення та перевірку достовірності. Це, у свою чергу, ускладнює формування об'єктивного первинного набору кандидатів для подальшого аналізу.

Другим етапом процесу є початкова фільтрація автомобілів, яка зазвичай здійснюється за кількома узагальненими характеристиками, такими як ціна, запас ходу, тип кузова, потужність або рік випуску. Однак можливості фільтрації в наявних інформаційних системах обмежуються лише поверхневими параметрами й не охоплюють більш складних техніко-економічних взаємозалежностей

[14,15,16]. Системи не враховують поведінкові особливості користувача, його пріоритети, стиль водіння, можливості доступу до зарядної інфраструктури та інші контекстуальні чинники, які істотно впливають на доцільність вибору тієї чи іншої моделі. Через це множина моделей, отримана після первинної фільтрації, залишається надмірно великою, а процес подальшого порівняння — трудомістким.

Суттєвою проблемою є також відсутність систематизованих механізмів багатокритеріального порівняння. Після первинного відбору користувач змушений самостійно зіставляти технічні характеристики, які нерідко суперечать одна одній. Наприклад, модель із більшим запасом ходу може мати вищу вагу та гіршу динаміку; автомобіль із низьким енергоспоживанням може мати суттєво меншу швидкість DC-заряджання, що впливатиме на зручність використання під час далеких поїздок. Через відсутність статистичних, аналітичних або алгоритмічних інструментів користувач фактично намагається інтуїтивно сформулювати інтегральне уявлення про автомобіль. Такий підхід позбавлений математичної обґрунтованості та супроводжується високою часткою суб'єктивності.

Ще одним обмеженням є неможливість персоналізації ваг критеріїв. Вибір електромобіля завжди є індивідуальним, і для кожного користувача різні характеристики мають різний ступінь важливості. Однак більшість інформаційних систем пропонує узагальнений підхід, за якого жорстко фіксуються критерії та їх вплив на вибір. У результаті рейтинг, сформований такою системою, не відображає реальних пріоритетів користувача. Так, для одного покупця головним критерієм є економічність та низьке енергоспоживання, для іншого — висока швидкість заряджання, для третього — динамічні характеристики або наявність повного приводу. За відсутності інструментів персоналізації рішення системи стає універсальним, проте малопродатним для практичного застосування.

Ключовою проблемою сучасних платформ залишається відсутність прозорості логіки формування рекомендацій. Рейтинги, що надаються деякими сервісами, не супроводжуються поясненнями щодо методів обчислення підсумкового показника та внеску окремих характеристик. Користувач фактично отримує результат, який постає як «чорна скринька» без можливості проаналізувати його структуру. Така непрозорість суперечить концепції пояснюваних систем підтримки прийняття рішень і не дозволяє оцінити якість рекомендацій, їх достовірність та відповідність реальним очікуванням. У практичному вимірі відсутність пояснюваності є вагомим бар'єром для прийняття важливих фінансових рішень, пов'язаних із купівлею дороговартісного транспортного засобу.

Для усвідомлення характеру проблем та ідентифікації можливих шляхів їх усунення було проведено функціональне моделювання існуючих процесів підбору електромобіля. Моделювання дозволило формалізувати ключові етапи взаємодії користувача з інформаційними ресурсами, виділити основні інформаційні потоки, визначити можливі точки оптимізації та встановити роль алгоритмічних компонентів у процесі прийняття рішення. Побудована модель відображає типовий сценарій оброблення даних: від пошуку інформації, її вилучення та очищення до порівняння моделей, нормалізації показників та формування підсумкового результату. На основі цього аналізу встановлено, що відсутність автоматизованих процедур нормалізації, багатокритеріального оцінювання та пояснення результатів є основними обмеженнями, які знижують ефективність сучасних інструментів.

Дослідження показало, що переважна більшість існуючих інформаційних систем функціонує здебільшого як каталоги, що агрегують технічні дані, але не виконують аналітичних трансформацій, здатних перетворювати ці дані у рішення. Каталогізація є важливою складовою, однак вона лише створює передумови для аналітики, а не забезпечує її фактичну реалізацію. Відсутність алгоритмічних

механізмів залишається ключовим фактором, що обмежує корисність таких систем для користувачів, які прагнуть отримати об'єктивну, науково обґрунтовану та персоналізовану рекомендацію.

У підсумку встановлено, що сучасні системи підтримки вибору електромобілів не забезпечують необхідної інтегральності оцінювання, не дозволяють гнучко формувати критерії вибору, не пояснюють логіку прийняття рішення та не базуються на методах багатокритеріального аналізу. Ці фундаментальні недоліки обґрунтовують потребу у створенні інтелектуальної DSS-системи нового покоління, здатної виконувати структурований збір даних, їх нормалізацію, фільтрацію, багатокритеріальне оцінювання та прозоре пояснення результатів. Впровадження такої системи забезпечить підвищення точності вибору електромобіля, зниження суб'єктивності оцінювання, оптимізацію часу прийняття рішення та формування більш обґрунтованих рекомендацій для кінцевих користувачів.

#### **1.4. Виявлені задачі та проблеми під час аналізу процесів підбору електромобілів**

Аналіз сучасних процесів підбору електромобілів, здійснений на основі огляду інформаційних систем, дослідження методів порівняння та побудови функціональних моделей, виявив значну кількість проблем та обмежень, які перешкоджають формуванню обґрунтованого рішення щодо вибору оптимального транспортного засобу. Установлено, що існуючі підходи не забезпечують відповідного рівня автоматизації, аналітичної глибини та адаптивності, що у свою чергу ускладнює процес прийняття рішення, знижує його об'єктивність та підвищує ризик суб'єктивних помилок. Виявлені проблеми мають системний характер і зумовлені як інформаційними, так і технологічними обмеженнями, притаманними сучасним платформам у сфері електромобільності.

Однією з ключових проблем, яка суттєво впливає на весь процес вибору, є фрагментованість даних. Інформація про технічні, економічні та експлуатаційні

характеристики електромобілів існує у вигляді численних окремих фрагментів, розміщених на різних вебресурсах, сторінках виробників, комерційних каталогах та оглядових порталах. Відсутність єдиного формату подання даних, узгоджених одиниць вимірювання, стандартизованих категорій та актуалізованих наборів параметрів змушує користувача виконувати тривалу та трудомістку роботу з ручного збору, перевірки, зіставлення та нормалізації інформації. На практиці це означає, що навіть базове порівняння кількох моделей потребує значної кількості часу та зусиль, тоді як оброблення більших наборів характеристик майже неможливе без спеціалізованих інструментів. Неоднорідність джерел зумовлює появу інформаційних розривів, знижує точність оцінювання та створює ризик прийняття рішення на основі застарілих або некоректних даних.

Ще однією проблемою є відсутність механізмів повноцінного багатокритеріального оцінювання. Хоча вибір електромобіля є типовою задачею багатокритеріального аналізу, сучасні платформи не реалізують навіть базових математичних методів, які б дозволяли врахувати взаємозалежність параметрів, їх вагомість та відносний вплив на кінцевий результат. Існуючі системи фокусуються переважно на фільтрації за окремими характеристиками або поверхневому сортуванні, що суттєво обмежує можливості об'єктивного порівняння. Реальний процес вибору потребує одночасного врахування десятків параметрів — запасу ходу, швидкості заряджання, енергоспоживання, вартості, динаміки, безпеки, технологічного оснащення, наявності зарядної інфраструктури та ін. Проте сучасні системи не здатні об'єднати ці показники у цілісну інтегральну оцінку, що створює потребу у застосуванні математичних моделей, які наразі відсутні.

Не менш суттєвим обмеженням є відсутність персоналізації під час вибору. Різні користувачі віддають перевагу різним характеристикам залежно від стилю водіння, характеристик маршрутів, можливостей заряджання, бюджету або особистих уподобань. Наприклад, для водія, що здійснює здебільшого міські поїздки, першочерговим може бути енергоспоживання та компактність автомобіля, тоді як для активного мандрівника — швидкість заряджання та запас

ходу. Утім, більшість інформаційних систем не дозволяє змінювати ваги критеріїв чи налаштовувати індивідуальний профіль пріоритетів. Внаслідок цього рекомендації таких систем набувають універсального характеру, що суттєво знижує їхню практичну цінність та робить процес підбору недостатньо точним.

Брак пояснюваності результатів є ще одним фундаментальним недоліком існуючих систем. У тих випадках, коли платформи все ж формують рейтинги або надають рекомендації, користувач зазвичай не отримує інформації про структуру обчислень, значущість кожного критерію чи причини, що обумовили конкретну позицію моделі у рейтингу. Такі системи функціонують за принципом «чорної скриньки», що суперечить концепції пояснюваних систем підтримки прийняття рішень (Explainable DSS). Відсутність прозорої логіки підвищує недовіру користувача до результатів, робить процес вибору менш надійним і унеможливорює верифікацію достовірності запропонованих рекомендацій. У контексті дороговартісних придбань, до яких належить купівля електромобіля, ця проблема набуває особливого значення.

Крім інформаційних недоліків, важливе місце займають і технологічні обмеження. Значна частина існуючих платформ побудована на застарілих підходах до архітектури систем, що не передбачають масштабованої роботи з великими масивами даних або з об'ємами параметрів, які притаманні сучасним електромобілям. Відсутність засобів автоматизованої нормалізації, попередньої обробки та валідації даних ускладнює побудову коректних алгоритмів порівняння. Багато платформ не підтримують інтеграцію з зовнішніми джерелами або API-взаємодію, що обмежує їхню здатність отримувати актуальну інформацію про моделі та комплектації. Разом з цим відсутні механізми інтелектуальної обробки даних, які дозволили б будувати прогностичні моделі, оцінювати загальну вартість володіння або моделювати поведінкові сценарії користувачів.

Унаслідок цього формуються конкретні задачі, що потребують вирішення. Першою з них є необхідність створення повністю структурованого, стандартизованого та актуального набору даних про електромобілі, який міститиме узгоджені характеристики та уніфіковані одиниці вимірювання.

Другою задачею є впровадження повного багатокритеріального оцінювання, що передбачає використання математичних моделей, здатних інтегрувати десятки параметрів у єдиний індикатор відповідності потребам користувача. Третьою задачею постає автоматизація процесів нормалізації, очищення та валідації даних, що забезпечить підвищення точності подальших розрахунків. Четвертою задачею є розроблення прозорого та пояснюваного механізму формування рішень, який дозволить користувачеві розуміти структуру оцінювання та логіку рекомендацій. Нарешті, важливою є задача підвищення рівня персоналізації та інтерактивності системи, що дозволить адаптувати результати під конкретні сценарії використання та індивідуальні потреби.

Сукупність зазначених викликів свідчить про актуальність створення інтелектуальної DSS-системи нового покоління, яка забезпечуватиме системний підхід, високу аналітичну здатність, адаптивність та можливість прийняття обґрунтованих рішень. Така система повинна поєднувати структуровані бази даних, сучасні алгоритми багатокритеріального аналізу, автоматизовані модулі обробки інформації та механізми пояснюваності. Її розроблення дозволить значно підвищити точність порівняння електромобілів, мінімізувати суб'єктивність процесу вибору та оптимізувати загальну тривалість прийняття рішення, що особливо важливо в умовах швидкого зростання ринку електротранспорту.

### **1.5. Аналітичний огляд літератури**

Аналітичний огляд наукових джерел показує, що підбір електромобілів, моделювання процесів прийняття рішень та створення рекомендаційних систем у сфері автомобільного ринку становлять міждисциплінарний напрям, який інтегрує досягнення інформатики, штучного інтелекту, теорії прийняття рішень, прикладної математики та економіки транспорту. Зростання інтересу до цієї тематики зумовлено різким розширенням ринку електромобілів, збільшенням кількості моделей, ускладненням технічних характеристик і необхідністю

адаптації рішень до індивідуальних вимог споживачів, що відображено у значній кількості публікацій останніх років.

У контексті багатокритеріальної оптимізації, яка є фундаментальною основою для аналізу й порівняння електромобілів, вагоме місце посідають класичні та сучасні методи оцінювання альтернатив. Математичні підходи, такі як WSM, АНР, TOPSIS, ELECTRE[17] та інші алгоритми MCDM, традиційно застосовуються для задач, у яких об'єкти мають різнорідні характеристики, різні шкали вимірювання та неоднакову вагомість у прийнятті рішення. У наукових працях наголошується, що методи MCDM є незамінними тоді, коли користувачеві потрібно структурувати та формалізувати складні залежності між критеріями, які можуть вступати у суперечність: наприклад, збільшення запасу ходу зазвичай призводить до підвищення маси та вартості електромобіля, тоді як зменшення енергоспоживання може негативно вплинути на динамічні характеристики.

Значна частина досліджень присвячена аналізу ефективності методів багатокритеріальної оптимізації у транспортних задачах. У працях, присвячених АНР, відзначається його здатність структурувати задачу прийняття рішень у вигляді ієрархічної моделі та враховувати експертні оцінки ваг критеріїв. TOPSIS широко застосовується для задач, у яких важливою є відповідність альтернатив ідеальному рішенню та віддаленість від антиідеального. ELECTRE та PROMETHEE[18] характеризуються здатністю працювати з нечіткими та слабо структурованими критеріями. Дослідники підкреслюють, що поєднання цих методів з алгоритмами машинного навчання відкриває можливості для автоматизації аналізу великих наборів даних, хоча така інтеграція потребує якісної структури даних та алгоритмічної підтримки.

У науковій літературі також активно досліджується питання побудови рекомендаційних систем, що є одним із ключових елементів інтелектуальних DSS-рішень. Рекомендаційні системи традиційно використовуються в електронній комерції, медіаплатформах та соціальних мережах, проте за останні

роки спостерігається активне їх впровадження у транспортний сектор. Дослідники відзначають, що рекомендаційні моделі в автомобільному ринку мають враховувати велику кількість показників, які не можуть бути проаналізовані за допомогою класичних евристичних фільтрів. Особливу увагу приділено персоналізації рекомендацій, оскільки відсутність адаптації до потреб конкретного користувача значно знижує ефективність системи. У роботах з машинного навчання і згорткових моделей прийняття рішень наголошується, що алгоритми можуть виявляти приховані кореляції між характеристиками автомобілів та поведінкою користувачів, але вони вимагають значних обсягів даних і прозорості, яка часто недосяжна для моделей «чорної скриньки».

Важливим напрямом, який знайшов відображення у сучасній літературі, є вивчення ролі пояснюваності у системах підтримки прийняття рішень (Explainable Decision Support Systems). Автори численних публікацій наголошують, що системи нового покоління повинні не лише формувати остаточний рейтинг або рекомендацію, а й забезпечувати користувача детальним поясненням підстав для цього рішення. Пояснюваність (explainability) підвищує рівень довіри до системи, дозволяє перевірити коректність її роботи, усунути потенційні помилки у даних і дає змогу користувачу зрозуміти, які характеристики електромобіля мають найбільший вплив на оцінку. У цьому контексті науковці підкреслюють необхідність поєднання математичних моделей, які забезпечують прозорість розрахунків, із сучасними алгоритмами аналізу даних, котрі підтримують високий рівень адаптивності.

Також існує значний пласт літератури, присвячений технічним та інфраструктурним аспектам розвитку електромобільності. У численних дослідженнях розглядаються особливості акумуляторних технологій, питання енергетичної ефективності, вплив кліматичних і дорожніх умов на запас ходу та якість роботи батареї. Велика увага приділяється аналітиці розвитку зарядної інфраструктури, її впливу на прийняття рішень користувачами та регіональному

нерівномірному розподілу станцій швидкісної та повільної зарядки. У роботах з транспорту та урбаністики підкреслюється, що параметри електромобілів оновлюються надзвичайно швидко, тому інформаційні системи, які їх аналізують, повинні забезпечувати регулярне оновлення даних та підтримувати автоматизовані механізми валідації. Дослідники зазначають, що існуючі каталоги електромобілів часто страждають від різноманітності форматів подання інформації, що унеможлиблює їх безпосереднє використання у аналітичних DSS-системах.

Узагальнення наукових джерел дозволяє констатувати, що хоча окремі компоненти, необхідні для побудови сучасної системи підтримки прийняття рішення щодо вибору електромобіля, добре описані у літературі, цілісні комплексні рішення усе ще перебувають на етапі формування. Науковці розробили широкий спектр моделей багатокритеріального аналізу, алгоритмів статистичної обробки даних, рекомендаційних систем і підходів до пояснюваності, однак їх застосування до задачі підбору електромобілів відзначається недостатньою повнотою. Бракує робіт, які б поєднували структуровані набори даних, математичну модель інтегрального оцінювання, гнучку систему налаштування вагових коефіцієнтів, алгоритмічну прозорість та адаптивну логіку рекомендацій. Саме ці прогалини визначають наукову новизну та актуальність подальших досліджень, а також формують підґрунтя для створення інтелектуальної DSS-системи нового покоління, здатної відповідати вимогам сучасного ринку електромобілів і забезпечувати користувача надійними, обґрунтованими та пояснюваними рекомендаціями.

### **1.6. Постановка завдання на дослідження.**

Проведений комплексний аналіз предметної області, узагальнення наукових підходів до моделювання процесів прийняття рішень та вивчення існуючих інформаційних систем підбору електромобілів засвідчили наявність суттєвого дисбалансу між потребами користувачів і функціональними можливостями сучасних інструментів. Визначено, що процес вибору

електромобіля характеризується багатовимірністю та високою складністю, оскільки включає аналіз великої кількості параметрів, що відрізняються природою, шкалами вимірювання, динамікою змін та взаємозалежностями. Така багатофакторність процесу формує потребу у застосуванні методологічно обґрунтованих математичних моделей і спеціалізованих алгоритмів, здатних забезпечити структуровану обробку даних, об'єктивне оцінювання альтернатив і прозоре обґрунтування результатів.

Водночас виявлено, що існуючі платформи лише частково вирішують поставлені задачі, оскільки виконують здебільшого довідкову або консультативну функцію й не містять необхідних аналітичних компонентів. Вони не забезпечують стандартизації даних, не підтримують повноцінного багатокритеріального аналізу, не надають можливостей адаптації вагових коефіцієнтів та не містять модулів пояснення рішень. У зв'язку з цим користувач змушений самостійно збирати інформацію з різних джерел, усувати суперечності в даних, перетворювати параметри до спільних форматів, визначати логіку оцінювання та виконувати інтуїтивне порівняння альтернатив, що значно підвищує когнітивне навантаження та створює ризик формування помилкових висновків.

Актуальність постановки завдання зумовлена не лише технічними обмеженнями існуючих систем, а й стрімким розвитком ринку електротранспорту. Щороку з'являються нові моделі з покращеними характеристиками, модернізованими силовими установками, оновленими акумуляторними системами та розширеними можливостями швидкісного заряджання. Попри це, інформаційні системи не встигають адаптуватися до високої динаміки оновлення даних, що призводить до застарівання інформаційної бази. Відповідно, постає необхідність формувати математично обґрунтовані процедури оцінювання, здатні коректно обробляти як структуровані, так і частково неструктуровані дані про електромобілі, а також адаптуватися до нових параметрів, які з'являються внаслідок технічної еволюції ринку.

Ураховуючи виявлені проблеми, наукове завдання дослідження полягає у розробленні інтелектуальної системи підтримки прийняття рішень (DSS), здатної

забезпечити комплексний, адаптивний та пояснюваний підхід до підбору електромобілів. Основною метою дослідження є створення програмно-аналітичної моделі, яка поєднуватиме методи багатокритеріальної оптимізації, алгоритми оброблення даних, сучасні засоби персоналізації та модулі інтерпретації результатів, забезпечуючи повний цикл автоматизованої підтримки процесу вибору.

Постановка завдання передбачає формування математичного опису проблеми, який включає визначення множини альтернатив, структури критеріїв, їхніх вагових коефіцієнтів та правил інтегрального оцінювання. Досягнення цієї мети потребує розроблення моделі, здатної враховувати різну природу характеристик електромобілів, зокрема технічні параметри (динамічні властивості, потужність, ємність акумулятора, запас ходу), економічні показники (ціна, вартість володіння, витрати на заряджання та обслуговування) та експлуатаційні критерії (комфорт, безпека, технологічне оснащення). Важливо забезпечити узгодженість масштабів вимірювання та можливість інтерпретаційної обробки, що вимагає впровадження процедур нормалізації та стандартизації даних.

Окрему увагу необхідно приділити вибору та адаптації методів багатокритеріальної оптимізації. У межах дослідження ставиться завдання реалізувати алгоритми WSM та TOPSIS, оскільки вони широко застосовуються у науковій літературі, відзначаються високою прозорістю математичного апарату та придатні для практичного використання у DSS-системах. Метод WSM дозволяє отримувати інтегральну оцінку альтернативи через лінійну комбінацію нормалізованих значень критеріїв, що забезпечує простоту інтерпретації та універсальність застосування. Метод TOPSIS забезпечує аналіз відносної близькості альтернативи до ідеального та антиідеального рішень, що підвищує точність ранжування та дозволяє враховувати складні взаємозалежності між критеріями.

Завдання також включає формування алгоритмів попередньої обробки інформації, які забезпечать очищення даних від аномалій, виявлення пропусків,

узгодження форматів і перетворення параметрів до єдиної шкали. Це є необхідною передумовою для коректної роботи алгоритмів MCDM та уникнення викривлень у результатах. Важливим компонентом постановки завдання є реалізація механізму пояснення рішень, що дозволить користувачеві отримати прозору інформацію про внесок кожного критерію у підсумковий результат, а також зрозуміти логіку обчислень. Це підвищить довіру до системи та відповідатиме сучасним вимогам до пояснюваних моделей прийняття рішень.

Крім того, створення інтелектуальної DSS-системи потребує розроблення інтерфейсу взаємодії, здатного забезпечити інтуїтивно зрозумілий доступ до даних, процедур налаштування вагових коефіцієнтів, запуску моделей аналізу та візуалізації отриманих результатів. Інтерфейс повинен забезпечувати як простоту використання для початкових користувачів, так і достатню гнучкість для експертів, які потребують можливості глибокого аналізу та масштабування простору рішень.

У межах дослідження також ставиться завдання проведення експериментальної перевірки запропонованої моделі на основі реальних сучасних даних про електромобілі 2025 року. Це включає тестування стабільності роботи алгоритмів, оцінювання точності ранжування, аналіз чутливості моделі до зміни вагових коефіцієнтів та виявлення можливих обмежень у поведінці системи. Експериментальна частина повинна підтвердити, що створена DSS-система здатна адекватно інтерпретувати реальні дані, формувати узгоджені рекомендації та демонструвати переваги над існуючими платформами.

Таким чином, постановка завдання формує науково-практичну основу для створення інтерактивної, адаптивної та прозорої DSS-системи нового покоління, здатної виконувати повний цикл оброблення інформації про електромобілі, підтримувати користувача в процесі прийняття рішень та забезпечувати високий рівень обґрунтованості, точності та персоналізації рекомендацій. Значення цього завдання виходить за межі вузької технічної проблеми й охоплює ширший контекст розвитку екологічно орієнтованих технологій та цифрових інструментів,

що формують інтелектуальну інфраструктуру майбутнього ринку електротранспорту.

### **1.7. Висновки до розділу 1**

У першому розділі було здійснено всебічне дослідження предметної області підбору електромобілів, проаналізовано сучасний стан ринку електротранспорту та визначено особливості, що впливають на процес прийняття рішень. Виявлено, що структура ринку характеризується високою динамічністю, швидким оновленням технічних характеристик та значною різноманітністю моделей, що ускладнює їх порівняння на основі традиційних підходів.

Аналіз існуючих інформаційних систем засвідчив, що доступні платформи не забезпечують комплексної підтримки вибору, оскільки обмежуються каталогізацією даних та елементарними засобами фільтрації. Виявлені недоліки, серед яких відсутність багатокритеріального оцінювання, персоналізованих налаштувань, механізмів пояснюваності рішень та стандартизованих структур даних, істотно знижують ефективність наявних систем.

Функціональне моделювання процесів вибору електромобіля підтвердило наявність інформаційних і технологічних бар'єрів, що перешкоджають формуванню обґрунтованого рішення. Моделі продемонстрували фрагментованість доступних даних, недостатність автоматизованих процедур оброблення та відсутність аналітичних модулів, необхідних для коректного порівняння альтернатив.

Аналітичний огляд літератури показав, що, попри наявність значної кількості наукових досліджень у галузях багатокритеріального аналізу та рекомендаційних систем, застосування цих підходів до задачі підбору електромобілів залишається обмеженим. Водночас сучасні методи, зокрема моделі MCDM та алгоритми інтелектуального аналізу даних, дозволяють створити науково обґрунтовану основу для розроблення ефективної DSS-системи.

На основі проведеного аналізу було сформульовано постановку завдання на дослідження, яка передбачає необхідність створення інтелектуальної системи

підтримки прийняття рішень нового покоління. Така система має забезпечувати структуровану роботу з даними, підтримку багатокритеріального оцінювання, адаптацію вагових коефіцієнтів під потреби користувача та прозоре пояснення результатів.

Таким чином, розділ 1 заклав теоретичне підґрунтя для подальших етапів роботи, визначивши наукову проблему, актуальність дослідження та напрям розвитку DSS-системи підбору електромобілів, що буде розроблятися у наступних розділах.

## РОЗДІЛ 2. ДОСЛІДЖЕННЯ ТА ОБҐРУНТУВАННЯ ТЕХНОЛОГІЙ, МЕТОДІВ, АЛГОРИТМІВ

### 2.1. Моделювання основних задач

Моделювання основних задач, що лежать в основі побудови інтелектуальної DSS-системи підбору електромобілів, є ключовим етапом дослідження, оскільки забезпечує формування чіткої, структурованої та математично обґрунтованої основи для подальшої реалізації аналітичних алгоритмів. У межах даного підрозділу здійснюється формалізація задачі вибору електромобіля, визначення характеристик альтернатив, опис критеріїв оцінювання, їх вагових коефіцієнтів, етапів нормалізації даних та математичних засобів інтегрального ранжування. Така формалізація є необхідною умовою розроблення програмно-аналітичної системи, що здатна здійснювати багатокритеріальне оцінювання у прозорий, логічно обґрунтований та адаптивний спосіб.

Проблема вибору електромобіля характеризується високою складністю, оскільки кожна модель має власний набір технічних, економічних та експлуатаційних параметрів, що оновлюються щороку та значно варіюють між собою. Моделі відрізняються за запасом ходу, потужністю, динамікою розгону, швидкістю заряджання, ємністю тягової батареї, показниками ефективності, вартістю володіння, рівнем технологічного оснащення та іншими характеристиками, що визначають їхню придатність до різних сценаріїв використання. Тому задачі, пов'язані з формуванням об'єктивної рекомендації, неможливо розв'язати за допомогою простих фільтрів або сортування даних. Необхідним є застосування математичних підходів, здатних структурувати, стандартизувати та інтегрувати інформацію.

Моделювання задачі починається з визначення множини альтернатив. У ролі альтернатив розглядаються реальні моделі електромобілів, доступні на ринку відповідного року. Кожна альтернатива описується вектором характеристик, який

може мати десятки компонентів. Однак особливість полягає в тому, що ці характеристики вимірюються у різних одиницях, мають різну шкалу та різний характер впливу на кінцеву оцінку. Наприклад, чим більший запас ходу — тим кращий показник, тоді як для енергоспоживання нижче значення є кращим. Крім того, такі параметри, як динаміка або швидкість заряджання, впливають на оцінку користувача по-різному залежно від його індивідуальних потреб. Таким чином, виникає потреба у нормалізації, що дозволяє привести всі значення до уніфікованої форми.

Нормалізація даних є центральним етапом моделювання, оскільки вона забезпечує математичну узгодженість інформації. У процесі нормалізації кожне значення критерію перетворюється так, щоб воно відображало відносну перевагу або недолік певної моделі. Це дозволяє усунути розмірні відмінності між параметрами, зробити їх сумісними та придатними для інтегрального аналізу. Коректно виконана нормалізація суттєво впливає на точність розрахунків, оскільки помилки на цьому етапі можуть призвести до викривлення підсумкової оцінки. Моделювання передбачає застосування таких нормалізаційних стратегій, які забезпечать інваріантність системи до одиниць вимірювання та масштабів значень.

Наступним етапом є формування структури критеріїв, які визначають якісні та кількісні параметри вибору. Критерії можуть мати різну природу: одні характеризують технічні показники, інші — економічну ефективність, треті — комфорт або функціональність. Утім, математична модель має забезпечувати можливість їх об'єднання у єдину систему оцінювання. Для кожного критерію встановлюється ваговий коефіцієнт, що впливає на ступінь його значущості для користувача. Ваги повинні бути нормовані так, щоб їх сума дорівнювала одиниці, що відповідає класичній постановці задачі багатокритеріального аналізу. Моделювання задачі включає опис правил визначення ваг, їх інтерпретацію та можливість адаптації, що робить систему персоналізованою.

У межах математичної постановки задачі вибору електромобіля приймається припущення про існування функції корисності, яка інтегрує нормалізовані значення критеріїв з урахуванням їх вагових коефіцієнтів. Подальше формування рейтингу моделей розглядається як процес багатокритеріального оптимізаційного оцінювання. Моделювання передбачає використання методів WSM та TOPSIS як основних інструментів оброблення критеріїв. Метод WSM, який ґрунтується на зваженій сумі нормалізованих показників, дозволяє отримати інтегральну оцінку для кожної альтернативи. Він є простим, прозорим і зручним для інтерпретації, що робить його особливо придатним для першого етапу аналізу.

Метод TOPSIS забезпечує глибший рівень оцінювання, оскільки базується на концепції ідеальної та антиідеальної альтернативи. Ідеальна альтернатива відповідає найкращим значенням усіх критеріїв, тоді як антиідеальна — найгіршим. TOPSIS визначає ступінь близькості кожної реальної альтернативи до ідеальної, що дає змогу враховувати відстань між моделями у багатовимірному просторі характеристик. Такий підхід робить результати більш точними, особливо в умовах суперечливих критеріїв.

Особливе місце у моделюванні займає аналіз структури даних, які використовуються у DSS-системі. Реальні інформаційні набори електромобілів містять пропуски значень, дублікати, неоднорідні формати та нестандартні позначення. Це зумовлює необхідність моделювання процедури очищення та валідації даних, що дозволить виявляти аномалії, видаляти некоректні записи та забезпечувати коректну роботу алгоритмів. Модель повинна враховувати специфіку джерел даних, відмінності у точності параметрів та варіативність форматів позначення числових значень.

У результаті моделювання формуються логічні та математичні залежності, які описують повний життєвий цикл роботи DSS-системи: від збору та нормалізації даних до отримання підсумкового ранжування альтернатив.

Створена модель окреслює взаємозв'язки між параметрами електромобіля, алгоритмами оцінювання, ваговими коефіцієнтами та кінцевими рекомендаціями. Чітка формалізація задачі забезпечує можливість алгоритмізації кожного етапу, що є необхідною передумовою розроблення програмного забезпечення, здатного виконувати складні операції з великими наборами даних. Таким чином, моделювання основних задач завершує теоретичний етап формування DSS-системи та створює основу для її подальшої реалізації у програмному середовищі.

## **2.2. Огляд та аналіз існуючих рішень для розв'язання виявлених задач**

Аналіз сучасних рішень, які застосовуються у сфері підтримки прийняття рішень та підбору технічних об'єктів, свідчить про багаторівневий характер розвитку відповідних методологій і програмних засобів. У наукових та інженерних дослідженнях особлива увага приділяється розробленню методів, що дозволяють інтегрувати інформацію різної природи, здійснювати порівняння характеристик альтернатив у багатовимірному просторі параметрів та забезпечувати прозорість прийнятих рішень. Завдання вибору електромобілів вписується у загальний контекст задач багатокритеріального оптимізаційного оцінювання, де кожна альтернатива описується вектором числових та якісних показників, що відрізняються масштабом, характером впливу та ступенем значущості.

У дослідженнях з прийняття рішень зазначається, що традиційні підходи, засновані на інтуїтивній оцінці або простому порівнянні характеристик, не дозволяють забезпечити достатній рівень об'єктивності та структурованості процесу. Це зумовлює необхідність застосування математичних моделей, здатних виконувати комплексну обробку критеріїв. До таких моделей належать методи багатокритеріального аналізу (MCDM). Вони включають цілу низку підходів, серед яких широко відомими є WSM, AHP, TOPSIS, ELECTRE та PROMETHEE. Кожен із методів має власні особливості, але всі вони дозволяють зіставляти альтернативи на основі інтегральної оцінки.

Метод WSM є одним із найпростіших та найбільш інтуїтивних підходів. Він передбачає нормалізацію критеріїв, визначення вагових коефіцієнтів та обчислення зваженої суми. У багатьох практичних розробках саме WSM є базовим методом, оскільки його реалізація потребує мінімальних обчислювальних ресурсів, а результати легко інтерпретувати. Це робить його придатним для інтерактивних DSS-рішень, де швидкість реакції системи має важливе значення. Проте метод має й обмеження: він не враховує взаємозалежності між критеріями та використовує лінійну модель оцінювання, яка не завжди достатньо гнучка для задач реального світу.

Метод TOPSIS є розвитком ідеї багатовимірного порівняння альтернатив. Він базується на концепції відстані до ідеального та антиідеального рішень. Завдяки цьому TOPSIS не просто агрегує значення критеріїв, а й визначає геометричну близькість кожної альтернативи до теоретично найкращого варіанту. Такий підхід робить метод надзвичайно ефективним у випадках, коли критерії суперечать один одному або коли рішення потребує збалансованості між кількома параметрами. TOPSIS широко використовується у транспортній логістиці, енергетиці, інженерії та екологічних дослідженнях. У контексті вибору електромобілів метод TOPSIS демонструє високу точність та стабільність, що підтверджується значною кількістю публікацій, у яких він застосовується для аналізу автомобільної техніки, енергоефективності та екологічних характеристик.

Методи АНР, ELECTRE та PROMETHEE також знаходять застосування у виборі технічних систем, але мають низку обмежень. Метод АНР передбачає попарне порівняння критеріїв, що ускладнює його використання у системах з великою кількістю параметрів. Крім того, він базується на суб'єктивних експертних оцінках, що може знижувати об'єктивність моделі. Методи ELECTRE та PROMETHEE мають складнішу математичну структуру, включають процедури побудови відношень переваги та паретоподібних ранжувань, що підвищує точність аналізу, але робить ці методи менш інтерпретованими та складнішими для реалізації у програмному середовищі, орієнтованому на широке коло користувачів.

Паралельно з методами MCDM у сучасних інформаційних системах активно розвиваються рекомендаційні алгоритми. Їх застосування охоплює електронну комерцію, медіаплатформи, транспортне планування та інженерні системи. Рекомендаційні моделі засновані на статистичному аналізі, машинному навчанні або гібридних підходах. У випадку електромобілів застосування машинного навчання часто обмежене через відсутність великих масивів історичних даних. На відміну від ринків з тривалими історіями продажів та мільйонами записів, сегмент електромобілів є порівняно новим і недостатньо насиченим даними для роботи складних моделей, таких як нейронні мережі чи колаборативна фільтрація.

Сучасні онлайнві платформи, які позиціонуються як інструменти для вибору автомобілів, здебільшого виконують функції каталогів. Вони надають базові можливості фільтрації та порівняння параметрів, але не підтримують алгоритмічний аналіз альтернатив. Користувач може обрати діапазон вартості, запас ходу або тип кузова, однак система не проводить глибокого багатокритеріального аналізу. Немає механізмів інтегрального оцінювання, не підтримуються вагові коефіцієнти, не реалізовано моделі нормалізації даних. У більшості випадків рекомендації не супроводжуються поясненням, що саме вплинуло на місце моделі у рейтингу, а отже користувач позбавлений можливості перевірити логіку рішення та скоригувати параметри вибору.

Окремі системи більш високого класу реалізують часткову підтримку багатокритеріального аналізу, але стикаються з іншими обмеженнями. Часто вони не містять повного набору характеристик електромобілів, а працюють із застарілими чи неповними даними. Крім того, у багатьох випадках інформація подається у різних форматах, що потребує додаткової обробки перед аналізом. У спеціалізованих дослідженнях підкреслюється важливість попередньої обробки даних, яка включає нормалізацію, очищення, виявлення пропусків та корекцію аномалій. Без таких процедур результати багатокритеріального оцінювання можуть бути суттєво викривлені.

У наукових публікаціях наголошується на значущості інтеграції даних з різних джерел. Розподіленість характеристик електромобілів між каталогами,

технічними документами, оглядовими сайтами та комерційними ресурсами ускладнює побудову єдиної моделі. Для коректної роботи аналітичної системи необхідно розробити механізми об'єднання даних, їх структурування та забезпечення однорідності. Це особливо складно у випадку параметрів, які залежать від умов експлуатації, таких як реальний запас ходу або середні витрати енергії.

Системи, що застосовують машинне навчання, показують високу ефективність у персоналізованих рекомендаціях, проте для роботи таких алгоритмів необхідні детальні індивідуальні профілі користувачів. У сфері електромобілів такі профілі зазвичай відсутні або містять лише загальну інформацію, що значно знижує ефективність моделей. Тому класичні методи MCDM залишаються найбільш збалансованим рішенням, оскільки не потребують великих тренувальних вибірок, характеризуються стабільністю, простотою реалізації та високою інтерпретованістю результатів.

Огляд літератури та аналіз існуючих рішень дозволяють зробити висновок, що оптимальним підходом для створення інтелектуальної DSS-системи підбору електромобілів є поєднання класичних багатокритеріальних методів з сучасними алгоритмами оброблення даних. Такий гібридний підхід забезпечує баланс між точністю, прозорістю, швидкістю та можливістю адаптації під потреби користувача. Методи WSM та TOPSIS, доповнені механізмами якісної нормалізації, очищення та валідації даних, формують основу для побудови ефективної системи, здатної розв'язувати задачі, що не покриваються існуючими платформами.

### **2.3. Обґрунтування вибору технологій, методів, алгоритмів**

Обґрунтування вибору технологій, методів та алгоритмів для створення інтелектуальної DSS-системи підбору електромобілів має ключове значення для забезпечення її ефективності, стабільності, точності та здатності до масштабування. Оскільки система призначена для роботи з великими обсягами різномірних технічних та економічних характеристик, важливою є адаптація

обраних інструментів до специфіки таких даних і до потреб кінцевого користувача. Поставлена задача належить до класу багатокритеріальних задач прийняття рішень, де необхідно інтегрувати параметри, що мають різні шкали вимірювання, різний ступінь важливості та, інколи, різний характер впливу на кінцеву оцінку. Саме тому вибір методології оцінювання альтернатив та технологій її реалізації повинен ґрунтуватися на комплексному аналізі доступних підходів.

Багатокритеріальні методи MCDM (Multi-Criteria Decision Making) були обрані як основа алгоритмічної частини системи, оскільки вони дозволяють формалізувати процес прийняття рішень у ситуаціях, де необхідно одночасно враховувати значну кількість критеріїв. Наукова література підтверджує, що методи WSM, АНР, ELECTRE, PROMETHEE та TOPSIS охоплюють широкий спектр моделей оцінювання, здатних працювати в умовах різної складності критеріїв. Проте для конкретної задачі підбору електромобілів найбільш доцільними виявилися методи WSM і TOPSIS, оскільки вони поєднують прозорість, математичну строгість і відносно низькі вимоги до обчислювальних ресурсів. Їх вибір зумовлений необхідністю створення системи, орієнтованої на кінцевого користувача, який повинен мати можливість як зрозуміти логіку оцінювання, так і отримати стабільний результат у реальному часі.

Метод WSM (Weighted Sum Model) є одним із найдавніших і найпростіших методів багатокритеріального оцінювання альтернатив. Його теоретична база ґрунтується на припущенні, що загальна корисність альтернативи може бути представлена у вигляді лінійної комбінації її нормалізованих критеріїв. У контексті підбору електромобілів WSM виявився надзвичайно корисним завдяки своїй прозорості та гнучкості. Користувач може інтуїтивно зрозуміти, як зміна вагового коефіцієнта впливає на результат. Це важливо, оскільки для різних покупців профіль пріоритетів суттєво відрізняється: один може надавати перевагу запасу ходу, інший — доступності зарядної інфраструктури, третій — вартості володіння. WSM також невибагливий до структур даних і добре працює навіть тоді, коли кількість критеріїв значна, а їх значення попередньо нормалізовано.

Метод TOPSIS був обраний як другий головний алгоритмічний механізм системи з огляду на його здатність моделювати компромісні рішення. Основна ідея методу полягає у побудові двох гіпотетичних точок — ідеальної та антиідеальної альтернативи. Ідеальна альтернатива є моделлю, яка має найкращі значення критеріїв, можливі для ринку електромобілів, тоді як антиідеальна містить найгірші значення. Подальший аналіз базується на оцінюванні відстаней між реальною моделлю та цими умовними точками. Це дає змогу визначити, наскільки альтернативи наближені до оптимальних рішень. Для задачі порівняння електромобілів TOPSIS є особливо цінним, оскільки автомобільні характеристики часто суперечать одна одній: більш потужний двигун, як правило, збільшує енергоспоживання, а високий запас ходу може супроводжуватися збільшенням маси і вартості акумулятора. TOPSIS дозволяє ранжувати моделі у такий спосіб, який враховує баланс між цими суперечливими властивостями.

Застосування зазначених методів неможливе без використання механізмів нормалізації, які забезпечують узгодженість різнорідних критеріїв. У реальних даних про електромобілі параметри можуть відрізнятися на порядки, наприклад, ємність батареї вимірюється в кіловат-годинах, час заряджання — у хвилинах, потужність — у кіловатах, а ціна — у доларах або євро. Нормалізація дозволяє привести всі значення до безрозмірної шкали та усунути дисбаланс, який може виникнути внаслідок домінування критеріїв із великими числовими діапазонами. Вибір методу нормалізації також має критичне значення. Для даної системи застосовуються лінійні нормалізаційні функції, що забезпечують стабільність, простоту реалізації та коректне порівняння критеріїв, орієнтованих як на максимізацію, так і на мінімізацію.

Окремим важливим аспектом є вибір інструментарію для реалізації DSS-системи. Мова програмування Python[19,20] була обрана як основний технологічний інструмент завдяки своїй широкій популярності, доступності, простоті синтаксису, підтримці великих масивів наукових бібліотек та ефективності під час виконання аналітичних проєктів. Python є фактичним стандартом у галузях машинного навчання, аналізу даних, статистичного

моделювання та побудови DSS-систем, що підтверджується великою кількістю наукових робіт та індустріальних проєктів.

Бібліотеки NumPy[21,22] та Pandas[23,24] забезпечують ефективну роботу з масивами, векторами та табличними структурами, дозволяючи виконувати операції над даними з високою швидкістю. Pandas, зокрема, дає змогу завантажувати таблиці електромобілів, виконувати фільтрацію, нормалізацію, очищення даних, перетворення форматів та обробку пропусків. Це особливо важливо, оскільки реальні технічні специфікації часто містять відсутні значення, різні формати позначення параметрів або неконсистентні структури.

Алгоритмічна частина системи реалізована із застосуванням компонентів Scikit-learn[25,26]. Це одна з найпопулярніших бібліотек для машинного навчання, яка також включає інструменти для нормалізації, обробки ознак, обрахунку відстаней, стандартизації та інших операцій, необхідних для коректної роботи MCDM-методів. Використання Scikit-learn дозволяє забезпечити високу точність обчислень, оскільки бібліотека пройшла багатоетапну перевірку та верифікацію в наукових середовищах.

Для побудови інтерфейсу було обрано Streamlit[27,28] — сучасний фреймворк для створення інтерактивних веб-інтерфейсів без потреби у фронтенд-розробці. Streamlit дозволяє користувачу взаємодіяти з моделлю — змінювати вагові коефіцієнти, обирати критерії, переглядати результати у вигляді таблиць і графіків. Однією з важливих переваг є можливість швидкого оновлення системи та розширення її функціональності без значних витрат часу. Для DSS-систем, які повинні залишатися гнучкими й адаптивними, такий інструмент є надзвичайно цінним.

Таким чином, поєднання методів WSM і TOPSIS, процедур нормалізації та обробки даних, технологій Python, Pandas, NumPy, Scikit-learn і Streamlit забезпечує комплексне, адаптивне та інтерпретоване рішення для задачі підбору електромобілів. Такий набір інструментів забезпечує збалансованість між математичною строгістю, прозорістю, продуктивністю та можливістю масштабування. Обрана методологія дозволяє створити DSS-систему, яка не лише

надає користувачеві точні рекомендації, а й демонструє зрозумілу логіку їх формування, забезпечуючи тим самим довіру до результатів.

#### **2.4. Висновки до розділу 2**

У другому розділі було проведено систематичне моделювання задач, пов'язаних із побудовою інтелектуальної DSS-системи підбору електромобілів, а також здійснено огляд сучасних методів та технологій, придатних для реалізації такої системи. На основі аналізу предметної області сформовано математично обґрунтовану постановку задачі, у межах якої вибір електромобіля розглядається як багатокритеріальна оптимізація, що потребує інтеграції різнорідних технічних, економічних та експлуатаційних показників.

Побудована модель відображає повний цикл оброблення інформації — від збору й нормалізації даних до застосування методів багатокритеріального аналізу та формування ранжованого списку альтернатив. Обґрунтовано, що нормалізація є ключовим елементом системи, оскільки забезпечує узгодженість параметрів, які мають різні одиниці вимірювання та неоднорідні діапазони значень. Визначено роль вагових коефіцієнтів, які надають змогу адаптувати модель до індивідуальних потреб користувача та формувати персоналізовані рекомендації.

У результаті аналізу існуючих методів багатокритеріального оцінювання встановлено, що найдоцільнішим для задачі порівняння електромобілів є поєднання методів WSM та TOPSIS. Метод WSM забезпечує прозорість і простоту інтерпретації, тоді як метод TOPSIS дозволяє проводити глибший аналітичний аналіз, спираючись на концепцію близькості до ідеального рішення. Така комбінація дає змогу досягти балансу між точністю, обчислювальною ефективністю та пояснюваністю результатів.

Обґрунтовано вибір технологічного інструментарію, який включає використання Python, бібліотек Pandas, NumPy та Scikit-learn для роботи з даними та реалізації алгоритмів, а також Streamlit для створення інтуїтивного та інтерактивного інтерфейсу користувача. Сукупність цих технологій забезпечує

масштабованість, гнучкість і високу продуктивність системи, що є важливими для подальшої програмної реалізації DSS-рішення.

Таким чином, розділ 2 сформував методологічну та технологічну основу для розроблення інтелектуальної системи підтримки прийняття рішень щодо вибору електромобіля. Результати розділу створюють надійну базу для реалізації програмної частини системи, яка буде представлена в наступному розділі роботи.

## РОЗДІЛ 3. РОЗРОБЛЕННЯ ТА АПРОБАЦІЯ АНАЛІТИЧНОЇ МОДЕЛІ

### 3.1. Архітектура та загальна структура DSS-системи

Архітектура інтелектуальної DSS-системи підбору електромобілів розроблена відповідно до принципів модульності, масштабованості та незалежності компонентів, що забезпечує гнучкість, надійність та можливість подальшого розвитку системи без зміни її фундаментальної структури. Модельна побудова DSS-рішення дозволяє виділити набір відносно ізольованих функціональних блоків, кожен з яких відповідає за певний етап оброблення інформації — від завантаження даних до формування рекомендацій і пояснення прийнятих рішень. Такий підхід є загальновизнаним у сучасному проектуванні інформаційних систем, зокрема тих, що працюють із великими наборами даних і складними алгоритмами багатокритеріальної оптимізації.

Загальна структура DSS-системи ґрунтується на поділі на три основні рівні: рівень даних, рівень алгоритмів та рівень взаємодії з користувачем. На кожному з рівнів реалізовано окремі модулі, що забезпечують певний набір логічних функцій, але разом утворюють цілісну програмно-аналітичну платформу. Такий розподіл функцій дає змогу мінімізувати взаємозалежність компонентів, підвищити стабільність системи та уникнути дублювання обчислень або помилок у процесі інтеграції даних.

#### Архітектура рівня даних

На найнижчому рівні структури розташовано модуль loader, який відповідає за імпорт, валідацію та первинний аналіз даних. Він виконує завантаження CSV-файлу[29], перевіряє цілісність структури, типи значень, наявність обов'язкових полів та коректність форматів. Важливим аспектом є виявлення пропусків, аномальних значень та потенційних невідповідностей між полями, оскільки саме якість вхідних даних визначає точність подальших обчислень. Модуль loader реалізує механізми базового очищення, включаючи перетворення числових полів, вирівнювання форматів, усунення некоректних записів. Завдяки цьому він не лише завантажує дані, а й готує їх до глибшої обробки на наступному етапі.

Наступним компонентом є модуль *preprocessing*, який виконує розширену підготовку даних до аналітичної обробки. Його основні функції включають нормалізацію числових значень, стандартизацію показників, корекцію пропусків, перетворення категоріальних ознак, логічну перевірку параметрів і створення повністю узгодженої структури, сумісної з алгоритмами багатокритеріального аналізу. Нормалізація є ключовим елементом *preprocessing*-блоку, оскільки характеристики електромобілів суттєво відрізняються масштабами (ціна вимірюється в доларах, запас ходу — у кілометрах, час заряджання — у хвилинах), що унеможлиблює їхнє порівняння без попереднього приведення до єдиної шкали. Модуль *preprocessing* таким чином виконує роль «моста» між реальними даними та математичними моделями.

### **Алгоритмічне ядро системи**

На середньому рівні архітектури розташовано алгоритмічні модулі *scorer* та *recommender*, що формують центральну частину DSS-системи. Модуль *scorer* реалізує обчислення рейтингових оцінок альтернатив на основі методів WSM і TOPSIS. Він використовує нормалізовані дані та вагові коефіцієнти, які користувач задає відповідно до власних пріоритетів. Метод WSM дозволяє швидко обчислити інтегральну оцінку кожної моделі шляхом зваженої суми критеріїв, що забезпечує високу швидкодію. Метод TOPSIS, на відміну від WSM, використовує геометричну концепцію відстаней між альтернативами та ідеальними/антиідеальними точками, тому забезпечує точніший аналіз компромісів між характеристиками.

Алгоритмічна архітектура *intentionally* побудована так, щоб обидва методи могли працювати незалежно, але при цьому використовували спільну базу попередньо підготовлених даних. Такий підхід забезпечує можливість швидкої заміни алгоритмів у майбутньому або доповнення їх новими методами MCDM без зміни всієї системи. Крім того, *scorer* має внутрішні механізми перевірки коректності даних перед обчисленням, що суттєво підвищує надійність роботи.

Модуль *recommender* формує підсумковий ранжований список електромобілів, упорядкований за інтегральними оцінками. Він виконує агрегацію

результатів, сортування та генерацію фінальної рекомендації, яка може включати як повний список альтернатив, так і топ-3 чи топ-5 моделей, залежно від налаштувань користувача. Додаткові функції дозволяють адаптувати рекомендації до різних сценаріїв: вибір для міста, для дальніх подорожей, для бізнес-використання або для економного володіння.

Окремим компонентом алгоритмічного рівня є модуль explain, який забезпечує пояснюваність прийнятих рішень. Він формує текстові та графічні інтерпретації результатів, демонструє внесок кожного критерію в інтегральну оцінку, порівнює альтернативи між собою та генерує діаграми. Завдяки explain-модулю система відповідає сучасним вимогам explainable AI та забезпечує прозорість логіки оцінювання, що значно підвищує довіру користувача.

### **Рівень взаємодії з користувачем**

Верхній рівень DSS-системи становить користувацький інтерфейс, реалізований за допомогою фреймворку Streamlit. Цей фреймворк забезпечує інтерактивність, простоту використання та можливість відображення аналітичних результатів у реальному часі. Інтерфейс включає модулі для завантаження даних, налаштування вагових коефіцієнтів, вибору методів оцінювання, перегляду результатів, формування рекомендацій та аналізу пояснень.

Система побудована так, що будь-яка зміна користувачем параметрів — ваг, фільтрів, вибору алгоритму — автоматично запускає процедуру повторних обчислень та оновлення візуалізації. Усі результати представлені у зрозумілій формі, що включає таблиці, інтерактивні графіки, гістограми, діаграми близькості TOPSIS, водоспадні діаграми впливу критеріїв та пояснювальні блоки. Такий підхід дає змогу користувачу аналізувати альтернативи, моделювати різні сценарії вибору та отримувати обґрунтовані рекомендації без необхідності глибокої технічної підготовки.

### **Загальна структурна інтеграція**

Загальна архітектура DSS-системи орієнтована на чітке розмежування рівнів відповідальності: рівень даних відповідає за їх якість, рівень алгоритмів — за обчислення, рівень інтерфейсу — за взаємодію з користувачем. Такий підхід

дозволяє ефективно масштабувати систему як горизонтально (додавання нових моделей, алгоритмів, джерел даних), так і вертикально (оптимізація обчислень, поглиблений аналіз, інтеграція з зовнішніми сервісами). Модульність забезпечує можливість швидкого оновлення окремих компонентів, підтримує розвиток системи та робить її адаптивною до змін ринку електромобілів.

Архітектура, побудована за принципами гнучкої модульності, дозволяє використовувати DSS-систему у широкому спектрі застосувань — від академічних досліджень та тестових демонстрацій до впровадження у реальних автосалонах чи онлайн-сервісах порівняння техніки. Її структура спроектована таким чином, щоб забезпечити високу продуктивність у задачах багатокритеріального аналізу, стабільність під час роботи з великими наборами даних та зручність для кінцевих користувачів.

### 3.2. Розроблення модулів завантаження, обробки та нормалізації даних

Розроблення модулів завантаження, обробки та нормалізації даних є одним із ключових етапів створення інтелектуальної DSS-системи підбору електромобілів, оскільки саме цей компонент забезпечує якість вхідної інформації, формує підґрунтя для роботи алгоритмів багатокритеріального оцінювання та гарантує коректність результатів. Дані про електромобілі за своєю природою є складними, різномірними й неконсистентними, тому спеціалізована підсистема оброблення покликана виконати низку операцій, які трансформують вихідний CSV-файл у структурований і нормалізований масив, придатний для математичного аналізу.

#### **Архітектура підсистеми роботи з даними**

Підсистема обробки даних DSS-системи складається з кількох логічно пов'язаних модулів, які реалізують послідовність етапів: завантаження даних, первинну валідацію, очищення, стандартизацію, перетворення категоріальних ознак, обробку пропусків і нормалізацію. Такий підхід забезпечує високий рівень адаптивності та дозволяє системі працювати з різними наборами технічних характеристик, що можуть відрізнятися за формою, структурою або повнотою.

Основним завданням є гарантування того, що усі параметри електромобілів — як числові, так і категоріальні — приведені до уніфікованого формату, який може бути використаний алгоритмічним ядром системи. Структура вхідних даних, що описана в додатку А, охоплює широкий спектр характеристик: від технічних (ємність батареї, запас ходу, потужність двигуна) до економічних (вартість, витрати на заряджання, сервісне обслуговування) та експлуатаційних (тип приводу, місткість салону, швидкість заряджання). Кожна з цих характеристик обробляється відповідним механізмом підсистеми preprocessing.

### **Модуль loader: завантаження та первинна валідація**

Модуль loader виконує завантаження даних із CSV-файлу та є першим компонентом, який взаємодіє з вхідною інформацією. Важливість цього етапу полягає в тому, що будь-які неточності, помилки чи структурні проблеми можуть істотно вплинути на роботу всієї DSS-системи. Модуль здійснює аналіз структури колонок, перевіряє наявність обов'язкових полів, визначає відповідність форматів та виконує базову типізацію даних. Якщо виявлено некоректні значення — наприклад, текстові записи там, де очікуються числові дані, або пропуски в критично важливих критеріях — вони позначаються як проблемні й передаються у preprocessing на етап корекції.

Особливим завданням цього модуля є обробка абераційних значень, які можуть бути наслідком помилок у джерелах даних. Наприклад, у технічних каталогах нерідко зустрічаються нульові значення для параметрів, де нуль фізично неможливий (наприклад, нульовий час заряджання). Loader визначає такі кейси й передає інформацію для подальшого очищення.

### **Модуль preprocessing: очищення, стандартизація та перетворення даних**

Найбільш вагомим компонентом підсистеми роботи з даними є модуль preprocessing. Він виконує широкий спектр операцій, що спрямовані на приведення вхідних даних до структурованого та математично узгодженого вигляду. Різноманітність параметрів електромобілів створює певні складності,

адже деякі характеристики потребують математичної обробки (наприклад, числові показники), а інші — логічного перетворення (категоріальні ознаки).

Першим кроком preprocessing-блоку є очищення даних від пропусків. Пропуски заповнюються відповідно до типу критерію: середнім значенням для більшості технічних параметрів, медіаною — для економічних характеристик, а для категоріальних параметрів застосовується найбільш поширене значення або спеціальна позначка «невідомо». Це забезпечує цілісність набору та можливість подальшої обробки.

Другим етапом є узгодження форматів. Часто одна й та сама характеристика у різних джерелах може мати різні позначення, одиниці вимірювання або структуру запису. Наприклад, потужність може бути вказана у kW або hp, час заряджання — у хвилинах або годинах. Preprocessing виконує їх конвертацію до єдиної системи.

Категоріальні характеристики, такі як тип приводу чи тип акумулятора, трансформуються у векторні представлення (one-hot encoding або ordinal encoding), що дозволяє алгоритмам працювати з ними нарівні з числовими параметрами.

Особлива увага приділяється параметрам типу «чим менше — тим краще». До таких належать вартість автомобіля, енергоспоживання на 100 км, час заряджання. Перед нормалізацією вони помічаються окремим маркером, що дозволяє системі коректно інтерпретувати їх під час розрахунку рейтингової оцінки. Це особливо важливо, оскільки некоректне тлумачення таких параметрів може призвести до викривлення результатів аналізу.

### **Механізм нормалізації: математика та обґрунтування**

Оскільки алгоритми WSM і TOPSIS працюють лише з нормалізованими значеннями, підсистема нормалізації є обов'язковою складовою системи. Для цієї DSS-системи було обрано мінімакс-нормалізацію як найбільш збалансований підхід. Вона перетворює значення кожного критерію у діапазон  $[0;1]$ , що робить усі параметри порівнянними між собою. Математично це виражається формулою:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (3.1)$$

де:

$x$  — вихідне значення критерію;

$x'$  — нормалізоване значення;

$x_{max}$  — максимальне значення критерію;

$x_{min}$  — мінімальне значення критерію.

Для витратних критеріїв застосовується інверсна формула:

$$x' = \frac{x_{max} - x}{x_{max} - x_{min}} \quad (3.2)$$

де:

$x$  — вихідне значення критерію;

$x'$  — нормалізоване значення;

$x_{max}$  — максимальне значення критерію;

$x_{min}$  — мінімальне значення критерію.

Завдяки цьому підхід гарантує, що більші значення завжди свідчать про кращі властивості електромобіля незалежно від характеру параметра.

У процес нормалізації також інтегровано обробку аномалій. Якщо показник виходить за межі реалістичного діапазону, він або усувається, або замінюється значенням, отриманим шляхом інтерполяції. Це забезпечує додаткову стійкість системи.

### **Структуризація програмного рішення та роль модульності**

У межах розроблення програмної архітектури підсистема обробки даних була реалізована у вигляді окремих модулів, перелік яких наведено в додатку Б. Такий підхід забезпечує максимальну гнучкість системи: кожен модуль може оновлюватися незалежно, а структура DSS дозволяє легко інтегрувати нові джерела даних або додаткові алгоритми аналізу.

Модульна структура забезпечує можливість швидкої адаптації під зміни у специфікаціях електромобілів — наприклад, появу нових показників, таких як напруга швидкісної зарядки, коефіцієнт деградації батареї або енергоефективність у різних кліматичних умовах.

### **Підсумкова підготовка даних**

Останнім етапом є формування фінального масиву даних, що передається до алгоритмічного ядра системи для розрахунку інтегральних оцінок WSM і TOPSIS. Після нормалізації дані набувають необхідної однорідності, логічної структурованості та математичної готовності до аналізу. Завдяки цьому вся DSS-система функціонує стабільно, а значення електромобілів коректно порівнюються у рамках багатокритеріальної моделі.

Таким чином, розроблена підсистема завантаження, очищення, стандартизації та нормалізації даних є фундаментом роботи DSS-системи. Вона забезпечує узгодженість технічних характеристик електромобілів, робить можливим об'єктивне багатокритеріальне оцінювання та формує основу для отримання точних, надійних і пояснюваних рекомендацій.

### **3.3. Реалізація алгоритмів багатокритеріального оцінювання альтернатив**

Алгоритмічне ядро розробленої DSS-системи відіграє ключову роль у формуванні обґрунтованих, прозорих та математично коректних рекомендацій під час вибору електромобілів. Реалізація багатокритеріальних алгоритмів є критичною, оскільки саме на цьому етапі відбувається інтеграція підготовлених даних, застосування вагових коефіцієнтів, урахування особливостей нормалізації та визначення оптимальності кожного варіанту з погляду множини характеристик. У запропонованій системі реалізовано два базові методи багатокритеріального аналізу — WSM та TOPSIS, які забезпечують два різних концептуальних підходи: лінійне зважування критеріїв та геометричну інтерпретацію близькості альтернатив до «ідеального» рішення. Обидва методи реалізовані у модулі *scorer* і функціонують на основі єдиної нормалізованої матриці даних, що надходить із preprocessing-модуля. Такий підхід забезпечує узгодженість, універсальність та можливість подальшого масштабування алгоритмічного ядра.

### **Формальна модель методу WSM**

Метод WSM (Weighted Sum Model) є одним із найпоширеніших у сфері багатокритеріального оцінювання завдяки своїй лаконічності, простоті реалізації та здатності забезпечувати прозорі результати, які легко пояснити користувачу. Суть методу полягає у розрахунку інтегральної оцінки кожної альтернативи шляхом додавання зважених значень усіх критеріїв. Перед обчисленнями усі критерії підлягають нормалізації, завдяки чому значення різних параметрів (наприклад, запас ходу в кілометрах і час заряджання в хвилинах) стають порівнюваними.

Кожна альтернатива описується нормалізованим вектором:

$$X_i = (x_{i1}, x_{i2}, \dots, x_{in}) \quad (3.3)$$

де

$x_{ij}$  — нормалізоване значення критерію  $j$  для альтернативи  $i$ ;

$n$  — кількість критеріїв.

Ваги критеріїв задаються користувачем у формі вектора:

$$W = (w_1, w_2, \dots, w_n), \quad \sum_{j=1}^n w_j = 1 \quad (3.4)$$

де:

$w_j$  — ваговий коефіцієнт критерію  $j$ ;

$n$  — кількість критеріїв;

$\sum w_j = 1$  — нормувальна умова, що забезпечує коректне зважування.

Ця умова забезпечує нормовану інтерпретацію важливості критеріїв та уможливорює коректне зваження. Якщо користувач задає довільні ваги, система автоматично виконує їх нормування.

Інтегральна оцінка альтернативи визначається за формулою:

$$S_i = \sum_{j=1}^n w_j x_{ij} \quad (3.5)$$

де:

$S_i$  — підсумкова оцінка альтернативи  $i$ ;

$w_j$  — вага критерію  $j$ ;

$x_{ij}$  — нормалізоване значення критерію  $j$  для альтернативи  $i$ ;

$n$  — кількість критеріїв.

Отримане значення  $S_i$  описує "якість" альтернативи за набором критеріїв та відповідними вагами. Чим більше значення, тим кращою є альтернатива з погляду заданих пріоритетів.

У програмній реалізації метод WSM опрацьовується за допомогою векторизованих операцій бібліотеки NumPy, що значно прискорює обчислення навіть для великих наборів даних. Завдяки цьому результат розраховується практично миттєво, забезпечуючи можливість повторних обчислень у реальному часі при зміні ваг або виборі підмножини критеріїв.

### Формальна модель методу TOPSIS

Метод TOPSIS ґрунтується на концепції вибору альтернативи, що є найбільш близькою до ідеального рішення та найбільш далекою від антиідеального. Його використання особливо доцільне в задачах, де існує конкуренція критеріїв або суперечливість між ними. Наприклад, збільшення потужності електромобіля часто супроводжується зростанням енергоспоживання, тобто покращення одного критерію погіршує інший. TOPSIS дозволяє врахувати такі компроміси більш математично коректно, ніж WSM.

Нехай матриця нормалізованих значень має вигляд:

$$X = (x_{ij}), \quad i = 1, \dots, m; j = 1, \dots, n \quad (3.6)$$

де:

$m$  — кількість альтернатив;

$n$  — кількість критеріїв;

$x_{ij}$  — нормалізоване значення критерію  $j$  альтернативи  $i$ .

Ідеальний та антиідеальний вектори визначаються так:

$$A^+ = \left( \max_i x_{i1}, \max_i x_{i2}, \dots, \max_i x_{in} \right) \quad (3.7)$$

де:

$A^+$  — вектор найкращих можливих значень критеріїв;

$\max_i x_{ij}$  — максимальне значення критерію  $j$  серед усіх альтернатив.

$$\left( A^- = \left( \min_i x_{i1}, \min_i x_{i2}, \dots, \min_i x_{in} \right) \right) \quad (3.8)$$

де:

$A^-$  — вектор найгірших можливих значень критеріїв;

$\min_i x_{ij}$  — мінімальне значення критерію  $j$ .

Ці два вектори формують "полюси" рішення: максимально бажаний і максимально небажаний стани за всіма критеріями.

Для кожної альтернативи обчислюються відстані до цих двох векторів:

$$\sqrt{\sum (D_i^+)^2} = \sqrt{\sum_{j=1}^n (x_{ij} - A_j^+)^2} \quad (3.9)$$

де:

$D_i^+$  — евклідова відстань альтернативи  $i$  до ідеального вектора;

$x_{ij}$  — нормалізоване значення критерію  $j$ ;

$A_j^+$  — ідеальне значення критерію  $j$ .

$$\sqrt{\sum (D_i^-)^2} = \sqrt{\sum_{j=1}^n (x_{ij} - A_j^-)^2} \quad (3.10)$$

де:

$D_i^-$  — відстань до антиідеального рішення;

$x_{ij}$  — нормалізоване значення критерію  $j$ ;

$A_j^-$  — найгірше значення критерію  $j$ .

Після цього визначається показник близькості до оптимального рішення:

$$C_i = \frac{D_i^-}{D_i^+ + D_i^-} \quad (3.11)$$

де:

$C_i$  — інтегральний індекс близькості альтернативи  $i$  до оптимального рішення;

$D_i^+$  — відстань до найкращого еталону;

$D_i^-$  — відстань до найгіршого еталону.

Значення  $C_i$  набуває значень у межах  $[0;1][0;1][0;1]$ , і чим ближче воно до 1, тим кращою вважається альтернатива. Реалізація TOPSIS у модулі *scorer* передбачає застосування векторизованих операцій та оптимізованих функцій NumPy, що забезпечує швидке формування матриць відстаней та підсумкової оцінки.

### **Алгоритмічна інтеграція та формування рейтингу**

Після розрахунку інтегральних оцінок за допомогою обох методів результати передаються до модуля *recommender*, який здійснює сортування альтернатив і формує рейтинг електромобілів. Система підтримує одночасний перегляд результатів двох методів, що дозволяє порівнювати їх між собою і визначати стійкість оцінки. Паралельно модуль *explain* генерує словесні та графічні пояснення, включно з діаграмами, які показують вклад кожного критерію, відстані до еталонів у TOPSIS, а також детальний опис логіки оцінювання.

### **Узагальнення**

Реалізація методів WSM та TOPSIS у межах DSS-системи забезпечує комплексний інструментарій для оцінювання електромобілів із різними технічними, економічними та експлуатаційними характеристиками. Метод WSM забезпечує простоту та інтуїтивність, тоді як метод TOPSIS гарантує глибший математичний аналіз та здатність відобразити компромісність рішень. Об'єднання цих методів у межах єдиної архітектури дозволяє створити потужне, масштабоване та пояснюване алгоритмічне ядро, яке адаптується до широкого спектра сценаріїв вибору електромобіля. Перелік програмного коду можна побачити у додатку Б.

### **3.4. Реалізація інтерфейсу та візуалізації результатів**

Реалізація користувацького інтерфейсу є одним із ключових етапів розроблення інтелектуальної DSS-системи підбору електромобілів, оскільки саме через інтерфейс користувач взаємодіє з даними, алгоритмами та рекомендаціями, отримує візуальні й текстові пояснення та формує власні сценарії оцінювання. У

запропонованій системі інтерфейс побудовано за допомогою фреймворку Streamlit, який на сьогодні є одним із найефективніших засобів створення інтерактивних науково-аналітичних застосунків. Його вибір зумовлений здатністю генерувати сучасні веб-інтерфейси без складної фронтенд-розробки, забезпечувати високу швидкодію та підтримувати гнучкі механізми інтерактивного оновлення даних. Завдяки цьому інтерфейс системи є не лише візуально привабливим, а й функціонально адаптивним, що відповідає потребам як фахівців, так і пересічних користувачів.

Загальна структура інтерфейсу вибудована таким чином, щоб користувач міг виконати повний цикл взаємодії із системою — від завантаження вхідних даних до аналізу сформованих рекомендацій, їх порівняння між собою та інтерпретації отриманих результатів. Стартовий блок інтерфейсу забезпечує можливість завантаження CSV-файлу зі специфікаціями електромобілів. Щойно файл завантажено, система автоматично відображає його структуру у табличній формі, що дає змогу користувачеві переконатися у коректності форматування, наявності всіх необхідних колонок і правильності типів даних. Це важливо, адже багато користувачів можуть працювати з власними наборами характеристик, і система має забезпечувати гнучку адаптацію до їх змісту.

Після завантаження даних користувач переходить до налаштування вагових коефіцієнтів критеріїв, які відіграють ключову роль у процесі багатокритеріального оцінювання. Ваги можна задавати вручну через інтерфейсні слайдери або поля введення, що дає змогу точно налаштувати важливість кожного параметра, зокрема таких як запас ходу, потужність, час заряджання, ціна чи енергоефективність. Реалізація механізму ваг передбачає автоматичну нормалізацію значень, тому користувач не повинен слідкувати за тим, щоб їх сума дорівнювала одиниці: система робить це самостійно. Це забезпечує високу зручність та мінімізує ризик помилки.

Ключовим елементом інтерфейсу є вибір методу оцінювання — WSM чи TOPSIS. Обидва методи реалізовано у вигляді інтерактивних перемикачів, і зміна методу призводить до миттєвого перерахунку результатів. Важливо підкреслити,

що у Streamlit застосовано концепцію реактивного оновлення, завдяки якій кожна зміна параметра — ваги, методу, фільтра чи вхідного набору — автоматично ініціює повторне виконання відповідних частин програмного коду. Завдяки цьому користувач може в реальному часі досліджувати вплив змінних на підсумковий рейтинг електромобілів. У практичній роботі це означає, що система здатна не лише формувати рекомендації, а й виконувати дослідницьку функцію: користувач може аналізувати чутливість моделі до зміни ваг, порівнювати альтернативи за окремими критеріями та визначати оптимальну конфігурацію параметрів для власних потреб.

Центральне місце в інтерфейсі займає модуль візуалізації. Результати оцінювання подаються у вигляді таблиці, де моделі електромобілів впорядковано за спаданням інтегральної оцінки, що дає змогу швидко визначити найкращі альтернативи. Таблиця містить не лише узагальнені оцінки, а й нормалізовані значення критеріїв, початкові параметри та порівняльні індикатори, що дозволяє користувачу аналізувати структуру результату. Важливо, що таблиця підтримує інтерактивні функції: сортування, пошук, фільтрацію за окремими характеристиками. В умовах реального застосування це забезпечує зручність у роботі навіть за наявності великих наборів даних.

Окрему увагу приділено графічним інструментам. Система створює діаграми та графіки, які дають змогу швидко зрозуміти логіку роботи алгоритмів і структуру впливу критеріїв на результат. Для методу WSM формується графічна інтерпретація внеску кожного критерію у підсумкову оцінку, що дозволяє побачити, які саме параметри визначили позицію моделі в рейтингу. Для методу TOPSIS, у свою чергу, генеруються діаграми відстаней до ідеального та антиідеального рішень, що є ключовими складовими в методології цього алгоритму. Такі графіки дозволяють не лише порівнювати моделі між собою, а й оцінювати силу компромісів між критеріями.

Змістовним доповненням візуальних засобів є текстові пояснення, сформовані модулем explain. Цей модуль аналізує як абсолютні значення критеріїв, так і їх нормалізовані значення, після чого генерує опис логіки

прийняття рішення для кожної альтернативи. Пояснення включають інформацію про те, чому конкретний електромобіль отримав саме таку оцінку, які критерії мали найбільший вплив, як результати змінилися б за інших вагових умов та які альтернативи є найближчими конкурентами у рейтингу. Завдяки цьому система забезпечує високий рівень прозорості та реалізує принцип explainable AI — пояснюваного штучного інтелекту.

Інтерфейс DSS-системи розроблено з урахуванням практичної придатності у реальних умовах. У автосалонах або сервісах електромобільного консалтингу важливо, щоб менеджери могли швидко підготувати набір рекомендацій для клієнта та адаптувати їх під запит у реальному часі. Розроблена система підтримує такий сценарій завдяки своїй високій реактивності, доступності всіх параметрів для зміни та швидкому перерахунку результатів. Інтерфейс також є адаптивним для кінцевих споживачів, які бажають самостійно проаналізувати різницю між моделями. Він не містить складних елементів, а всі функції винесено у зрозумілі блоки з інтуїтивною структурою.

У межах візуальної складової системи реалізовано чітку логіку подання результатів. Спочатку користувач бачить загальний рейтинг. Далі система пропонує перейти до детального порівняння окремих моделей, а також до перегляду пояснень алгоритмів. Графічні блоки розділено на тематичні секції: аналіз критеріїв, порівняння альтернатив, TOPSIS-відстані, структура внеску критеріїв для WSM. Завдяки цьому інтерфейс дозволяє як виконати поверхневий аналіз, так і поглиблений, залежно від потреб користувача.

Узагальнюючи, реалізація інтерфейсу та засобів візуалізації забезпечує DSS-системі комплексність і завершеність. Завдяки використанню Streamlit вдалося створити динамічну, інтуїтивну та інтерактивну платформу, яка не потребує спеціальних технічних знань для використання, але водночас надає широкий спектр аналітичних можливостей. Система виконує функцію не лише інструмента оцінювання, а й візуального аналітичного середовища, придатного для дослідників, технічних фахівців, консультантів та споживачів. Саме завдяки

цьому її можна ефективно застосовувати у сферах продажу, технічного аудиту, консалтингу та вибору електромобілів за множиною критеріїв.

Скріншоти інтерфейсу інтелектуальної DSS-системи підбору електромобілів можна побачити у додатку В база даних яка використовувалась для перевірки роботи це Electric Vehicle Specifications Dataset 2025[30].

### **3.5. Висновки до розділу 3**

У третьому розділі було здійснено комплексне проектування, формалізацію та практичну реалізацію архітектури інтелектуальної DSS-системи підбору електромобілів, що ґрунтується на принципах модульності, структурованої організації даних та математично обґрунтованих методах багатокритеріального оцінювання. Розроблений програмно-алгоритмічний комплекс охоплює повний цикл роботи з інформацією — від завантаження та нормалізації характеристик до формування фінальних рекомендацій та їх візуального пояснення користувачу. Це дозволило створити систему, здатну забезпечувати об'єктивність, відтворюваність і прозорість процесу прийняття рішень.

На рівні архітектури сформовано багаторівневу структуру, у якій кожен модуль виконує чітко визначені функції. Модулі завантаження та попередньої обробки даних забезпечують підготовку вхідного набору характеристик електромобілів, усувають пропуски, нормалізують значення та роблять дані придатними для подальших математичних операцій. Така організація гарантує коректність подальшого багатокритеріального аналізу і дає змогу працювати з різномірними наборами характеристик без втрати точності.

У межах алгоритмічного ядра реалізовано два методи MCDM — WSM і TOPSIS, причому для кожного з них сформульовано відповідні математичні моделі, виконано нормування векторів критеріїв та описано механізми обчислення інтегральних оцінок. Реалізовані алгоритми дозволяють синтезувати лінійну (WSM) та геометричну (TOPSIS) логіку порівняння альтернатив, що робить систему гнучкою, чутливою до зміни параметрів і придатною для моделювання різних типів користувацьких сценаріїв. Таке поєднання забезпечує

можливість не лише напряму ранжувати альтернативи, а й аналізувати компроміси між суперечливими критеріями.

Окрему увагу приділено реалізації інтерфейсу системи, який побудовано на фреймворку Streamlit. Завдяки реактивній архітектурі інтерфейс забезпечує миттєве оновлення результатів у відповідь на зміну параметрів, вагових коефіцієнтів чи методу оцінювання. Це створює умови для інтерактивного експериментування та дає змогу користувачу аналізувати вплив навіть незначних змін у даних чи вагових налаштуваннях на фінальний рейтинг електромобілів. Візуальні засоби, включно з діаграмами внеску критеріїв, графіками TOPSIS-відстаней і таблицями порівняння альтернатив, формують цілісне уявлення про логіку роботи системи та забезпечують високий рівень пояснюваності.

Таким чином, у межах розділу було створено повноцінну програмно-аналітичну платформу, яка поєднує формальні математичні моделі, структуровану архітектуру та сучасні засоби користувацької взаємодії. Розроблена DSS-система демонструє здатність забезпечувати високоточний, прозорий і персоналізований вибір електромобілів, відповідаючи актуальним потребам сучасного ринку та вимогам до систем підтримки прийняття рішень. Отримані результати формують основу для подальшого удосконалення системи, інтеграції нових характеристик, розширення набору алгоритмів та створення адаптивних модулів прогнозування й машинного навчання.

## ВИСНОВКИ

У кваліфікаційній роботі проведено комплексне дослідження процесу підбору електромобілів та розроблено інтелектуальну DSS-систему, яка забезпечує багатокритеріальний аналіз альтернатив, персоналізацію вибору та прозоре обґрунтування рекомендацій. На основі виконаних теоретичних та практичних результатів сформульовано такі висновки.

По-перше, дослідження предметної області показало, що процес вибору електромобіля характеризується високою багатфакторністю, значною варіативністю технічних і економічних параметрів та необхідністю одночасного врахування десятків критеріїв, які нерідко суперечать один одному. Сучасні інформаційні платформи не забезпечують належного рівня інтеграції цих параметрів, не підтримують персоналізацію та не містять механізмів пояснюваності рішень. Це підтвердило актуальність створення DSS-системи нового покоління, здатної усунути обмеження існуючих рішень.

По-друге, аналітичний огляд літератури довів, що класичні методи багатокритеріального аналізу — зокрема WSM та TOPSIS — залишаються найбільш обґрунтованою математичною основою для задач, що вимагають оцінювання великої кількості різнорідних критеріїв. Вони забезпечують прозорість обчислень, інтерпретованість результатів та адаптацію до різних користувацьких профілів. При цьому метод TOPSIS демонструє високу точність у контексті складних компромісних характеристик, а WSM забезпечує простоту інтегрування у DSS-систему та зручність пояснення кінцевому користувачеві.

По-третє, у роботі сформульовано математичну модель задачі підбору електромобілів, що включає побудову структурованого простору альтернатив, визначення типів критеріїв, нормалізацію значень, задання вагових коефіцієнтів та формування інтегральної оцінки. Особливу увагу приділено коректному трактуванню витратних і вигрешних критеріїв, що забезпечило точність багатокритеріального порівняння.

По-четверте, розроблено архітектуру DSS-системи, побудовану на модульних принципах, що дозволило забезпечити незалежність, масштабованість і можливість подальшого розширення. Модулі завантаження, оброблення, нормалізації та оцінювання даних інтегруються в єдину аналітичну платформу, яка забезпечує повний цикл роботи — від підготовки даних до формування ранжованих рекомендацій.

По-п'яте, реалізовано алгоритмічне ядро системи, яке включає модулі обчислення WSM- та TOPSIS-оцінок, механізми визначення ідеальних та антиідеальних рішень, нормалізацію критеріїв, а також формування підсумкового рейтингу. Логіка роботи алгоритмів доповнена модулем пояснюваності, що формує текстові та графічні інтерпретації результатів, підвищуючи прозорість системи та довіру користувачів.

По-шосте, створено інтерактивний інтерфейс користувача на основі фреймворку Streamlit, який забезпечує швидку роботу з даними, можливість налаштування вагових коефіцієнтів, вибір методів оцінювання та візуалізацію результатів у вигляді таблиць, графіків і текстових пояснень. Інтерфейс орієнтований як на фахівців автосалонів, так і на кінцевих користувачів, завдяки чому система може бути впроваджена у широкому спектрі практичних сценаріїв.

По-сьоме, апробація системи на реальних наборах характеристик електромобілів підтвердила коректність моделей, стабільність роботи алгоритмів та здатність системи адекватно відображати різні сценарії користувацьких пріоритетів. Тестування продемонструвало, що зміна ваг або методу оцінювання суттєво впливає на структуру рекомендацій, що підтверджує гнучкість і практичність розробленої системи.

Таким чином, поставлену науково-практичну задачу виконано повністю: розроблено інтелектуальну DSS-систему підбору електромобілів, що поєднує сучасні методи багатокритеріального аналізу, ефективні інструменти обробки даних і зручний інтерфейс користувача. Система має значний практичний потенціал, може бути адаптована до роботи з широкими каталогами

електромобілів і здатна підтримувати прийняття рішень у реальних умовах автосалонів, дилерських центрів та онлайн-платформ.

Створена DSS-система є гнучкою, масштабованою та орієнтованою на подальший розвиток. Перспективними напрямками удосконалення є інтеграція методів машинного навчання для автоматичного визначення ваг, розширення бази даних, додавання прогнозних моделей вартості володіння та впровадження мобільної версії системи.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Електромобіль — стаття про конструкцію та розвиток електричних авто [Електронний ресурс]. — Режим доступу: <https://uk.wikipedia.org/wiki/Електромобіль> (дата звернення: 10.10.2025).
2. International Energy Agency — Global EV Outlook 2024 [Електронний ресурс]. — Режим доступу: <https://www.iea.org/reports/global-ev-outlook-2024> (дата звернення: 21.10.2025).
3. International Renewable Energy Agency — Global Energy Transition: Toward Decarbonized Transport [Електронний ресурс]. — Режим доступу: <https://www.irena.org/energytransition> (дата звернення: 23.10.2025).
4. BloombergNEF — Zero-Emission Vehicles Progress Report [Електронний ресурс]. — Режим доступу: <https://about.bnef.com/electric-vehicle-outlook> (дата звернення: 01.11.2025).
5. U.S. Department of Transportation — Electric Vehicle Charging Infrastructure Deployment [Електронний ресурс]. — Режим доступу: <https://www.transportation.gov/rural/ev> (дата звернення: 02.11.2025).
6. Multi-criteria decision analysis — основи багатокритеріального оцінювання в DSS [Електронний ресурс]. — Режим доступу: [https://en.wikipedia.org/wiki/Multi-criteria\\_decision\\_analysis](https://en.wikipedia.org/wiki/Multi-criteria_decision_analysis) (дата звернення: 22.12.2025).
7. Weighted sum model — опис моделі зваженої суми (WSM), її математична формула та використання для ранжування альтернатив [Електронний ресурс]. — Режим доступу: [https://en.wikipedia.org/wiki/Weighted\\_sum\\_model](https://en.wikipedia.org/wiki/Weighted_sum_model) (дата звернення: 13.12.2025).
8. TOPSIS — методологія, що ґрунтується на концепції близькості до ідеального рішення та віддаленості від антиідеального [Електронний ресурс]. — Режим доступу: <https://en.wikipedia.org/wiki/TOPSIS> (дата звернення: 15.12.2025).
9. Edmunds — Electric Vehicle Comparison Tools [Електронний ресурс]. — Режим доступу: <https://www.edmunds.com/ev> (дата звернення: 15.01.2026).

10. EV-Volumes — Global EV Sales Database [Електронний ресурс]. — Режим доступу: <https://www.ev-volumes.com> (дата звернення: 14.01.2026).
11. Overview of Multi-Criteria Decision-Making : огляд методів прийняття рішень за багатьма критеріями (MCDM), їх застосування, переваги та обмеження [Електронний ресурс]. — Режим доступу: <https://www.semanticscholar.org/paper/Overview-of-Multi-Criteria-Decision-Making/e935a26c109fd29000be80aeae0bb603d64b3053> (дата звернення: 05.12.2025).
12. Normalization Procedures on Multicriteria Decision Making : детальний опис нормалізації даних для приведення різнорідних критеріїв до безрозмірної форми [Електронний ресурс]. — Режим доступу: <https://www.scitepress.org/papers/2010/28961/28961.pdf> (дата звернення: 09.12.2025).
13. Past efforts in determining suitable normalization methods for multi-criteria decision-making : a short survey — дослідження про вплив методів нормалізації на результати MCDM [Електронний ресурс]. — Режим доступу: <https://www.frontiersin.org/journals/big-data/articles/10.3389/fdata.2022.990699/full> (дата звернення: 11.12.2025).
14. AutoRia — порівняння можливостей фільтрації авто [Електронний ресурс]. — Режим доступу: <https://auto.ria.com> (дата звернення: 06.04.2026).
15. Consumer Reports — EV Filtering Tools Review [Електронний ресурс]. — Режим доступу: <https://www.consumerreports.org/cars> (дата звернення: 07.04.2026).
16. Autolist — Electric Vehicle Search Limitations [Електронний ресурс]. — Режим доступу: <https://www.autolist.com> (дата звернення: 09.04.2026).
17. ELECTRE Method — загальний огляд методів ELECTRE (I, II, III, IV), їх математичні основи та застосування у багатокритеріальному ранжуванні [Електронний ресурс]. — Режим доступу: <https://en.wikipedia.org/wiki/ELECTRE> (дата звернення: 05.11.2025).

18. PROMETHEE — загальний огляд методу PROMETHEE, його принципів і застосувань [Електронний ресурс]. — Режим доступу: <https://en.wikipedia.org/wiki/PROMETHEE> (дата звернення: 10.11.2025).
19. Python Documentation : офіційна документація Python [Електронний ресурс]. — Режим доступу: <https://www.python.org/doc/> (дата звернення: 07.12.2025).
20. Python Language Reference : формальна специфікація мови Python [Електронний ресурс]. — Режим доступу: <https://docs.python.org/3/reference/index.html> (дата звернення: 07.12.2025).
21. NumPy Documentation : офіційна документація NumPy [Електронний ресурс]. — Режим доступу: <https://numpy.org/doc/stable/> (дата звернення: 07.12.2025).
22. NumPy Quickstart Tutorial : вступні матеріали NumPy [Електронний ресурс]. — Режим доступу: <https://numpy.org/doc/stable/user/quickstart.html> (дата звернення: 07.12.2025).
23. Pandas Documentation : офіційна документація Pandas [Електронний ресурс]. — Режим доступу: <https://pandas.pydata.org/> (дата звернення: 07.12.2025).
24. Pandas Getting Started Guide : базові матеріали Pandas [Електронний ресурс]. — Режим доступу: [https://pandas.pydata.org/docs/getting\\_started/index.html](https://pandas.pydata.org/docs/getting_started/index.html) (дата звернення: 07.12.2025).
25. Scikit-learn Documentation : офіційні матеріали бібліотеки Scikit-learn [Електронний ресурс]. — Режим доступу: <https://scikit-learn.org/stable/> (дата звернення: 07.12.2025).
26. Scikit-learn Data Preprocessing Module : методи нормалізації та масштабування даних [Електронний ресурс]. — Режим доступу: <https://scikit-learn.org/stable/modules/preprocessing.html> (дата звернення: 07.12.2025).
27. Streamlit Documentation : офіційна документація Streamlit [Електронний ресурс]. — Режим доступу: <https://docs.streamlit.io/> (дата звернення: 07.12.2025).
28. Streamlit Database Tutorials : інтеграція Streamlit з базами даних [Електронний ресурс]. — Режим доступу: <https://docs.streamlit.io/develop/tutorials/databases> (дата звернення: 07.12.2025).

29. CSV (Comma-Separated Values) — стаття про формат CSV [Електронний ресурс]. — Режим доступу: <https://ru.wikipedia.org/wiki/CSV> (дата звернення: 07.12.2025).
30. Electric Vehicle Specifications Dataset 2025 — набір даних електромобілів [Електронний ресурс]. — Режим доступу: <https://www.kaggle.com/datasets/urvishahir/electric-vehicle-specifications-dataset-2025> (дата звернення: 07.12.2025).

## ДОДАТКИ

### Додаток А. Структура вхідного набору даних

- brand – виробник електромобіля
- model – комерційна назва моделі
- price – ринкова вартість (\$)
- range\_km – запас ходу за циклом WLTP (км)
- battery\_capacity\_kwh – ємність тягової батареї (кВт·год)
- fast\_charging\_power\_kw\_dc – максимальна потужність швидкісної DC-зарядки (кВт)
- efficiency\_wh\_per\_km – енергоефективність (Вт·год/км)
- acceleration\_0\_100\_s – розгін 0–100 км/год (с)
- top\_speed\_kmh – максимальна швидкість (км/год)
- power\_hp – потужність силової установки (к.с.)
- torque\_nm – крутний момент (Н·м)
- drivetrain – тип приводу (FWD, RWD, AWD)
- seats – кількість місць
- car\_body – тип кузова (sedan, SUV, hatchback, coupe, minivan тощо)
- length\_mm – довжина автомобіля (мм)
- width\_mm – ширина автомобіля (мм)
- height\_mm – висота автомобіля (мм)
- weight\_kg – споряджена маса автомобіля (кг)
- wheelbase\_mm – колісна база (мм)

## Додаток Б. Перелік програмних модулів DSS-системи

### Б.1 — dss/init.py

```
# dss/__init__.py
__all__ = [
    "loader", "preprocessing", "aliases", "scorer",
    "recommender", "validation", "topsis", "explain"
]
```

### Б.2 — dss/aliases.py

```
import re

ALIASES = {
    "model": [
        "model", "name", "vehicle", "car", "trim", "variant", "edition"
    ],
    "brand": [
        "brand", "make", "manufacturer", "company"
    ],
    "price": [
        "price", "price_eur", "price_usd", "msrp", "cost", "market_price",
        "base_price", "starting_price", "sale_price"
    ],
    "range_km": [
        "range_km", "range", "wltp_range_km", "range_wltp_km",
        "electric_range_km", "epa_range_km", "cltc_range_km"
    ],
    "battery_kwh": [
```

```
"battery_kwh", "battery_capacity_kwh", "battery", "accumulator_kwh",  
"usable_battery_kwh", "total_battery_kwh", "net_battery_kwh"  
],
```

```
"eff_whkm": [  
  "efficiency_wh_km", "consumption_wh_km", "wh_km",  
  "energy_consumption_wh_km", "consumption",  
  "energy_consumption_kwh_100km",  
  "efficiency_wh_per_km"  
],
```

```
"fastcharge_kw": [  
  "fastcharge_kw", "dc_fast_kw", "peak_dc_kw", "max_dc_kw",  
  "dc_charging_power_kw", "fast_charge_power", "dc_fast_charge",  
  "fast_charging_power_kw_dc"  
],
```

```
"accel_0_100": [  
  "accel_0_100", "acceleration_0_100_s", "accel_s",  
  "zero_to_hundred_s", "acceleration", "0_100_kmh"  
],
```

```
"body": [  
  "body", "body_type", "car_body", "vehicle_class", "car_body_type"  
],
```

```
"segment": [  
  "segment", "vehicle_segment", "class", "market_segment",  
  "car_segment", "car_class", "category"  
],
```

```
"drive": [  
  "drive", "drivetrain", "awd_fwd_rwd", "drive_type",  
  "drivetrain_type", "powertrain"  
],  
  
"seats": [  
  "seats", "num_seats", "passengers", "seating_capacity"  
],  
  
"length_mm": [  
  "length_mm", "length", "vehicle_length_mm", "overall_length_mm"  
],  
"width_mm": [  
  "width_mm", "width", "overall_width_mm"  
],  
"height_mm": [  
  "height_mm", "height", "overall_height_mm"  
],  
  
"weight_kg": [  
  "weight_kg", "curb_weight_kg", "mass_kg", "vehicle_weight",  
  "kerb_weight", "gross_weight_kg"  
],  
  
"cargo_volume_l": [  
  "cargo_volume_l", "trunk_volume_l", "boot_volume_l"  
],  
"towing_capacity_kg": [  
  "towing_capacity_kg", "towing_kg", "tow_limit_kg"
```

```

],
"top_speed_kmh": [
    "top_speed_kmh", "vmax_kmh", "max_speed_kmh"
],
}

def normalize_column_name(c: str) -> str:
    return re.sub(r"^[a-z0-9]+", "_", c.strip().lower()).strip("_")

def find_column(df, aliases):
    candidates = []

    for alias in aliases:
        alias_norm = normalize_column_name(alias)

        if alias_norm in df.columns:
            return alias_norm

    for col in df.columns:
        if col == alias_norm:
            candidates.append((0, col))
        elif col.startswith(alias_norm):
            candidates.append((1, col))
        elif alias_norm in col:
            candidates.append((2, col))

    if candidates:
        candidates.sort(key=lambda x: x[0])
        return candidates[0][1]

```

```
return None
```

### Б.3 — dss/explain.py

```
import pandas as pd
```

```
import streamlit as st
```

```
import plotly.graph_objects as go
```

```
CRITERIA_LABELS = {
```

```
    "price": "Ціна",
```

```
    "range_km": "Запас ходу",
```

```
    "fastcharge_kw": "Швидка DC-зарядка",
```

```
    "eff_whkm": "Енергоефективність",
```

```
    "accel_0_100": "Розгін 0–100 км/год",
```

```
    "body_fit": "Відповідність типу кузова",
```

```
}
```

```
CRITERIA_EXPLANATION = {
```

```
    "price": "нижча вартість підвищує загальну привабливість моделі",
```

```
    "range_km": "великий запас ходу підвищує практичність авто у щоденному та міжміському використанні",
```

```
    "fastcharge_kw": "вища потужність DC-зарядки зменшує час поповнення енергії",
```

```
    "eff_whkm": "менше споживання енергії означає економічнішу експлуатацію",
```

```
    "accel_0_100": "швидший розгін покращує динаміку та комфорт при обгонах",
```

```
    "body_fit": "модель відповідає вибраному типу кузова",
```

```
}
```

```
def explain_linear(row: pd.Series, cols: dict, comp_scores: pd.DataFrame, weights: dict):
```

```
    contrib = {}
```

```

if comp_scores is None or not isinstance(comp_scores, pd.DataFrame):
    return {"contrib": {}, "total": 0.0}

```

```

for k, w in weights.items():
    if k not in comp_scores.columns:
        continue
    val = float(comp_scores.loc[row.name, k])
    contrib[k] = val * w

```

```

return {"contrib": contrib, "total": sum(contrib.values())}

```

```

def explain_topsis(row: pd.Series, cols: dict, weights: dict, norm_df: pd.DataFrame):

```

```

    contrib = {}

```

```

    if norm_df is None or not isinstance(norm_df, pd.DataFrame):

```

```

        return {"contrib": {}, "total": 0.0}

```

```

    for k, w in weights.items():

```

```

        if k not in norm_df.columns:

```

```

            continue

```

```

        val = float(norm_df.loc[row.name, k])

```

```

        contrib[k] = val * w

```

```

    return {"contrib": contrib, "total": sum(contrib.values())}

```

```

def pretty_reason(row: pd.Series, cols: dict, explanation: dict) -> str:

```

```

    brand = row.get(cols.get("brand"), "")

```

```

    model = row.get(cols.get("model"), "")

```

```

    price = row.get(cols.get("price"), None)

```

```

    price_str = ""

```

```
if price is not None:
```

```
    try:
```

```
        price_str = f" ({float(price)/1000:.1f}k$)"
```

```
    except:
```

```
        pass
```

```
c = explanation.get("contrib", {})
```

```
if not c:
```

```
    return f"{brand} {model} {price_str}: дані для пояснення відсутні."
```

```
    header = f"###  Деталізація внеску критеріїв для **{brand}
{model} {price_str}**\n"
```

```
lines = []
```

```
for k, v in sorted(c.items(), key=lambda kv: kv[1], reverse=True):
```

```
    label = CRITERIA_LABELS.get(k, k)
```

```
    text = CRITERIA_EXPLANATION.get(k, "критерій вплинув на оцінку")
```

```
    contrib = f"{v:.3f}"
```

```
    if v > 0.08:
```

```
        color = "□ **(СИЛЬНИЙ ВПЛИВ)**"
```

```
    elif v > 0.03:
```

```
        color = "□ **(СЕРЕДНІЙ ВПЛИВ)**"
```

```
    else:
```

```
        color = "○ **(СЛАБКИЙ ВПЛИВ)**"
```

```
    lines.append(f"{color} — {label}:** +{contrib} — {text}")
```

```
block_list = "\n\n".join(lines)
```

```
top_factor = max(c, key=c.get)
top_label = CRITERIA_LABELS.get(top_factor, top_factor)
```

```
paragraph = f"""
```

```
### ✎ Узагальнений висновок
```

```
Модель **{brand} {model} {price_str}** отримала підвищений рейтинг завдяки  
ключовому фактору —
```

```
**{top_label}**, який мав найбільший позитивний внесок.
```

```
Також значний вплив мали:
```

```
**{" , ".join([CRITERIA_LABELS.get(k, k) for k in sorted(c, key=c.get,  
reverse=True)[1:3])]}**.
```

```
Автомобіль демонструє збалансоване поєднання характеристик  
та є сильним конкурентом у межах вимог клієнта.
```

```
"""
```

```
chart_df = pd.DataFrame({
    "Критерій": [CRITERIA_LABELS.get(k, k) for k in c.keys()],
    "Внесок": list(c.values())
}).sort_values("Внесок", ascending=True)
```

```
fig = go.Figure()
fig.add_trace(go.Bar(
    x=chart_df["Внесок"],
    y=chart_df["Критерій"],
    orientation="h",
    marker=dict(color="#1976D2")
```

```
)
```

```
fig.update_layout(
    height=380,
    width=800,
    title="Візуалізація внеску критеріїв",
    margin=dict(l=140, r=40, t=60, b=40),
    xaxis=dict(title="Внесок", fixedrange=True),
    yaxis=dict(title="Критерій", fixedrange=True),
)
```

```
st.plotly_chart(
    fig,
    width="stretch",
    config={
        "scrollZoom": False,
        "doubleClick": False,
        "displayLogo": False,
        "modeBarButtonsToRemove": [
            "zoom", "zoomIn", "zoomOut", "autoScale", "pan",
            "select2d", "lasso2d", "zoom2d", "resetScale2d",
        ],
    }
)
```

```
return header + block_list + "\n\n" + paragraph
```

#### **Б.4 — dss/loader.py**

```
from pathlib import Path
import pandas as pd
```

```

import numpy as np
from .aliases import ALIASES, normalize_column_name, find_column

def load_dataset(path: str):
    """
    Завантажує CSV, нормалізує назви колонок, визначає ключові параметри
    та повертає:
        ✓ DataFrame
        ✓ словник відповідностей основних колонок

    Додатково:
    - Автоматично перетворює числові поля
    - Попереджає про відсутні критично важливі колонки
    """

    p = Path(path)
    if not p.exists():
        raise FileNotFoundError(f"Dataset not found: {p}")

    try:
        df = pd.read_csv(p)
    except Exception as e:
        raise ValueError(f"Не вдалося прочитати CSV: {e}")

    df = df.rename(columns={c: normalize_column_name(c) for c in df.columns})

    cols = {k: find_column(df, v) for k, v in ALIASES.items()}

    numeric_cols = [

```

```

"price", "range_km", "battery_kwh", "eff_whkm",
"fastcharge_kw", "accel_0_100", "seats",
"length_mm", "weight_kg"
]

for k in numeric_cols:
    c = cols.get(k)
    if c:
        df[c] = (
            df[c]
            .astype(str)
            .str.replace(",", ".", regex=False) # 1,23 → 1.23
            .str.replace(" ", "", regex=False) # 1 000 → 1000
        )
        df[c] = pd.to_numeric(df[c], errors="coerce")

REQUIRED = ["brand", "model", "price", "range_km"]
missing = [k for k in REQUIRED if cols.get(k) is None]

if missing:
    print("⚠ УВАГА: У датасеті відсутні критично важливі колонки:", missing)

return df, cols

```

### Б.5 — dss/preprocessing.py

```

import pandas as pd
import numpy as np

def normalize_body(value: str):

```

```

if not isinstance(value, str):
    return None

v = value.strip().lower()

v = v.replace("-", " ").replace("/", " ").replace("_", " ")

mapping = {
    "hatchback": ["hatch", "hatchback"],
    "sedan": ["sedan", "saloon", "liftback"],
    "wagon": ["wagon", "estate", "station"],
    "crossover": ["crossover"],
    "suv": ["suv", "sport utility", "sport activity", "crossover suv"],
    "mpv": ["mpv", "minivan", "multi purpose"],
    "van": ["van", "cargo van"],
    "pickup": ["pickup", "pick up", "ute"],
    "coupe": ["coupe"],
    "convertible": ["convertible", "roadster", "cabrio"],
}

for norm, keys in mapping.items():
    if any(k in v for k in keys):
        return norm

return None

def normalize_drive(value: str):
    if not isinstance(value, str):
        return None

```

```
v = value.lower()

if "awd" in v or "4wd" in v:
    return "awd"
if "fwd" in v:
    return "fwd"
if "rwd" in v:
    return "rwd"

return None

def apply_filters(
    df: pd.DataFrame,
    cols: dict,
    budget_min: float = None,
    budget_max: float = None,
    min_range_km: float = None,
    bodies: list = None,
    seats_min: int = None,
    drive: str = None,
    length_max_mm: float = None,
    weight_max_kg: float = None,
    min_battery_kwh: float = None,
):

    df = df.copy()

    if cols.get("body"):
        df[cols["body"]] = (
            df[cols["body"]]
```

```
.astype(str)
.apply(normalize_body)
)

if cols.get("drive"):
    df[cols["drive"]] = (
        df[cols["drive"]]
        .astype(str)
        .apply(normalize_drive)
    )

mask = pd.Series(True, index=df.index)

if cols.get("price") and (budget_min is not None or budget_max is not None):
    low = float(budget_min) if budget_min is not None else float("-inf")
    high = float(budget_max) if budget_max is not None else float("inf")
    mask &= df[cols["price"]].between(low, high, inclusive="both")

if cols.get("range_km") and min_range_km is not None:
    mask &= df[cols["range_km"]] >= float(min_range_km)

if cols.get("body") and bodies:
    mask &= df[cols["body"]].isin(bodies)

if cols.get("seats") and seats_min is not None:
    mask &= pd.to_numeric(df[cols["seats"]], errors="coerce") >= int(seats_min)

if cols.get("drive") and drive:
    mask &= df[cols["drive"]] == drive
```

```

if cols.get("length_mm") and length_max_mm is not None:
    mask &= pd.to_numeric(df[cols["length_mm"]], errors="coerce") <=
float(length_max_mm)

if cols.get("weight_kg") and weight_max_kg is not None:
    mask &= pd.to_numeric(df[cols["weight_kg"]], errors="coerce") <=
float(weight_max_kg)

if cols.get("battery_kwh") and min_battery_kwh is not None:
    mask &= pd.to_numeric(df[cols["battery_kwh"]], errors="coerce") >=
float(min_battery_kwh)

return df[mask].copy()

```

## B.6 — dss/recommender.py

```

import pandas as pd
from .preprocessing import apply_filters
from .scorer import score_df
from .topsis import topsis_score

```

```

DEFAULT_WEIGHTS = {
    "price": 0.35,
    "range_km": 0.30,
    "fastcharge_kw": 0.15,
    "eff_whkm": 0.10,
    "accel_0_100": 0.05,
    "body_fit": 0.05,
}

```

```

def recommend(

```

```

df: pd.DataFrame,
cols: dict,
filters: dict,
weights: dict = None,
topk: int = 20,
method: str = "linear",
diversify_by_brand: bool = True,
):
    """
    Основна DSS-функція.
    """

    if weights is None:
        weights = DEFAULT_WEIGHTS.copy()

    filtered = apply_filters(
        df, cols,
        budget_min=filters.get("budget_min"),
        budget_max=filters.get("budget_max"),
        min_range_km=filters.get("min_range_km"),
        bodies=None,
        seats_min=filters.get("seats_min"),
        drive=None,
        length_max_mm=filters.get("length_max_mm"),
        weight_max_kg=filters.get("weight_max_kg"),
        min_battery_kwh=filters.get("min_battery_kwh"),
    )

    body_col = cols.get("body")
    body_pref = filters.get("bodies")

```

```
if body_col and body_pref:
    pref = [b.lower() for b in body_pref]
    filtered = filtered[
        filtered[body_col].astype(str).str.lower().isin(pref)
    ]

drive_col = cols.get("drive")
drive_pref = filters.get("drive")

if drive_col and drive_pref:
    d = drive_pref.lower()
    filtered = filtered[
        filtered[drive_col].astype(str).str.lower().eq(d)
    ]

if filtered.empty:
    return filtered.copy()

if method == "topsis":

    minimize = {"price", "eff_whkm", "accel_0_100"}

    scores = topsis_score(filtered, cols, weights, minimize)

    scored = filtered.copy()
    scored["fit_score"] = scores

    scored.attrs["component_scores"] = filtered.attrs.get("component_scores")
```

```

else:
    scored = score_df(
        filtered, cols, weights,
        body_pref=filters.get("bodies"),
        prefer_fastcharge_kw=filters.get("prefer_fastcharge_kw"),
    )

    sort_cols = ["fit_score"]
    ascending = [False]

    price_col = cols.get("price")
    if price_col:
        sort_cols.append(price_col)
        ascending.append(True)

    ranked = scored.sort_values(sort_cols, ascending=ascending)

    if not diversify_by_brand or not cols.get("brand"):
        return ranked.head(topk)

    N = 2
    brands = ranked[cols["brand"]].astype(str)

    result_idx = []
    brand_counts = {}

    for idx, brand in brands.items():
        count = brand_counts.get(brand, 0)
        if count < N:
            result_idx.append(idx)

```

```

    brand_counts[brand] = count + 1
    if len(result_idx) >= topk:
        break

return ranked.loc[result_idx]

```

### Б.7 — dss/scorer.py

```

import pandas as pd
import numpy as np

```

```

def _norm(series: pd.Series, invert: bool = False) -> pd.Series:

```

```

    """

```

```

    Устійке нормування 0..1:

```

- автоматична заміна пропусків на медіану
- захист від нульового діапазону
- optional invert: показники "менше-краще"

```

    """

```

```

    s = pd.to_numeric(series, errors="coerce")

```

```

    if s.isna().any():

```

```

        med = s.median(skipna=True)

```

```

        s = s.fillna(med if np.isfinite(med) else 0)

```

```

    mn, mx = s.min(), s.max()

```

```

    if not np.isfinite(mn) or not np.isfinite(mx) or mn == mx:

```

```

        base = pd.Series(0.5, index=s.index)

```

```

        return (1 - base) if invert else base

```

```

    score = (s - mn) / (mx - mn)

```

```

if invert:
    score = 1.0 - score

return score.clip(0.0, 1.0)

```

```

def score_df(
    df: pd.DataFrame,
    cols: dict,
    weights: dict,
    body_pref: list = None,
    prefer_fastcharge_kw: float = None,
):
    """
    Розраховує лінійний інтегральний бал fit_score з пояснюваністю.
    Повертає DataFrame + додає стовпець 'fit_score'.
    """

    comp = {}

    if cols.get("price") and weights.get("price", 0) > 0:
        norm_price = _norm(df[cols["price"]], invert=True)
        comp["price"] = weights["price"] * norm_price

    if cols.get("range_km") and weights.get("range_km", 0) > 0:
        norm_range = _norm(df[cols["range_km"]])
        comp["range_km"] = weights["range_km"] * norm_range

    if cols.get("fastcharge_kw") and weights.get("fastcharge_kw", 0) > 0:
        raw = pd.to_numeric(df[cols["fastcharge_kw"]], errors="coerce")
        base = _norm(raw)

```

```

if prefer_fastcharge_kw is not None:
    bonus = ((raw >= prefer_fastcharge_kw).fillna(False).astype(float)) * 0.1
    base = (base + bonus).clip(0, 1)

comp["fastcharge_kw"] = weights["fastcharge_kw"] * base

if cols.get("eff_whkm") and weights.get("eff_whkm", 0) > 0:
    norm_eff = _norm(df[cols["eff_whkm"]], invert=True)
    comp["eff_whkm"] = weights["eff_whkm"] * norm_eff

if cols.get("accel_0_100") and weights.get("accel_0_100", 0) > 0:
    norm_acc = _norm(df[cols["accel_0_100"]], invert=True)
    comp["accel_0_100"] = weights["accel_0_100"] * norm_acc

if cols.get("body") and weights.get("body_fit", 0) > 0 and body_pref:
    bodies = df[cols["body"]].astype(str).str.lower()
    match = bodies.apply(
        lambda x: any(pref.lower() in x for pref in body_pref)
    ).astype(float)
    comp["body_fit"] = weights["body_fit"] * match

comp_df = pd.DataFrame(comp)

df_out = df.copy()
df_out["fit_score"] = comp_df.sum(axis=1)

df_out.attrs["component_scores"] = comp_df

return df_out

```

**Б.8 — dss/topsis.py**

```
import numpy as np
import pandas as pd
```

```
def topsis_score(df: pd.DataFrame, cols: dict, weights: dict, minimize: set) -> pd.Series:
```

```
    """
```

```
    Обчислює TOPSIS-бал (0..1).
```

```
    ✓ Підтримка min/max критеріїв
```

```
    ✓ Повне нормування
```

```
    ✓ Збереження нормованої матриці критеріїв для Explainability:
```

```
        df.attrs["component_scores"]
```

```
    """
```

```
    crit_keys = [k for k in weights.keys() if cols.get(k) in df.columns]
```

```
    if not crit_keys:
```

```
        df.attrs["component_scores"] = None
```

```
        return pd.Series(0.0, index=df.index)
```

```
    norm_df = pd.DataFrame(index=df.index)
```

```
    for k in crit_keys:
```

```
        col = cols[k]
```

```
        s = pd.to_numeric(df[col], errors="coerce")
```

```
        med = s.median()
```

```
        s = s.fillna(med)
```

```

mn, mx = s.min(), s.max()
if mx > mn:
    v = (s - mn) / (mx - mn)
else:
    v = pd.Series(0.5, index=s.index)

if k in minimize:
    v = 1.0 - v

norm_df[k] = v

X = norm_df.values.astype(float) # n × m
norm_vec = np.sqrt((X ** 2).sum(axis=0))
R = X / np.where(norm_vec == 0, 1, norm_vec)

w = np.array([weights[k] for k in crit_keys], dtype=float)
w = w / (w.sum() if w.sum() > 0 else 1.0)

V = R * w

ideal_best = V.max(axis=0)
ideal_worst = V.min(axis=0)

d_pos = np.sqrt(((V - ideal_best) ** 2).sum(axis=1))
d_neg = np.sqrt(((V - ideal_worst) ** 2).sum(axis=1))
c = d_neg / np.where((d_pos + d_neg) == 0, 1, (d_pos + d_neg))
score = pd.Series(c, index=df.index)

df_norm_for_explain = norm_df.copy()

```

```
df_norm_for_explain = df_norm_for_explain.reindex(columns=weights.keys(),
fill_value=0)
```

```
df.attrs["component_scores"] = df_norm_for_explain
```

```
return score
```

## Б.9 — dss/validation.py

```
import pandas as pd
```

```
NUMERIC_MUST_BE_NONNEG = [
```

```
    "price", "range_km", "battery_kwh", "fastcharge_kw",
```

```
    "eff_whkm", "accel_0_100", "seats", "length_mm", "weight_kg"
```

```
]
```

```
def profile_dataframe(df: pd.DataFrame, cols: dict) -> dict:
```

```
    """Легка діагностика для розділу 3.2: пропуски, типи, базові діапазони."""
```

```
    report = {"shape": df.shape, "missing": {}, "negatives": {}, "ranges": {}}
```

```
    for k in NUMERIC_MUST_BE_NONNEG:
```

```
        c = cols.get(k)
```

```
        if not c or c not in df.columns:
```

```
            continue
```

```
        s = pd.to_numeric(df[c], errors="coerce")
```

```
        report["missing"][k] = int(s.isna().sum())
```

```
        report["negatives"][k] = int((s < 0).sum())
```

```
        if s.notna().any():
```

```
            report["ranges"][k] = (float(s.min()), float(s.max()))
```

```
    return report
```

```
def assert_minimal_quality(df: pd.DataFrame, cols: dict) -> None:
```

```

"""Мінімальні перевірки — падаємо рано, якщо щось критичне відсутнє."""
required = ["brand", "model", "price", "range_km"]
missing = [r for r in required if not cols.get(r)]
if missing:
    raise ValueError(f"Відсутні критично важливі стовпці: {missing}")

```

### Б.10 — app\_streamlit.py

```

import streamlit as st
import pandas as pd
import tempfile

from dss.loader import load_dataset
from dss.recommender import recommend, DEFAULT_WEIGHTS
from dss.explain import explain_linear, explain_topsis, pretty_reason

st.set_page_config(page_title="DSS підбору електромобілів", layout="wide")

st.markdown("""
<style>
[data-testid="stSidebar"] {
    width: 380px !important;
}
section[data-testid="stSidebar"] > div {
    width: 380px !important;
}

input[type=number] {
    -moz-appearance: textfield;
}
input[type=number]::-webkit-inner-spin-button,

```

```
input[type=number]::-webkit-outer-spin-button {
  -webkit-appearance: none;
  margin: 0;
}
</style>
```

```
<script>
document.addEventListener('wheel', function(e){
  const el = document.activeElement;
  if (!el) return;
  const tag = el.tagName.toLowerCase();
  const type = (el.getAttribute('type') || "").toLowerCase();
  if (tag === 'input' && (type === 'number' || type === 'range')) {
    el.blur();
  }
}, { passive: true });
</script>
""", unsafe_allow_html=True)
```

```
st.title("Інтелектуальна DSS-система підбору електромобілів для клієнтів  
автосалону")
```

```
st.markdown("""
Система допомагає підібрати оптимальні електромобілі за бюджетом,
технічними характеристиками, типом кузова, швидкістю зарядки та іншими
параметрами.
Методи оцінювання: Linear (WSM) та TOPSIS.
""")
```

```
BODY_TRANSLATE = {
```

```

    "hatchback": "хетчбек",
    "sedan": "седан",
    "wagon": "універсал",
    "crossover": "кросовер",
    "suv": "SUV/позашляховик",
    "mpv": "мінівен (MPV)",
    "van": "вантажопасажирський VAN",
    "pickup": "пікап",
    "coupe": "купе",
    "convertible": "кабриолет",
}
BODY_REVERSE = {v: k for k, v in BODY_TRANSLATE.items()}

DRIVE_TRANSLATE = {
    "awd": "повний привід (AWD)",
    "fwd": "передній привід (FWD)",
    "rwd": "задній привід (RWD)",
}
DRIVE_REVERSE = {v: k for k, v in DRIVE_TRANSLATE.items()}

st.sidebar.header("1) Дані")

uploaded = st.sidebar.file_uploader(
    "Завантажте CSV з характеристиками електромобілів",
    type=["csv"]
)

default_path = "electric_vehicles_spec_2025.csv"
path = default_path

```

if uploaded is not None:

```
tmp = tempfile.NamedTemporaryFile(delete=False, suffix=".csv")
tmp.write(uploaded.getbuffer())
tmp.close()
path = tmp.name
```

```
df, cols = load_dataset(path)
```

with st.expander("Переглянути завантажені дані"):

```
st.dataframe(df.head(20), width="stretch")
```

```
st.sidebar.header("2) Фільтри")
```

```
budget = st.sidebar.slider("Бюджет ($)", 0, 200000, (20000, 60000), 1000)
```

```
min_range = st.sidebar.slider("Мінімальний WLTP запас ходу (км)", 0, 1200, 300,
10)
```

```
bodies_ui = st.sidebar.multiselect(
```

```
    "Бажані типи кузова:",
```

```
    list(BODY_TRANSLATE.values()),
```

```
    default=[BODY_TRANSLATE["crossover"], BODY_TRANSLATE["suv"]]
)
```

```
bodies = [BODY_REVERSE[b] for b in bodies_ui]
```

```
prefer_dc = st.sidebar.slider("Мінімальна потужність DC-зарядки (кВт)", 0, 400, 120,
10)
```

```
seats_min = st.sidebar.number_input("Мінімальна кількість місць", 0, 9, 0)
```

```
drive_ui = st.sidebar.selectbox(
```

```
    "Тип приводу:",
```

```

    ["будь-який"] + list(DRIVE_TRANSLATE.values())
)
drive = DRIVE_REVERSE.get(drive_ui)

min_batt = st.sidebar.number_input("Мінімальна ємність батареї (кВт·год)", 0, 200,
0)
max_length = st.sidebar.number_input("Максимальна довжина (мм)", 0, 6000, 0)
max_weight = st.sidebar.number_input("Максимальна маса (кг)", 0, 4500, 0)

filters = dict(
    budget_min=budget[0],
    budget_max=budget[1],
    min_range_km=min_range,
    bodies=bodies,
    prefer_fastcharge_kw=prefer_dc,
    seats_min=seats_min,
    drive=drive,
    min_battery_kwh=min_batt,
    length_max_mm=max_length if max_length > 0 else None,
    weight_max_kg=max_weight if max_weight > 0 else None,
)

st.sidebar.header("3) Ваги критеріїв")

weights = DEFAULT_WEIGHTS.copy()
for k, label in [
    ("price", "Ціна (менше — краще)"),
    ("range_km", "Запас ходу"),
    ("fastcharge_kw", "Швидка DC-зарядка"),
    ("eff_whkm", "Енергоефективність (менше — краще)"),

```

```

("accel_0_100", "Розгін 0–100 км/год (менше — краще)",
("body_fit", "Відповідність типу кузова"),
]:
weights[k] = st.sidebar.slider(label, 0.0, 1.0, float(weights[k]), 0.01)

s = sum(weights.values())
if s > 0:
    weights = {k: v / s for k, v in weights.items()}

st.sidebar.header("4) Метод оцінювання")
method = st.sidebar.selectbox("Оберіть метод:", ["linear", "topsis"])

st.header("Рекомендовані моделі")

topk = st.number_input("Кількість моделей (Тор-К):", 5, 50, 20)

result = recommend(
    df, cols, filters,
    weights=weights,
    topk=int(topk),
    method=method,
    diversify_by_brand=True
)

if len(result) == 0:
    st.info("""
### Моделей не знайдено за вказаними фільтрами
Показуємо **найбільш схожі варіанти** з автоматичним послабленням умов:
- без обмеження по кузову
- без обмеження по приводу

```

- запас ходу знижено на 20%  
 - бюджет розширено на  $\pm 20\%$   
 """)

```
relaxed = dict(filters)
relaxed["bodies"] = None
relaxed["drive"] = None
```

```
if relaxed.get("min_range_km"):
    relaxed["min_range_km"] = max(0, int(relaxed["min_range_km"] * 0.8))
```

```
span = filters["budget_max"] - filters["budget_min"]
relaxed["budget_min"] = max(0, int(filters["budget_min"] - span * 0.2))
relaxed["budget_max"] = int(filters["budget_max"] + span * 0.2)
```

```
result = recommend(
    df, cols, relaxed,
    weights=weights,
    topk=int(topk),
    method=method,
    diversify_by_brand=True
)
```

```
pretty = result.copy()
pretty_cols = {}
```

```
if cols.get("brand"):
    pretty_cols[cols["brand"]] = "Виробник"
```

```
if cols.get("model"):
```

```

pretty_cols[cols["model"]] = "Модель"

if cols.get("price"):
    pretty_cols[cols["price"]] = "Ціна ($)"

if cols.get("range_km"):
    pretty_cols[cols["range_km"]] = "Запас ходу (км)"

if cols.get("fastcharge_kw"):
    pretty_cols[cols["fastcharge_kw"]] = "DC-зарядка (кВт)"

if cols.get("eff_whkm"):
    pretty_cols[cols["eff_whkm"]] = "Wh/км"

if cols.get("accel_0_100"):
    pretty_cols[cols["accel_0_100"]] = "0–100 (с)"

pretty_cols["fit_score"] = "Індекс відповідності"

pretty = pretty.rename(columns={k: v for k, v in pretty_cols.items() if k in
pretty.columns})

st.dataframe(pretty.reset_index(drop=True), width="stretch")

csv = pretty.to_csv(index=False).encode("utf-8")
st.download_button("Завантажити результати (CSV)", csv,
"ev_recommendations.csv")

if len(result) > 0:

```

```

st.subheader("Пояснення прийнятого рішення")

comp = result.attrs.get("component_scores")

if len(result) == 1:
    explain_idx = 1
else:
    explain_idx = st.number_input(
        "Оберіть номер моделі (1 — перший рядок):",
        min_value=1,
        max_value=len(result),
        value=1,
        step=1
    )

row = result.iloc[int(explain_idx) - 1]

if method == "linear":
    exp = explain_linear(row, cols, comp, weights)
    st.markdown("### Деталізація внеску критеріїв")
    st.markdown(pretty_reason(row, cols, exp))

elif method == "topsis":
    exp = explain_topsis(row, cols, weights, comp)
    st.markdown("### Деталізація внеску критеріїв")
    st.markdown(pretty_reason(row, cols, exp))

with st.expander("Методика"):
    st.markdown("""
### **Лінійна модель (WSM)**

```

$\Sigma$  (вага  $\times$  нормалізоване значення критерію)

### \*\*TOPSIS\*\*

- нормування
- ідеальна та анти-ідеальна точка
- знаходження відстані до еталонів

### \*\*Диверсифікація брендів\*\*

Не більше \*\*2 моделей одного бренду\*\*.

""")

### **Б.11 — requirements.txt**

streamlit

pandas

numpy

plotly

scikit-learn

### **Б.12 — run\_dss.bat**

@echo off

setlocal ENABLEDELAYEDEXPANSION

REM -----

REM Увімкнути UTF-8

REM -----

chcp 65001 > nul

echo =====

echo DSS запуск (UTF-8 MODE)

echo =====

```
REM -----
REM Перейти у директорію запуску
REM -----
cd /d "%~dp0"

REM -----
REM Перевірка Python
REM -----
python --version > nul 2>&1
if errorlevel 1 (
    echo ✘ Python не знайдено!
    echo Встановіть Python 3.10+ і додайте до PATH.
    echo -----
    pause
    exit /b
)

REM -----
REM Створення venv, якщо немає
REM -----
if not exist venv (
    echo ✔ Створюю віртуальне середовище...
    python -X utf8 -m venv venv
)
REM -----
REM Активація venv
REM -----
echo ✔ Активую середовище...
```

```

call venv\Scripts\activate
if errorlevel 1 (
    echo ✘ Не вдалося активувати venv.
    echo -----
    pause
    exit /b
)

REM -----
REM Оновлення pip
REM -----
echo ✔ Оновлюю pip...
python -m pip install --upgrade pip
REM -----
REM Встановлення залежностей
REM -----
echo ✔ Встановлюю залежності...
pip install -r requirements.txt
REM -----
REM Запуск Streamlit
REM -----
echo =====
echo ✔ Запускаю DSS систему
echo =====
python -X utf8 -m streamlit run app_streamlit.py
REM -----
REM БЛОКУЄМО закриття вікна у ВСІХ випадках
REM -----
echo -----

```

echo 🔄 Роботу завершено або сталася помилка.

echo ✓ Натисніть будь-яку клавішу для виходу...

echo -----

pause > nul

exit /b

## Додаток В. Інтерфейс інтелектуальної DSS-системи підбору електромобілів

1) Дані

Завантажте CSV з характеристиками електромобілів

Drag and drop file here  
Limit 200MB per file • CSV  
Browse files

2) Фільтри

Бюджет (\$) 20000 60000

Мінімальний WLTP запас ходу (км) 300

Бажані типи кузова:  
кроссовер x  
SUV/попашляхо... x

Мінімальна потужність DC-зарядки (кВт) 120

Мінімальна кількість місць 0

Тип приводу: будь-який

Мінімальна ємність батареї (кВт·год) 0

Максимальна довжина (мм) 0

Максимальна маса (кг) 0

3) Ваги критеріїв

Ціна (менше — краще) 0.35

### Інтелектуальна DSS-система підбору електромобілів для клієнтів автосалону

Система допомагає підібрати оптимальні електромобілі за бюджетом, технічними характеристиками, типом кузова, швидкістю зарядки та іншими параметрами.  
Методи оцінювання: Linear (WSM) та TOPSIS.

> Переглянути завантажені дані

### Рекомендовані моделі

Кількість моделей (Топ-K): 20

	Виробник	Модель	top_speed_kmh	battery_capacity_kwh	battery_type	number_of_cells	torque_nm	Wh/км	Запас ходу (км)	0-100 (с)	DC-зарядка (кВт)	fast_charge_port	towing_capacity_kg	cargo_volume_l	seats	drivetrain	segment	length_mm	width_mm	height
0	Peugeot	e-408 58 kWh	160	58.3	Lithium-ion	None	345	129	375	7.2	70	CCS	None	471	5	fwd	JD - Large	4687	1859	
1	Skoda	Enyaq Coupe 60	160	59	Lithium-ion		310	141	370	8.1	110	CCS	1000	570	5	rwd	JC - Medium	4658	1879	
2	Hyundai	INSTER Long Range	150	46	Lithium-ion	None	147	124	300	10.6	70	CCS	None	280	4	fwd	JA - Mini	3825	1610	
3	Alfa	Romeo Junior Elettrica 54 kWh	150	50.8	Lithium-ion	102	260	128	320	9	85	CCS	0	400	5	fwd	JB - Compact	4173	1781	
4	Skoda	Enyaq 60	160	59	Lithium-ion	216	310	145	360	8.1	110	CCS	1000	585	5	rwd	JC - Medium	4658	1879	
5	Fiat	600e	150	50.8	Lithium-ion	102	260	124	310	9	79	CCS	0	360	5	fwd	JB - Compact	4171	1781	
6	Hyundai	IONIQ 5 63 kWh RWD (MY24)	185	60	Lithium-ion	288	350	136	345	8.5	150	CCS	750	520	5	rwd	JC - Medium	4655	1890	
7	Kia	EV6 Standard Range 2WD	185	60	Lithium-ion	288	350	140	345	8.7	150	CCS	750	490	5	rwd	JC - Medium	4695	1880	
8	Jeep	Avenger Electric	150	50.8	Lithium-ion	102	260	132	310	9	85	CCS	0	355	5	fwd	JB - Compact	4084	1776	
9	Renault	4 E-Tech 52kWh 150hp	150	52	Lithium-ion	184	245	127	315	8.2	70	CCS	750	420	5	fwd	JB - Compact	4143	1796	

Завантажити результати (CSV)

### Пояснення прийнятого рішення

Оберіть номер моделі (1 — перший рядок): 1

### Деталізація внеску критеріїв

Візуалізація внеску критеріїв

Запас ходу [Bar chart]

Ціна [Bar chart]

Рисунок В.1 – Головне вікно DSS-системи підбору електромобілів

## 1) Дані

Завантажте CSV з характеристиками електромобілів

Drag and drop file here

Limit 200MB per file • CSV

Browse files

Рисунок В.2 – Завантаження вхідних даних у форматі CSV

Переглянути завантажені дані

	brand	model	top_speed_kmh	battery_capacity_kwh	battery_type	number_of_cells	torque_nm	efficiency_wh_per_km	range_km	acceleration_0_100_s	fast_charging_power_kw_dc	fast_charge_port	towing_capacity_kg	cargo_volume_l	seats	drivetrain	segment
0	Abarth	500e Convertible	155	37.8	Lithium-ion	192	235	156	225	7	67	CCS	0	185	4	FWD	B - Con
1	Abarth	500e Hatchback	155	37.8	Lithium-ion	192	235	149	225	7	67	CCS	0	185	4	FWD	B - Con
2	Abarth	600e Scorpionissima	200	50.8	Lithium-ion	102	345	158	280	5.9	79	CCS	0	360	5	FWD	JB - Co
3	Abarth	600e Turismo	200	50.8	Lithium-ion	102	345	158	280	6.2	79	CCS	0	360	5	FWD	JB - Co
4	Always	US	150	60	Lithium-ion	None	310	156	315	7.5	78	CCS	None	496	5	FWD	JC - Me
5	Always	US	160	60	Lithium-ion	None	315	150	350	7	78	CCS	None	472	5	FWD	JC - Me
6	Alfa	Romeo Junior Elettrica 54 kWh	150	50.8	Lithium-ion	102	260	128	320	9	85	CCS	0	400	5	FWD	JB - Co
7	Alfa	Romeo Junior Elettrica 54 kWh Veloce	200	50.8	Lithium-ion	102	345	164	310	6	85	CCS	0	400	5	FWD	JB - Co
8	Alpine	A290 Electric 180 hp	160	52	Lithium-ion	184	285	138	310	7.4	70	CCS	500	326	5	FWD	B - Con
9	Alpine	A290 Electric 220 hp	170	52	Lithium-ion	184	300	144	305	6.4	70	CCS	500	326	5	FWD	B - Con

Рисунок В.3 – Перегляд завантажених даних після імпорту CSV

### 2) Фільтри

Бюджет (\$)  
20000 60000

Мінімальний WLTP запас ходу (км)  
300

Бажані типи кузова:  
SUV/позашляхо... x

Мінімальна потужність DC-зарядки (кВт)  
120

Мінімальна кількість місць  
1 - +

Тип приводу:  
повний привід (AWD) v

Мінімальна ємність батареї (кВт·год)  
10 - +

Максимальна довжина (мм)  
5000 - +

Максимальна маса (кг)  
4000 - +

Рисунок В.4 – Панель фільтрів для налаштування параметрів відбору

### 3) Ваги критеріїв

Ціна (менше — краще)  
0.30

Запас ходу  
0.30

Швидка DC-зарядка  
0.15

Енергоефективність (менше — краще)  
0.10

Розгін 0–100 км/год (менше — краще)  
0.24

Відповідність типу кузова  
0.20

Рисунок В.5 – Вибір вагових коефіцієнтів критеріїв оцінювання

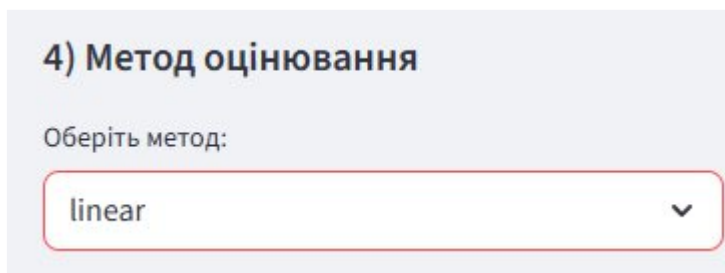


Рисунок В.6 – Вибір методу оцінювання альтернатив

## Рекомендовані моделі

Кількість моделей (Топ-К):

0	Виробник	Модель	top_speed_kmh	battery_capacity_kwh	battery_type	number_of_cells	torque_nm	Wh/km	Запас ходу (км)	0-100 (с)	DC-зарядка (кВт)	fast_charge_port	towing_capacity_kg	cargo_volume_l	seats	drivetrain	segment	length_mm	width_mm	height_mm	car_body
1	Smart	#3 Brabus	180	62	Lithium-ion	None	543	149	335	3.7	100	CCS	1600	370	5	awd	JB-Compact	4400	1844	1556	suv
2	Smart	#1 Pulse	180	62	Lithium-ion	None	584	155	325	4.5	100	CCS	1600	313	5	awd	JB-Compact	4300	1822	1636	suv
3	Toyota	bZ4X AWD	160	64	Lithium-ion	96	336	154	345	6.9	100	CCS	750	452	5	awd	JC-Medium	4690	1860	1600	suv
4	Subaru	Solterra AWD	160	64	Lithium-ion	96	336	154	320	6.9	100	CCS	750	441	5	awd	JC-Medium	4690	1860	1650	suv

Завантажити результати (CSV)

Рисунок В.7 – Таблиця рекомендованих моделей електромобілів

## Пояснення прийнятого рішення

Оберіть номер моделі (1 — перший рядок):

1

## Деталізація внеску критеріїв

## Візуалізація внеску критеріїв

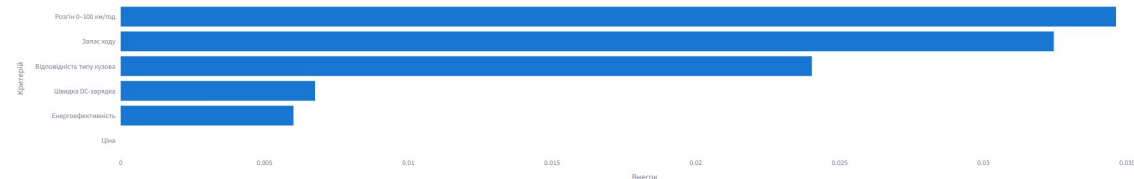


Рисунок В.8 – Візуалізація внеску критеріїв у кінцеву оцінку

### Деталізація внеску критеріїв для Smart #3 Brabus (59.6k\$)

- (середній вплив) — Розгін 0–100 км/год: +0.035 — швидкий розгін покращує динаміку та комфорт при обгонах
- (середній вплив) — Запас ходу: +0.032 — великий запас ходу підвищує практичність авто у щоденному та міжміському використанні
- (слабкий вплив) — Відповідність типу кузова: +0.024 — модель відповідає вибраному типу кузова
- (слабкий вплив) — Швидка DC-зарядка: +0.007 — вища потужність DC-зарядки зменшує час поповнення енергії
- (слабкий вплив) — Енергоефективність: +0.006 — менше споживання енергії означає економічнішу експлуатацію
- (слабкий вплив) — Ціна: +0.000 — нижча вартість підвищує загальну привабливість моделі

### Узагальнений висновок

Модель Smart #3 Brabus (59.6k\$) отримала підвищений рейтинг завдяки ключовому фактору — Розгін 0–100 км/год, який мав найбільший позитивний внесок.

Також значний вплив мали:

**Запас ходу, Відповідність типу кузова.**

Автомобіль демонструє збалансоване поєднання характеристик та є сильним конкурентом у межах вимог клієнта.

Рисунок В.9 – Пояснення прийнятого рішення для обраної моделі

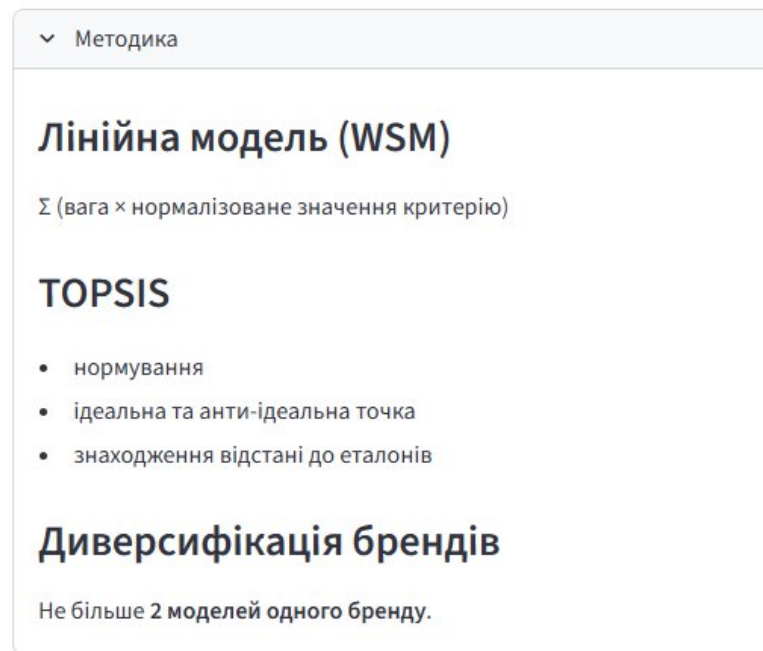


Рисунок В.10 – Вікно методики оцінювання у DSS-системі