

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерних наук _____
 Кафедра _____ програмної інженерії _____
 Рівень вищої освіти _____ другий (магістерський) _____
 Спеціальність _____ 121 – Інженерія програмного забезпечення _____
 Тип програми _____ освітньо-наукова програма _____
 Освітня програма _____ Інженерія програмного забезпечення _____
 (шифр і назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
 (підпис)
 «____» _____ 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові _____ Павленко Ярославу Володимировичу _____
 (прізвище, ім'я, по батькові)

1. Тема роботи «Дослідження методів оцінки моделі на основі параметрів на back-end частині»

Затверджена наказом по університету від 29.03.2024р. № 250 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 15.06.2024

3. Вихідні дані до роботи опис моделі на основі параметрів, опис та реалізація досліджуваних методів розрахунку оцінки моделі, опис реалізації методів різними способами, опис рекомендацій вибору методів оцінки, мова програмування JavaScript, технології Node.js, Android Studio, середовище розробки Visual Studio Code

4. Перелік питань, що потрібно опрацювати в роботі аналіз моделі на основі параметрів, аналіз та порівняння методів оцінки моделі, реалізація методів різними способами, опис проблематики для проведення експериментів, розробка методів оцінки, проведення експериментів, аналіз результатів експериментів та аналіз отриманих результатів

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз предметної галузі та постановка задачі	29.03 – 14.04.24	<i>виконано</i>
2	Аналіз та вибір методів для дослідження	15.04 – 24.04.24	<i>виконано</i>
3	Опис методів оцінки та моделювання предметної області	25.04 – 28.04.24	<i>виконано</i>
4	Планування експериментів	29.04 – 02.05.24	<i>виконано</i>
5	Програмна реалізація обраних методів оцінки	03.04 – 09.05.24	<i>виконано</i>
6	Експериментальні дослідження	10.05 – 24.05.24	<i>виконано</i>
7	Аналіз результатів експериментальних досліджень та розробка рекомендацій	25.05 – 28.05.24	<i>виконано</i>
8	Написання та оформлення статті та тез доповіді	21.04 – 30.04.24	<i>виконано</i>
9	Підготовка пояснювальної записки	31.05 – 01.06.24	<i>виконано</i>
10	Підготовка презентації та доповіді	02.06 – 05.06.24	<i>виконано</i>
11	Нормоконтроль	8.06 – 10.06.24	<i>виконано</i>
12	Рецензування	12.06 – 15.06.24	<i>виконано</i>
13	Занесення диплома в електронний архів	15.06.2024	<i>виконано</i>
14	Попередній захист	15.06.2024	<i>виконано</i>
15	Допуск до захисту у зав. кафедри	17.06.2024	<i>виконано</i>

Дата видачі завдання 29 березня 2024р.

Студент _____

(підпис)

_____ Павленко Я. В.

Керівник роботи _____

(підпис)

_____ доц. Валенда Н. А.

(посада, прізвище, ініціали)

РЕФЕРАТ / ABSTRACT

Пояснювальна записка містить 60 стор., 36 рис., 9 табл., 12 джерел.

ВАГОВІ КОЕФІЦІЕНТИ, МЕТОДИ ОЦІНКИ, ПАРАМЕТРИ МОДЕЛІ, ТЕХНОЛОГІЇ СЕРВЕРНОЇ ЧАСТИНИ, JAVASCRIPT, MYSQL.

Об'єкт дослідження – методи розрахунку оцінки моделі на основі параметрів.

Мета роботи – дослідження методів розрахунку загальної оцінки моделі на основі параметрів, дослідження способів реалізації алгоритмів на back-end частині з використанням необхідних технологій.

Результат роботи – реалізована back-end частина та основний функціонал для дослідження методів оцінки, реалізовані методи дослідження, створений веб-застосунок для відображення результатів виконання методів оцінки, реалізовані основні аспекти системи на веб застосунку, розроблений мобільний застосунок для відображення та виконання результатів методів оцінки моделі та основних оцінок параметрів системи.

WEIGHT COEFFICIENT, METHODS OF ASSESSMENT, PARAMETERS OF A MODEL, SERVER PART TECHNOLOGIES, JAVASCRIPT, MYSQL.

Object of research – methods and means the parameter-based model evaluation.

Purpose – Researching methods for calculating the overall estimate of a parameter-based model, researching ways to implement algorithms on the back-end part using the necessary technologies.

Research summary – implemented back-end part and main evaluation method research functionality, research method successfully implemented, web-application, designed for evaluation method execution results representation, main aspects of the system have been embodied on the web application, mobile application for display and execution of the results of evaluation model methods and main system parameters

assessment has been created.

Заява щодо самостійного виконання кваліфікаційної роботи та можливості її публікації в електронному архіві відкритого доступу EIArKhNURE.

Я, Павленко Ярослав Володимирович, студент гр. ПЗм-22-4, здобувач вищої освіти на другому (магістерському) рівні кафедри «Програмна інженерія», заявляю: моя кваліфікаційна робота на тему «Дослідження методів оцінки моделі на основі параметрів на back-end частині», що буде представлена в екзаменаційну комісію для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу EIArKhNURE. Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлений з діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

ЗМІСТ

Вступ.....	7
1 Аналіз предметної області.....	9
1.1 Аналіз предметної галузі	9
1.2 Постановка задачі.....	10
1.3 Створення завдання дослідження.....	10
1.4 Технології проведення дослідження.....	11
2 Аналіз існуючих методів	13
2.1 Аналіз методів оцінки	13
2.2 Аналіз способів реалізації методів.....	17
2.3 Очікувані результати.....	18
3 Підготовка до проведення дослідження та реалізація.....	19
3.1 Підготовка технологій для дослідження.....	19
3.2 Вибір технологій back-end частини.....	19
3.3 Структура компонентів дослідження.....	25
3.4 Створення UI/UX та реалізація методів у системі.....	28
4 Порівняльний аналіз.....	34
4.1 Порівняльний аналіз методів.....	34
Висновки.....	43
Перелік джерел посилання.....	45
Додаток А Перелік джерел посилання за науковими напрямками керівника та науковців кафедри програмної інженерії.....	46
Додаток Б Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ.....	47
Додаток В Слайди презентації.....	48
Додаток Г Апробація результатів роботи.....	57
Додаток Д Експертний висновок результатів перевірки кваліфікаційної роботи на відповідальність оформлення вимогам ДСТУ 3008: 2015.....	60

ВСТУП

У сучасному світі все більш інтенсивно розвиваються та автоматизуються різноманітні інформаційні системи. Інформаційні технології не перестають впроваджуватись в різноманітні галузі життєдіяльності людини, починаючи від транспортних систем і охорони природи до соціальної та космічної сфер. Кожне виробництво зацікавлене у інтеграції нових швидкодієвих систем у свої процеси. Такі дії виникають завдяки тому, що галузь ІТ значно поліпшує та спрощує працю.

Також, окремим трендом запроваджується інтеграція смарт пристроїв у різні процеси моніторингу починаючи від систем охорони та систем розпізнавання образів до систем моніторингу та датчиків фіксування оцінки показників навколишнього середовища. Такі інформаційні пристрої генерують та моніторять велику кількість даних, зокрема числових, в тому числі, оцінки багатьох різних показників з різних предметних областей використання. Зі збільшенням інформації виникає потреба в автоматизованому аналізі даних [1]. В залежності від типу інформації потрібно застосовувати різні методи. Наприклад, для обробки текстових даних потрібно вирішити окремі проблеми, а для числових – інші. В задачах обробки природньої мови є проблеми пов'язані з класифікацією тексту [2]. Також, в задачах пов'язаних з автоматизацією обробки текстових даних є відомі методи, які спрощуть цей процес [3]. В свою чергу, для методів оцінки моделі потрібно вирішити інші проблеми: втрати даних та розрахування оцінки моделі в такій ситуації, а також дослідити певні методи пов'язані з обробкою числових даних.

Окрім, впровадження процесів збору числових даних про систему, за допомогою використання смарт пристроїв, використовуються й інші способи збору числових даних та оцінок показників та параметрів системи. Такі способи можна використовувати в тих випадках, коли деякі показники неможливо оцінити завдяки відповідних датчиків. До таких способів можна віднести експертну оцінку або модераторську або користувачьку оцінку стосовно параметрів розглядаємої системи.

Стрімке накопичення таких числових даних та оцінок параметрів різних систем дозволяє розглядати методи оцінки та прогнозування загальної оцінки всієї системи. Таким чином, зростає актуалізація методів та способів розрахунку загальної оцінки системи на основі всіх або деякої частини наданих окремих оцінок її параметрів. При розрахунку загальної оцінки, також, треба враховувати, те що деякі показники можуть незначно впливати на всю систему, а деякі навпаки значно впливають, тому слід розглядати модель, яка включає параметри системи та їх вагові коефіцієнти і досліджувати вже методи розрахунку оцінки такої моделі.

До проблематики методів оцінки моделі можна віднести нестабільне обчислення при відсутності деяких даних про власні оцінки параметрів. Наприклад, деякі методи можуть обчислювати загальну оцінку моделі більш стабільно при можливому неповному об'ємі числових даних про параметри моделі.

Отже, в ході дослідження необхідно протестувати та порівняти результати обчислення методів на різних даних, в тому числі з неповними даними про параметри [4].

Цілью дослідження є визначення кращих та доцільних методів оцінки моделі в різних ситуаціях, в тому числі, при відсутності частини даних про оцінки параметрів моделі. Отримані результати дослідження можна буде використати в різних сферах для оцінки різноманітних об'єктів на основі інформації про їх параметри. Таким чином, це може значно покращити автоматизацію аналізу числових даних.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз предметної галузі

Оскільки, в світі збільшується кількість різноманітних даних з різних галузей та об'єктів, то також збільшується актуальність врахування цієї інформації для оцінки об'єктів. Також, стрімко зростає необхідність обрання швидких та правильно працюючих методів та інструментів для розрахунку такої загальної оцінки моделі.

Надання оцінки моделі системи потрібно та корисно з різних причин. Наприклад, на основі загальної оцінки про розглядаємий об'єкт можна приймати різноманітні рішення щодо нього та прогнозувати майбутні дії. Спектр прийняття рішень залежить безпосередньо від області в якій застосована модель параметрів показників та їх вагових коефіцієнтів.

Розглядаєму модель можна застосувати до багатьох областей життєдіяльності та науки. Наприклад, для сфери оцінки рівня екології природних об'єктів. В такому разі їхніми параметрами будуть слугувати: вплив глобального потепління, популярність та відвідуваність природнього об'єкту, охорона природного об'єкту та інші. Такі параметри можуть оцінювати експерти екологи. А параметрами, які будуть отримуватися з смарт пристроїв та датчиків, можуть бути, наприклад: індекс забруднення води, оцінка вологості, температура, кількість важких металів у повітрі та інші. Також, розглядаєму модель можна застосовувати і в інших областях, наприклад, в сфері охорони, медицини, транспорту та інших. Зрозуміло, що в різних галузях в моделі будуть присутні параметри на основі яких оцінюються аспекти конкретної обраної сфери.

Для проведення дослідження методів та способів оцінки доцільно обрати сферу застосування для розглядаємої моделі. Тому, під час дослідження методів оцінки застосуємо модель до сфери оцінки загального рівня екології природнього об'єкту.

Також, дослідження методів оцінки актуально проводити на технологіях back-end. Тому що, переважна більшість числових даних збирається в базу даних, а вже потім за допомогою різних алгоритмів оцінки та різними способами

обчислюється саме на back-end частині.

Таким чином, можна провести у повному обсязі дослідження стосовно методів та способів оцінки моделі, яка базується на параметрах та вагових коефіцієнтах.

1.2 Створення завдання магістерського дослідження

Для створення завдання потрібно проаналізувати додаткові цілі роботи. Також, необхідно підготувати інформацію про технології проведення дослідження та зібрати дані стосовно технологій.

Під час магістерського дослідження потрібно виконати додаткові цілі:

- оцінити параметри з галузі застосування моделі або дати інформацію, на основі якої можна оцінити їх;
- знайти інформацію стосовно потенційного впливу параметрів на загальну оцінку;
- інтегрувати оцінки у модель завдання;
- реалізувати методи оцінки на back-end частині;
- описати процес створення методів на back-end частині у записці дослідження;
- описати у записці магістерського дослідження процес реалізації методів різними способами;
- надати інформацію про технології проведення дослідження.

Після опису завдання дослідження постановки задачі, перейдемо до опису технологій проведення дослідження,

1.3 Технології проведення дослідження

Реалізація дослідження буде проводитись на різних технологіях та мові програмування JavaScript [5].

Для розробки програмного коду буде використано середовище Visual Studio Code.

Під час проведення дослідження будуть використані наступні технології:

Express.js – це програмний каркас застосунків. Фактично представляє собою бібліотеку, яка реалізована як вільне та відкрите програмне забезпечення. Каркас спроектований для створення застосунків та API. Активно використовується на back-end частинах. Під час проведення дослідження буде використовуватися для реалізації методів оцінки на серверній частині.

MySQL – система керування базами даних. Ця система має подвійне ліцензування. У ході проведення магістерського дослідження буде використовуватися для зберігання даних стосовно оцінок параметрів, а також їх вагових коефіцієнтів [6].

Math.js – це бібліотека для JavaScript. Вона переважно застосовується для математичних операцій. Містить багато реалізованих та цікавих математичних рішень. Під час дослідження буде застосовуватися для реалізації методів.

Lodash – це бібліотека JavaScript. Її можна застосувати для роботи із структурами даних. Під час магістерського дослідження буде використовуватись для реалізації методів на back-end частині.

Результати реалізації методів та способів обчислення загальної оцінки буде проаналізовано критеріально.

Отже, було наведено завдання дослідження та обрано технології. Тепер перейдемо до теоретичної частини дослідження.

1.4 Постановка задачі

До головної цілі роботи відноситься проведення дослідження методів оцінки. Основою проведення дослідження є дослідження методів оцінок моделі. Тобто перевірка, реалізація та порівняння методів та способів оцінки моделі з урахуванням оцінок параметрів та їх вагових коефіцієнтів, з метою виявлення кращих та доцільних із них.

Для проведення дослідження потрібно реалізувати такі завдання:

- створити та проаналізувати складові завдання;
- надати основну інформацію стосовно розглядаємої моделі;
- опрацювати та розібрати інформацію для дослідження моделі та галузі її

- застосування – екології природних об'єктів;
- надати основну інформацію стосовно методів оцінки моделі;
 - надати основну інформацію стосовно способів застосування методів на back-end частині;
 - обрати технології back-end для проведення дослідження;
 - реалізувати деякі методи оцінки моделі за допомогою різних способів на back-end частині;
 - зробити порівняльний аналіз результатів дослідження і порівняти методи оцінки моделі критеріально;
 - створити детально обґрунтований висновок результату проведеного дослідження.

Після постановки задачі, наступним кроком перейдемо до аналізу існуючих методів та підготовки до проведення дослідження.

2 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ

2.1 Аналіз методів оцінки

В ході дослідження потрібно дослідити методи загальної оцінки моделі на основі параметрів. Ця модель представляє собою список оцінок параметрів природнього об'єкту та список ваги кожного параметру у вигляді певного коефіцієнту.

Приклад моделі зображено на рисунку 2.1.

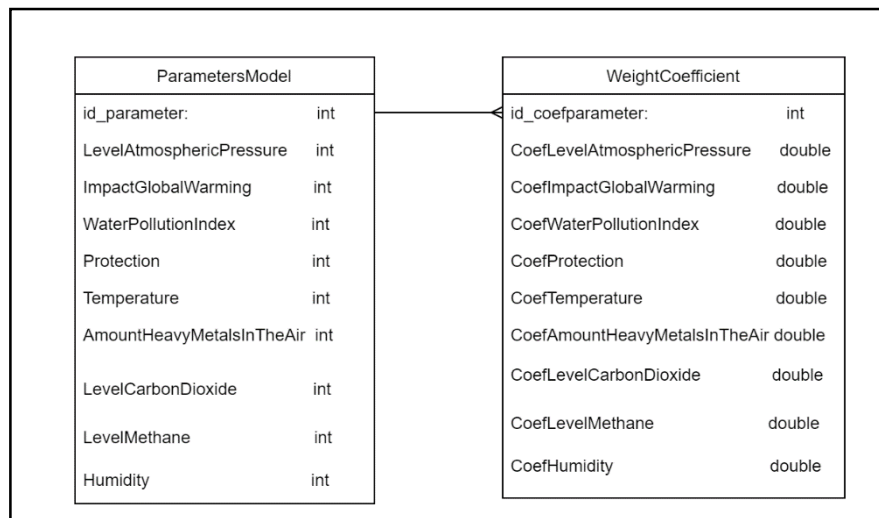


Рисунок 2.1 – Приклад моделі

ParametersModel це оцінки параметрів одного природнього об'єкту. WiegthCoefficient це вагові коефіцієнти відповідно до кожного параметра природнього об'єкту.

Також, ця модель передбачає те, що деякі параметри або вагові коефіцієнти можуть бути відсутні, оскільки, відсутність деякої інформації про об'єкт часто може бути при отриманні даних з різних пристроїв та джерел. Саме тому, в ході дослідження методів оцінки, також, необхідно враховувати цей аспект.

Також, до проблематики дослідження необхідно віднести той факт, що через відсутність деяких даних про один природній об'єкт, його оцінка може бути значно меншою і не відобразити реальний її стан.

Тобто, алгоритм розрахунку може правильно спрацювати, але отриманий результат хоч і буде коректним з точки зору оцінки на основі відомих параметрів

об'єкту, але у порівнянні з загальними оцінками інших об'єктів значно відрізняться. Наприклад, оцінка може бути значно меншою, через відсутність інформації про деякі дані, але на практиці об'єкти можуть мати приблизно однаковий рівень екології.

Отже, в ході дослідження необхідно порівняти відхилення результатів методів при різній наявності даних про параметри.

Деяка частина даних про екологію природнього об'єкту отримана з різних датчиків. Така концепція не обов'язково передбачає наявність шуканих даних про параметри та успішну передачу цих даних до серверної частини.

Приклад такої взаємодії зображено на діаграмі компонентів на рисунку 2.2.

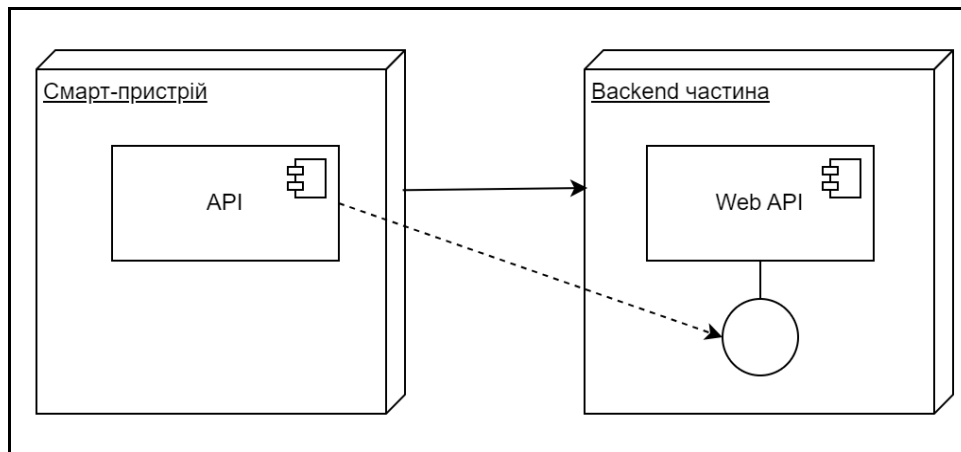


Рисунок 2.2 – Діаграма компонентів взаємодії

До методів оцінки моделі на основі параметрів з урахуванням вагових коефіцієнтів відносяться наступні методи.

Метод суми добутків.

Для реалізації методу необхідно деякі відомі параметри моделі подати у вигляді як $p_1 \dots p_n$, а їх вагові коефіцієнти у вигляді $k_1 \dots k_n$.

Тоді, шукана загальна оцінка моделі буде знаходитися за формулою 2.1:

$$O = p_1 * k_1 + \dots + p_n * k_n \quad (2.1)$$

де O — це обчислена загальна оцінка.

Такий підхід до оцінки моделі може бути інколи простим та швидким, але

містити недоліки. До таких недоліків можна віднести надання, через відсутність деякої частини даних, некоректного результату в порівнянні з іншими методами розрахунку моделі одного об'єкту.

Приклад моделі з можливими невідомими даними про об'єкт зображено на рисунку 2.3.

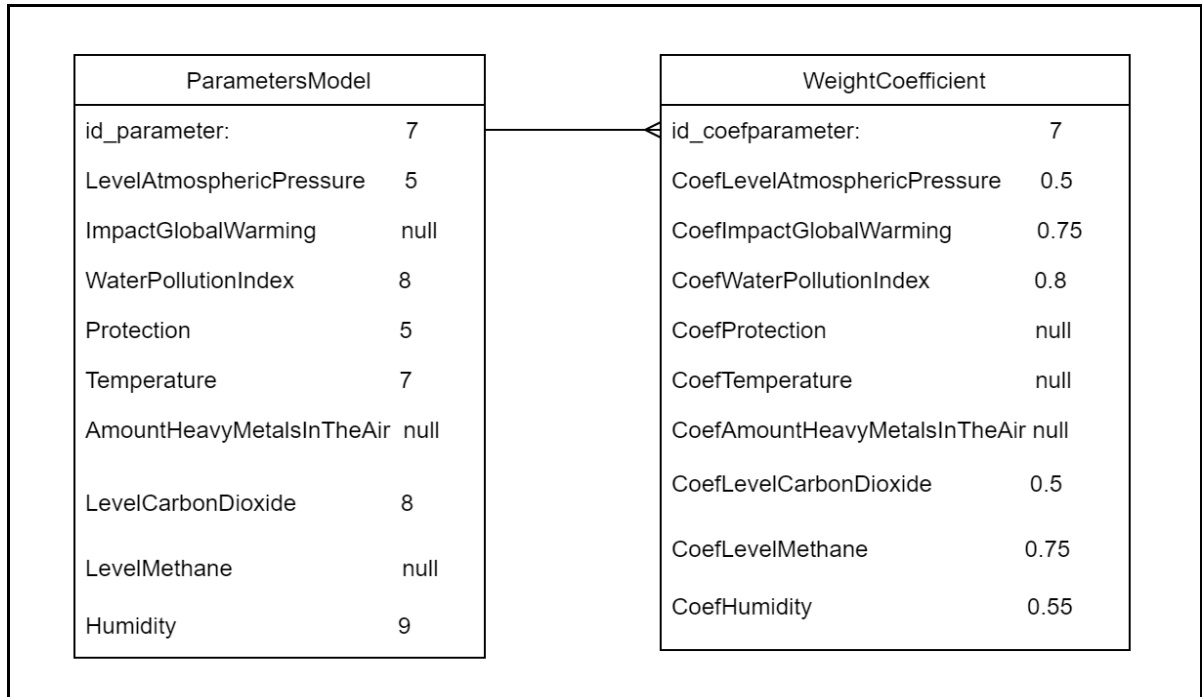


Рисунок 2.3 – Приклад моделі з деякими невідомими даними

На рисунку зображено модель при відсутності частини даних про параметри та вагові коефіцієнти об'єкту.

Метод комбінації.

Для виконання розрахунків завдяки цьому методу потрібно параметри моделі задати у вигляді $p_1 \dots p_n$, а вагові коефіцієнти $k_1 \dots k_n$, тоді, загальна оцінка може розраховуватись формулою 2.2:

$$O = (\sum_{i=1}^n (k_i)) / n * (\sum_{i=1}^n (p_i)) \quad (2.2)$$

де O — це обчислена загальна оцінка.

Реалізація цього методу демонструє добуток показника всіх вагових коефіцієнтів та суми оцінок параметрів моделі.

До переваг цього методу можна віднести потенційне зменшення коливань різниць між загальними оцінками моделі при різній відсутності даних про параметри.

Метод виключення всіх невідомих.

У ході цього методу виконується декілька етапів обчислень. Спочатку аналізуються дані моделі та відкидаються параметри та коефіцієнти значення яких невідоме.

Приклад моделі виключення невідомих зображено на рисунку 2.4.

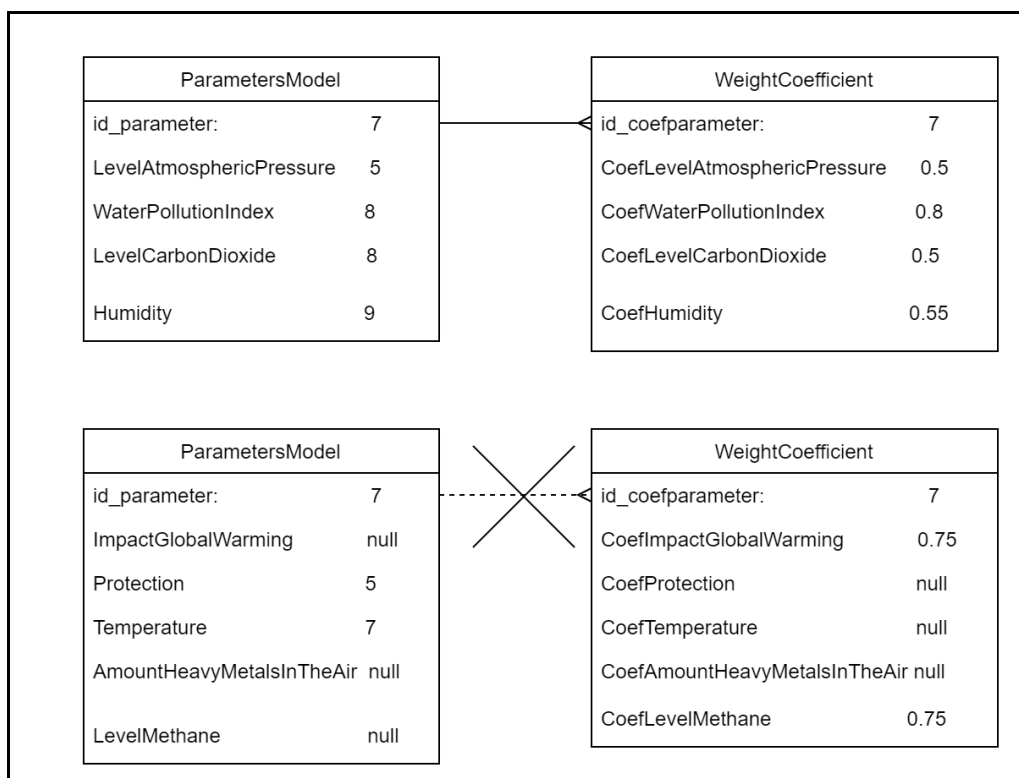


Рисунок 2.4 – Приклад виключення невідомих

На рисунку зображено виключення невідомих даних з моделі, таким чином що невідоме значення параметру, також, виключає з моделі його ваговий коефіцієнт і навпаки. Такий процес описує перший етап методу. Другим етапом буде застосування методу комбінації до оцінок параметрів та вагових коефіцієнтів, що залишилися у моделі.

Метод прогнозування невідомих коефіцієнтів.

Цей метод передбачає виключення тільки невідомих параметрів, а якщо у них є невідомими лише коефіцієнти, то вони залишаються у моделі. Першим

кроком буде надання прогнозованого значення невідомим ваговим коефіцієнтам. Цей процес відбувається за формулою 2.3:

$$E = (\sum_{i=1}^n (1 - k_i)) / n \quad (2.3)$$

де E — це прогнозований невідомий коефіцієнт.

Наступним кроком значення невідомих коефіцієнтів позначається, як знайдене E , та виконується розрахунок загальної оцінки моделі методом комбінації.

Метод врахування невідомих.

Цей метод схожий на метод виключення всіх невідомих, але на відміну від нього передбачає, також, вплив виключених даних на розрахунок.

Спочатку знаходяться прогнозуємі оцінки невідомих параметрів та коефіцієнти за допомогою використання першого кроку методу прогнозування. Наступним кроком розраховуються середні значення загальних оцінок, які були обчислені метод суми добутоків для відомих значень та прогнозованих.

2.2 Аналіз способів реалізації методів

Методи оцінки моделі на основі параметрів можна реалізувати на back-end різними способами. Така концепція передбачає деякі відмінності у результатах виконання методів. Наприклад, може відрізнитися час роботи алгоритму.

На back-end частині методи можна реалізувати такими способами:

- реалізація у кодї за допомогою сервісу на back-end частині. Така концепція є універсальною щодо можливості створення всіх методів;
- реалізація завдяки sql-запитам та використанню Express.js. Таким способом можна зробити прості методи, наприклад, метод добутоків.

Для реалізації різних способів потрібно створити чітку структуру компонента. Реалізація різних способів, важлива для проведення експериментів для порівняння методів за різними аспектами.

Потрібно зробити зрозумілу архітектуру компонента [7]. Можливі способи

реалізації у вигляді структури компонента системи зображено на рисунку 2.5.

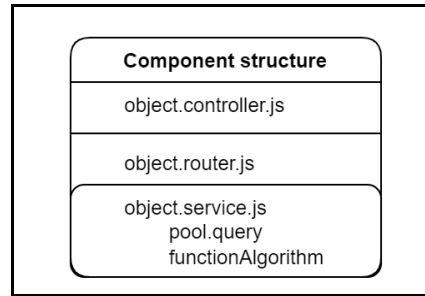


Рисунок 2.5 – Структура компонента

На рисунку зображено `pool.query` та `functionAlgorithm`. В `pool.query` можна реалізувати методи за допомогою `sql`. А в `functionAlgorithm` можна реалізувати методи за допомогою JavaScript та бібліотек.

2.3 Очікувані результати

В результаті дослідження потрібно реалізувати методи оцінки. Деякі з них можна реалізувати різними способами. Потрібно проаналізувати результати тестування методів на різних наборах даних, оцінити реалізованні методи на основі критеріїв та порівняння методи.

Методи дослідження моделі потрібно порівняти за критеріями:

- чутливість результату до відсутності деяких даних;
- простота реалізації методу;
- можливість реалізації різними способами;
- швидкість виконання алгоритму;
- стабільність результатів до різних наборів даних.

Отже, в результаті проведення дослідження очікується ретельний аналіз методів та способів оцінки моделі та їх порівняння, та надання рекомендацій стосовно кращих методів.

3 ПІДГОТОВКА ДО ПРОВЕДЕННЯ ДОСЛІДЖЕННЯ ТА РЕАЛІЗАЦІЯ

3.1 Підготовка технологій для дослідження

Після процесу опису методів та способів оцінки моделі на основі параметрів та вагових коефіцієнтів, перейдемо до наведення їх реалізації стосовно обраних технологій та back-end частини. В першу чергу, для реалізації методів потрібно використати деякі бібліотеки. А також, для реалізації методів оцінки різними способами потрібно зробити конфігурацію Express.js [8].

Для підготовки даних дослідження, потрібно виконати такі дії:

- обрати природні об'єкти для проведення дослідження;
- обрати відповідні параметри оцінок об'єктів;
- позначити їх ваговий внесок;
- налаштувати отримання даних для реалізації методів;
- виконати конфігурацію Express.js;
- налаштувати бібліотеки;
- спроектувати методи оцінки.

Наступним кроком перейдемо до вибору та порівняння технологій back-end частини цього дослідження.

3.2 Вибір технологій back-end частини

Під час проведення дослідження буде порівняна реалізація методів оцінки моделі на основі параметрів на back-end частині. Для проведення дослідження потрібно визначитись з технологіями створення back-end частини, на якій будуть реалізовуватись методи оцінки моделі.

Для цього скористаємося лінійною адаптивною згортою з ваговими коефіцієнтами.

Виберемо серед таких технологій розробки back-end частини.

Java Servlets.

Це технологія для створення динамічних застосунків на платформі Java. Вона включає достатню швидкість виконання і простоту розробки, але має

низький рівень підтримку бібліотек та низькі можливості у реалізації методів.

Java Spring.

Це потужний фреймворк для розробки Java-додатків [9]. Він включає достатню швидкість виконання та підтримку досить значної кількості бібліотек, але має значну складність розробки.

Node.js.

Це платформа для розробки back-end частини, яка забезпечує достатню швидкість виконання алгоритмів. Вона включає підтримку великої кількості бібліотек та можливість гнучкого використання різних методів [10].

ASP.NET.

Ця платформа включає підтримку великої кількості бібліотек та достатню швидкість виконання, але вона має не дуже велику актуальність.

Python Django.

Це фреймворк для створення back-end частини. Він включає підтримку багатьох бібліотек та простоту розробки, але має невеликі складності для реалізації всіх методів оцінки.

У якості критеріїв для порівняння технологій, було обрано деякі параметри.

Рівень підтримки бібліотек (для оцінки моделі).

Порядкова шкала від 1 до 10. Де 1 позначає відсутність необхідних бібліотек, а 10 повну підтримку бібліотек. Перший за важливістю критерій розробки. Він впливає на рівень складності розробки та швидкість. Для вибору треба орієнтуватися на більший рівень підтримки бібліотек.

Швидкість роботи.

Порядкова шкала від 1 до 10. Де 1 позначає повільніше виконання алгоритму, а 10 найшвидше. Третій за важливістю критерій, який прямо впливає на результат дослідження.

Можливість реалізації всіх алгоритмів.

Порядкова шкала від 1 до 10. Де 1 позначає неможливість реалізації алгоритмів, а 10 реалізацію всіх необхідних алгоритмів. Другий за важливістю критерій розробки. Він впливає на проведення повного дослідження всіх методів.

Рівень простоти розробки.

Порядкова шкала від 1 до 10. Де 1 позначає найскладнішу реалізацію алгоритмів, а 10 найпростішу реалізацію. Для вибору необхідно враховувати простішу розробку методів оцінки моделі.

Актуальність платформи.

Абсолютна шкала, що показує відсоток використання технології створення back-end частини. Для вибору необхідно орієнтуватися на актуальні платформи. Це впливає на підтримку бібліотек і актуальність проведення дослідження.

Наведемо векторний опис альтернатив за обраними критеріями у вигляді таблиці 3.1.

Таблиця 3.1 – Векторний опис альтернатив (виконана самостійно)

Технології створення back-end	Критерії				
	Рівень підтримки бібліотек	Швидкість роботи	Можливість реалізації всіх алгоритмів	Рівень простоти розробки	Актуальність платформи
Java Servlets	5	6	5	6	8.3%
Java Spring	8	9	7	7	11.1%
Node.js	10	8	10	8	17,8%
ASP.NET	8	7	7	7	5.5%
Python Django	9	8	8	9	25,7%

Використаємо принцип Парето [11]. Оскільки, Java Servlets та ASP.NET мають гірші характеристики у порівнянні з Java Spring, за принципом Парето, їх

можна виключити.

Внесемо зміни до таблиці 3.2.

Таблиця 3.2 – Змінений векторний опис альтернатив (виконана самостійно)

Технології створення back-end	Критерії				
	Рівень підтримки бібліотек	Швидкість роботи	Можливість реалізації всіх алгоритмів	Рівень простоти розробки	Актуальність платформи
Java Spring	8	9	7	7	11.1%
Node.js	10	8	10	8	17,8%
Python Django	9	8	8	9	25,7%

Отже, перейдемо до нормування критеріїв.

Тобто, визначимо найгірші та найкращі значення критеріїв.

Рівень підтримки бібліотек:

- найгірше – 8;
- найкраще – 10.

Швидкість роботи:

- найгірше – 8;
- найкраще – 9.

Можливість реалізації всіх алгоритмів:

- найгірше – 7;
- найкраще – 10.

Рівень простоти розробки:

- найгірше – 7;
- найкраще – 9.

Актуальність платформи:

- найгірше – 11,1;
- найкраще – 25,7.

Внесемо зміни до таблиці 3.3.

Таблиця 3.3 – Нормований векторний опис альтернатив (виконана самостійно)

Технології створення back-end	Критерії				
	Рівень підтримки бібліотек	Швидкість роботи	Можливість реалізації всіх алгоритмів	Рівень простоти розробки	Актуальність платформи
Java Spring	0	1	0	0	0
Node.js	1	0	1	0,5	0,46
Python Django	0,5	0	0,33	1	1

У цій задачі критерії мають різний пріоритет.

Використаємо лінійною адаптивною згорткою з ваговими коефіцієнтами. Скорегуємо пріоритет деяких критеріїв над іншими.

Визначимо коефіцієнти згорткової моделі. Критерій рівень підтримки бібліотек є найважливішим у цьому дослідженні. Його коефіцієнт:

$$\frac{5}{1 + 2 + 3 + 4 + 5} = \frac{5}{15} = \frac{1}{3}$$

Критерій можливості реалізації всіх алгоритмів другий за важливістю. Його коефіцієнт дорівнює:

$$\frac{4}{1 + 2 + 3 + 4 + 5} = \frac{4}{15}$$

Критерій швидкість роботи третій за важливістю. Його коефіцієнт буде

дорівнювати:

$$\frac{3}{1 + 2 + 3 + 4 + 5} = \frac{3}{15} = \frac{1}{5}$$

Критерій рівень простоти розробки четвертий за важливістю. Його коефіцієнт:

$$\frac{2}{1 + 2 + 3 + 4 + 5} = \frac{2}{15}$$

Критерій актуальність платформи ще трохи менш важливий. Його коефіцієнт дорівнює:

$$\frac{4}{1 + 2 + 3 + 4 + 5} = \frac{4}{15}$$

Внесемо коефіцієнти до таблиці 3.4.

Таблиця 3.4 – Опис альтернатив з коефіцієнтом (виконана самостійно)

Технології створення back-end	Критерії				
	Рівень підтримки бібліотек	Швидкість роботи	Можливість реалізації всіх алгоритмів	Рівень простоти розробки	Актуальність платформи
Java Spring	0	1	0	0	0
Node.js	1	0	1	0,5	0,46
Python Django	0,5	0	0,33	1	1
Коефіцієнт	$\frac{1}{3}$	$\frac{4}{15}$	0,2	$\frac{2}{15}$	$\frac{4}{15}$

Виконаємо розрахунки корисності завдяки коефіцієнтам.

$$k1 = \frac{1}{3} * 0 + \frac{1}{5} * 1 + \frac{4}{15} * 0 + \frac{2}{15} * 0 + \frac{1}{15} * 0 = 0,2$$

$$k2 = \frac{1}{3} * 1 + \frac{1}{5} * 0 + \frac{4}{15} * 1 + \frac{2}{15} * 0,5 + \frac{1}{15} * 0,46 = 0,7$$

$$k_3 = \frac{1}{3} * 0,5 + \frac{1}{5} * 0 + \frac{4}{15} * 0,33 + \frac{2}{15} * 1 + \frac{1}{15} * 1 = 0,45$$

Внесемо коефіцієнти до таблиці 3.5.

Таблиця 3.5 – Фінальний опис альтернатив (виконана самостійно)

Технології створення back-end	Критерії					Корисність
	Рівень підтримки бібліотек	Швидкість роботи	Можливість реалізації всіх алгоритмів	Рівень простоти розробки	Актуальність платформи	
Java Spring	0	1	0	0	0	0,2
Node.js	1	0	1	0,5	0,46	0,7
Python Django	0,5	0	0,33	1	1	0,45
Коефіцієнт	$\frac{1}{3}$	$\frac{4}{15}$	0,2	$\frac{2}{15}$	$\frac{4}{15}$	

Отже, за результатами вирішення задачі вибору, під час проведення дослідження, для реалізації back-end, краще використовувати технологію Node.js. Хоча, технологія має одну з найнижчих швидкостей роботи, однак має найвищі показники за важливими критеріями.

3.3 Структура компонентів дослідження

Після остаточного обрання технології back-end перейдемо до створення структури дослідження та back-end частини.

Потрібно описати на схемі архітектуру компонентів розглянутої системи. Її

зображено у вигляді діаграми розгортання на рисунку 3.1.

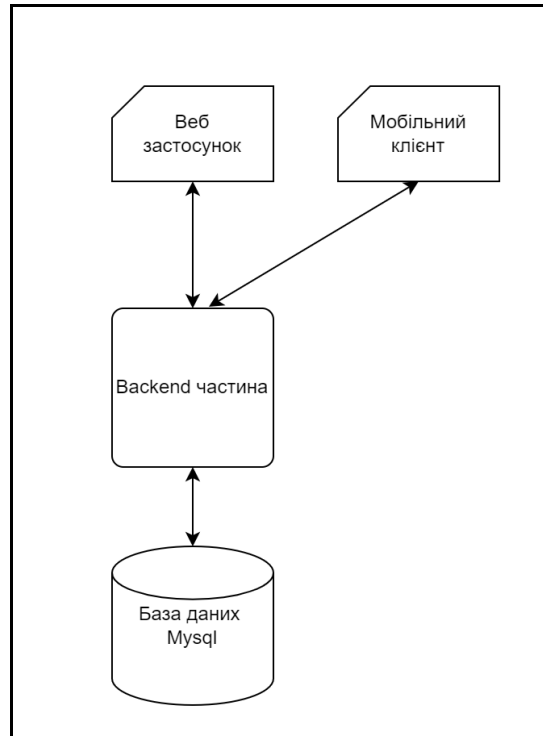


Рисунок 3.1 – Діаграма розгортання

На рисунку зображено взаємодію веб-застосунку і мобільного клієнту з back-end частиною та взаємодію бази даних з back-end частиною.

Далі необхідно описати та спроектувати схему за якою відбувається транспортування даних.

Алгоритм транспортування даних важливий для створення запитів між компонентами системи.

Алгоритм транспортування даних у розглянутій системі зображено на рисунку 3.2.

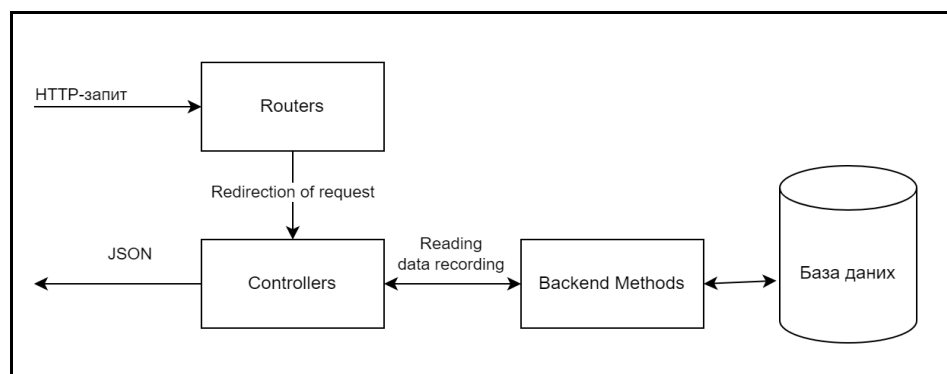


Рисунок 3.2 – Алгоритм транспортування даних

На рисунку відображено схематично принцип транспортування даних. А саме показана робота з JSON та запитами. Показана взаємодія між back-end частиною та Controllers і Routers.

Також, оскільки Node.js було визначено технологією back-end частини, необхідно описати схематично взаємодію між пакетами на back-end.

Для створення якісного коду потрібно зробити чітку діаграму пакетів [12]. Діаграму, яка показує взаємодію між пакетами серверної частини показано на рисунку 3.3.

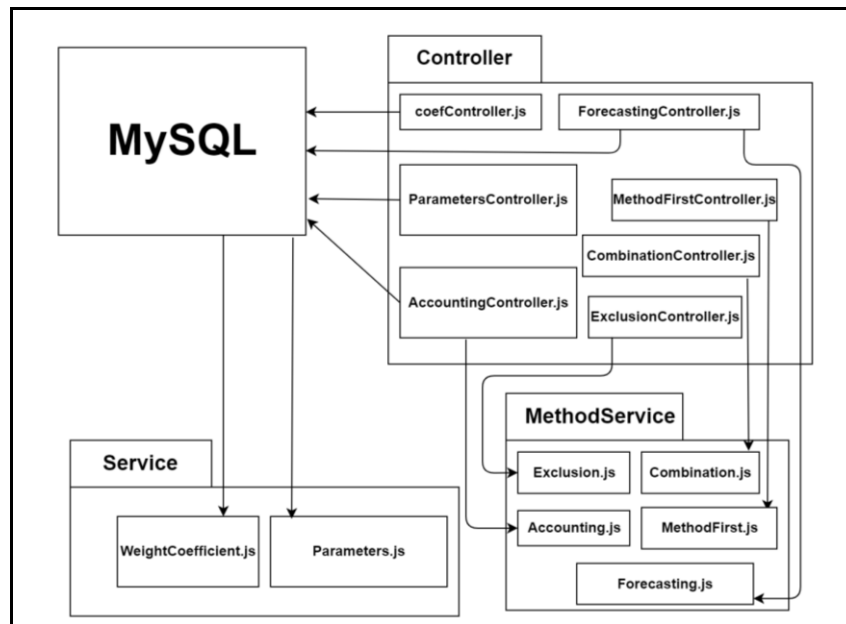


Рисунок 3.3 – Діаграма пакетів

Також, треба зробити прототипування схеми в якому вигляді потрібно подавати результати оцінки.

Прототип відображення кінцевих обчислень оцінок моделі зображено на рисунку 3.4.

На рисунку зображено результат обчислень. Прототип результату подано у вигляді картки у яких інформація про природній об'єкт подана списком параметрів та їх оцінок.

Спочатку вказано назву природнього об'єкту до якого буде застосована досліджуема модель. Потім продемонстровані параметри моделі та їх оцінки. Оцінка представляє собою число.

Natural object	
Name	Mississippi river
Level of atmospheric pressure	5
Impact of global warming	7
Popularity of natural objects	8
Protection of the natural object	7
Water pollution index	8
Temperature	7
The amount of heavy metals in the air	10
Carbon dioxide level	8
Methane level	5
Humidity	3
Estimation algorithm1	77
Estimation algorithm2	38
Estimation algorithm3	83

Рисунок 3.4 – Прототип відображення результату обчислення оцінок

А також, в кінці списку вказані результати загальної оцінки моделі на основі параметрів. Обчислені результати вказані для кожного методу оцінки окремо. Такий спосіб подання інформації надає гарне представлення результатів обчислення методів для подальшого їх аналізу, опису та критеріального порівняння.

Отже, було надано спосіб реалізації поставлених задач. Подано результат дослідження, завдяки чому можна реалізувати та дослідити методи оцінки моделі на основі параметрів на back-end частині.

3.4 Створення UI/UX та реалізація методів у системі

Після проведення проектування системи, була виконана реалізація

компонент системи. Почалося створення back-end частини та веб-застосунку і мобільного застосунку.

Також, було створено контролер, який оброблює результати запиту до back-end частини, який зображений на рисунку 3.5.

```
const {
  getEstimation
} = require("../getEstimation.service");

module.exports = {
  getEstimation: (req, res) => {
    const Estimation = req.params.Estimation;
    getEstimation(Estimation, (err, results) => {
      if(err){
        console.log(err);
        return;
      }
      if (!results){
        return res.json({
          success: 0,
          message: "Record not Found"
        });
      }
      return res.json({
        success: 0,
        data: results
      });
    });
  }
};
```

Рисунок 3.5 Контролер розрахунку оцінки моделі (виконаний самостійно)

Для серверної частини було використано Node.js та Express. Було створено підключення через налаштування конфігурації для з'єднання з базою даних, яке зображене на рисунку 3.6.

```
const {createPool} = require("mysql");

const pool = createPool({
  port:process.env.DB_PORT,
  host:process.env.DB_HOST,
  user:process.env.DB_USER,
  password:process.env.DB_PASS,
  database:process.env.MYSQL_DB,
  connectionLimit: 10
});
```

Рисунок 3.6 З'єднання з базою даних (виконаний самостійно)

Також, було створено деякі методи оцінки моделі на основі параметрів. Наприклад, код який відповідає за розрахунок методу виключення всіх невідомих зображено на рисунку 3.7.

```
const [paramdata, coefficdata] = await Promise.all([paramdataPromise, coefficdataPromise]);
const paramArr = Object.values(paramdata);
const coefficArr = Object.values(coefficdata);
const aParamArr = [];

const aCoefficArr = [];

for (let i = 0; i < paramArr.length; i++) {
  if (paramArr[i] !== 0 && coefficArr[i] !== 0) { aParamArr.push(paramArr[i]);
    aCoefficArr.push(coefficArr[i]);
  }
}

let resmethod = 0;

for (let i = 0; i < aParamArr.length; i++) { resmethod += aParamArray[i] * aCoefficArray[i]; }
res.json({
  success: 1,
  data: [
    { resmethod: resmethod }
  ]
});
```

Рисунок 3.7 – Код методу виключення всіх невідомих (виконаний самостійно)

Деякі методи було реалізовано різними способами. Метод суми добутків, який реалізований через sql-запит, зображений на рисунку 3.8.

```
const pool = require("../config/database");

module.exports = {
  getMethod: callBack => {
    pool.query(
      `select Distinct parameter.id_parameter,
      ((parameter.levelAtmosphericPressure * excoefficient.coeflevelAtmosphericPressure) +
      (parameter.impactGlobalWarming * excoefficient.CoefimpactGlobalWarming) +
      (parameter.waterPollutionIndex * excoefficient.coefwaterPollutionIndex) +
      (parameter.protection * excoefficient.coefprotection) +
      (parameter.temperature * excoefficient.coeftemperature) +
      (parameter.amountHeavyMetalsInTheAir * excoefficient.coefamountHeavyMetalsInTheAir) +
      (parameter.levelCarbonDioxide * excoefficient.coeflevelCarbonDioxide) +
      (parameter.levelMethane * excoefficient.coeflevelMethane) +
      (parameter.humidity * excoefficient.coefhumidity)) AS MethodS from parameter, excoefficient
      WHERE parameter.id_parameter = excoefficient.id_parameter`,
      [],
      (error, results, fields) => {
        if(error) {
          return callBack(error);
        }
        return callBack(null, results);
      }
    );
  },
};
```

Рисунок 3.8 Метод суми добутків через sql-запит (виконаний самостійно)

Веб-застосунок був реалізований за допомогою JS компонентів. Взаємодія з back-end частиною виконується через XMLHttpRequest. Це API використовується для скриптових мов браузерів.

Код звернення до back-end частини для завантаження даних, зображений на рисунку 3.9.

```
var estimationArr = {};  
  
window.onload = function() {  
  
    requestURL = 'http://localhost:3030/api/estimation/' + localStorage.getItem('estimation');  
    let request = new XMLHttpRequest();  
    request.open('GET', requestURL, true);  
  
    request.responseType = 'json';  
    request.send();  
    request.onload = function() {  
        var estimationT = request.response;  
        estimationArr = JSON.parse(JSON.stringify(estimationT))  
    }  
}
```

Рисунок 3.9 Звернення до серверу за інформацією (виконаний самостійно)

При вході до веб-клієнту, потрібно увійти до акаунту, шляхом введення логіну та паролю. Авторизація у веб-застосунку зображена на рисунку 3.10.

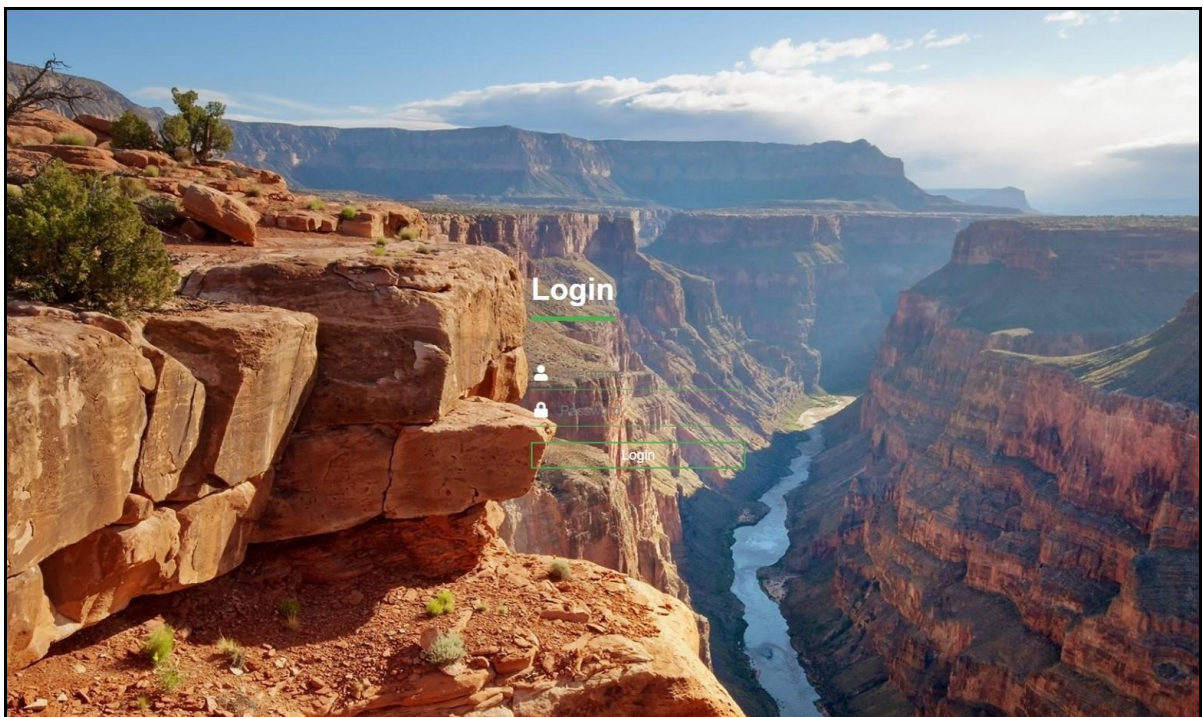


Рисунок 3.10 – Сторінка авторизації

Сторінка перегляду основної інформації про головні параметри моделі зображена на рисунку 3.11.

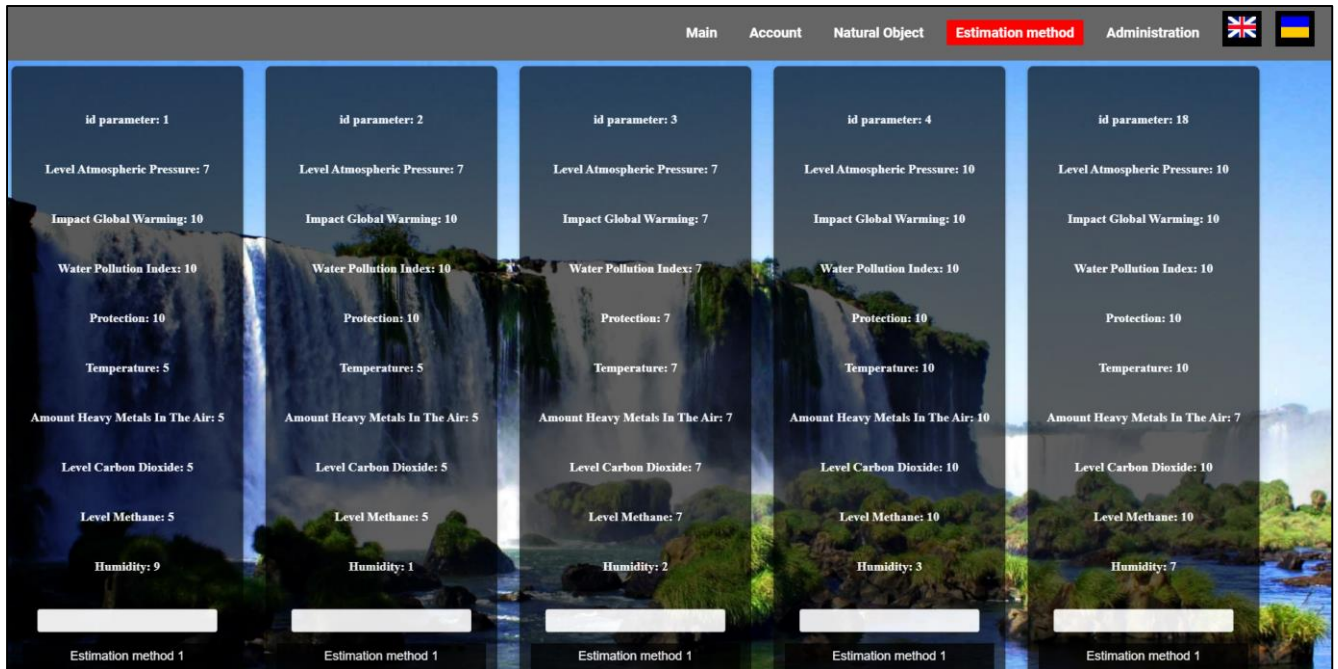


Рисунок 3.11 – Сторінка перегляду інформації

Сторінка зі входом у мобільний застосунок містить можливість вводу пароля та зображена на рисунку 3.12

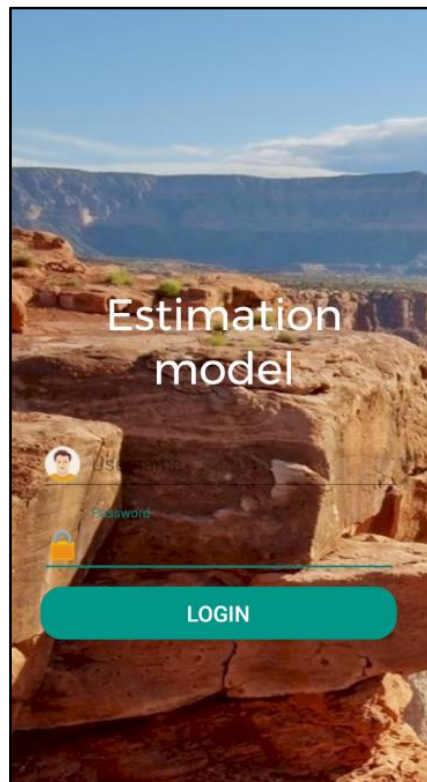


Рисунок 3.12 – Сторінка входу у мобільний застосунок

Сторінка мобільного застосунку з оцінками природнього об'єкту зображена на рисунку 3.13.

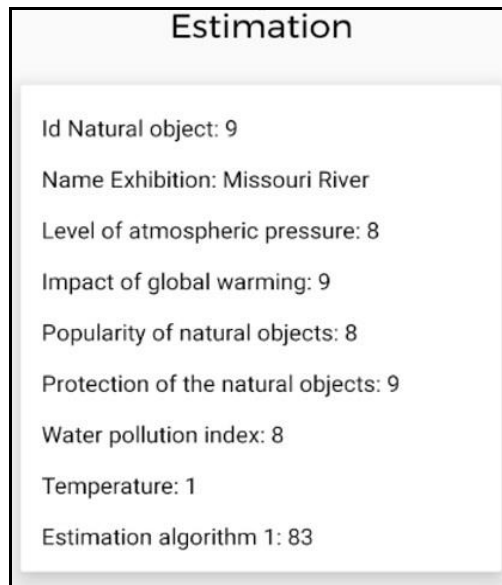


Рисунок 3.13 – Сторінка оцінок природнього об'єкту

Сторінка розрахунку загальної оцінки на основі параметрів відображає дані у вигляді карток з інформацією про кожен природній об'єкт. Картка надає інформацію у вигляді списку. Кожна картка містить інформацію про оцінки параметрів природнього об'єкту та загальну розраховану оцінку на основі параметрів.

Отже, в результаті роботи реалізовані методи оцінки моделі на основі параметрів. Також в ході роботи були створені сторінки програмної системи у веб-застосунку та мобільному застосунку для відображення результатів розрахунків методів оцінки моделі. Деякі методи оцінки були виконані різними способами на back-end частині.

4 ПОРІВНЯЛЬНИЙ АНАЛІЗ

4.1 Порівняльний аналіз методів

В результаті були реалізовані методи оцінки моделі. У ході дослідження на back-end частині були проведені експерименти над реалізованими методами на різних наборах даних. В результаті експериментів були надані оцінки методам за критеріями які додані до таблиці 4.1 та таблиці 4.2. Отже, перейдемо до порівняльного аналізу результатів дослідження методів.

Для цього скористаємося лінійною адаптивною згорткою.

Було протестовано наступні методи у дослідженні.

Метод суми добутоків.

Простий метод, який можна реалізувати різними способами. До його переваг відноситься його простота, а до недоліків чутливість результату до втрати даних.

Метод комбінації.

Трохи складніший метод на відміну від попереднього. Дозволяє зменшити вплив втрати даних та розрахунок результату. До недоліків можна віднести складність створення різними способами.

Метод виключення всіх невідомих.

Оскільки метод використовує декілька етапів обчислень це робить результат більш стійким до втрати деяких даних про коефіцієнти та параметри, але збільшує складність та потребує додаткового використання масивів на back-end частині.

Метод прогнозування невідомих коефіцієнтів.

Цей метод також більш складний через виконання у декілька етапів, але його перевагою є стабільність результатів до різних наборів даних.

Метод врахування невідомих.

Перевагою цього методу є знижена чутливість результату до відсутності даних та стабільність результатів, а до недоліки складність та швидкість виконання.

У якості критеріїв для порівняння методів, було обрано деякі параметри.

Чутливість результату до відсутності деяких даних.

Порядкова шкала від 1 до 10. Де 1 позначає значні коливання результату через втрати деяких даних, а 10 повну стійкість. Перший за важливістю критерій розробки.

Простота реалізації методу.

Порядкова шкала від 1 до 10. Де 1 позначає складну реалізацію алгоритму, а 10 найпростішу. Впливає на швидкість та на реалізації різними способами.

Можливість реалізації різними способами.

Порядкова шкала від 1 до 2. Де 1 позначає неможливість реалізації методів різними способами, а 2 можливість розробки різними способами.

Швидкість виконання алгоритму.

Порядкова шкала від 1 до 10. Де 1 позначає найповільнішу швидкість, а 10 найшвидше виконання.

Стабільність результатів до різних наборів даних.

Порядкова шкала від 1 до 10. Де 10 позначає, те що розрахована оцінка знаходиться в заданому діапазоні при різних наборах даних, а 1, означає те що може сильно коливатися.

Додамо виміри швидкості до таблиці 4.1.

Таблиця 4.1 – Швидкість виконання алгоритму (виконана самостійно)

Методи оцінки моделі на основі параметрів	Критерії				
	Метод суми добутоків	Метод комбінації	Метод виключення всіх невідомих	Метод прогнозування невідомих коефіцієнтів	Метод врахування невідомих
Швидкість виконання, ms	5	10	29	46	59

Кінець таблиці 4.1

Методи оцінки моделі на основі параметрів	Критерії				
	Метод суми добутоків	Метод комбінації	Метод виключення всіх невідомих	Метод прогнозування невідомих коефіцієнтів	Метод врахування невідомих
Швидкість виконання алгоритму, за порядковою шкалою	10	7	5	3	3

Наведемо векторний опис альтернатив за обраними критеріями у вигляді таблиці 4.2.

Таблиця 4.2 – Векторний опис (виконана самостійно)

Методи оцінки моделі на основі параметрів	Критерії				
	Чутливість до відсутності деяких даних	Просто та реалізація методу	Можливість реалізації різними способами	Швидкість виконання алгоритму	Стабільність результатів до різних наборів даних
Метод суми добутоків	1	10	2	10	5
Метод комбінації	5	8	1	7	7
Метод виключення	7	5	1	5	9

Кінець таблиці 4.2

Методи оцінки моделі на основі параметрів	Критерії				
	Чутливість до відсутності деяких даних	Простота реалізації методу	Можливість реалізації різними способами	Швидкість виконання алгоритму	Стабільність результатів до різних наборів даних
всіх невідомих					
Метод прогнозування невідомих коефіцієнтів	9	2	1	3	7
Метод врахування невідомих	8	2	1	3	8

Використаємо принцип Парето. Оскільки, метод врахування невідомих має гірші характеристики у порівнянні з методом виключення всіх невідомих, за принципом Парето, його можна виключити.

Внесемо зміни до таблиці 4.3.

Таблиця 4.3 – Змінений векторний опис (виконана самостійно)

Методи оцінки моделі на основі параметрів	Критерії				
	Чутливість до відсутності деяких даних	Простота реалізації методу	Можливість реалізації різними способами	Швидкість виконання алгоритму	Стабільність результатів до різних наборів даних
Метод суми	1	10	2	10	5

Кінець таблиці 4.3

Методи оцінки моделі на основі параметрів	Критерії				
	Чутливість до відсутності деяких даних	Простота реалізації методу	Можливість реалізації різними способами	Швидкість виконання алгоритму	Стабільність результатів до різних наборів даних
добутків					
Метод комбінації	5	8	1	7	7
Метод виключення всіх невідомих	7	5	1	5	9
Метод прогнозування невідомих коефіцієнтів	9	2	1	3	7

Наступним кроком перейдемо до нормування векторного опису альтернатив і обчислення необхідних розрахунків кожного методу. Методи розрахуємо відповідно критеріям.

Чутливість до відсутності деяких даних:

– метод суми добутоків: $\frac{1-1}{9-1} = \frac{0}{8} = 0$;

– метод комбінації: $\frac{5-1}{9-1} = \frac{4}{8} = 0,5$;

– метод виключення всіх невідомих: $\frac{7-1}{9-1} = \frac{6}{8} = 0,75$;

– метод прогнозування невідомих коефіцієнтів: $\frac{9-1}{9-1} = \frac{8}{8} = 1$.

Простота реалізації методу:

- метод суми добутків: $\frac{10-2}{10-2} = \frac{8}{8} = 1$;
- метод комбінації: $\frac{8-2}{10-2} = \frac{6}{8} = 0,75$;
- метод виключення всіх невідомих: $\frac{5-2}{10-2} = \frac{3}{8} = 0,375$;
- метод прогнозування невідомих коефіцієнтів: $\frac{2-2}{10-2} = \frac{0}{8} = 0$.

Можливість реалізації різними способами:

- метод суми добутків: $\frac{2-1}{2-1} = \frac{1}{1} = 1$;
- метод комбінації: $\frac{1-1}{2-1} = \frac{0}{1} = 0$;
- метод виключення всіх невідомих: $\frac{1-1}{2-1} = \frac{0}{1} = 0$;
- метод прогнозування невідомих коефіцієнтів: $\frac{1-1}{2-1} = \frac{0}{1} = 0$.

Швидкість виконання алгоритму:

- метод суми добутків: $\frac{10-3}{10-3} = \frac{1}{1} = 1$;
- метод комбінації: $\frac{7-3}{10-3} = \frac{4}{7} = 0,571$;
- метод виключення всіх невідомих: $\frac{5-3}{10-3} = \frac{2}{7} = 0,285$;
- метод прогнозування невідомих коефіцієнтів: $\frac{3-3}{10-3} = \frac{0}{7} = 0$.

Стабільність результатів до різних наборів даних:

- метод суми добутків: $\frac{5-5}{9-5} = \frac{0}{4} = 0$;
- метод комбінації: $\frac{7-5}{9-5} = \frac{2}{4} = 0,5$;
- метод виключення всіх невідомих: $\frac{9-5}{9-5} = \frac{4}{4} = 1$;
- метод прогнозування невідомих коефіцієнтів: $\frac{7-5}{9-5} = \frac{2}{4} = 0,5$.

Внесемо зміни до таблиці 4.4.

Вибір загорткової моделі. У записаному нами порівняльному аналізі критерії мають різну важливість. Тому для вирішення задачі вибору, використаємо лінійну адаптивну згортку з ваговими коефіцієнтами.

Таблиця 4.4 – Нормований векторний опис альтернатив (виконана самостійно)

Технології створення back-end	Критерії				
	Чутливість до відсутності деяких даних	Простота реалізації методу	Можливість реалізації різними способами	Швидкість виконання алгоритму	Стабільність результатів до різних наборів даних
Метод суми добутоків	0	1	1	1	0
Метод комбінації	0,5	0,75	0	0,571	0,5
Метод виключення всіх невідомих	0,75	0,375	0	0,285	1
Метод прогнозування невідомих коефіцієнтів	1	0	0	0	0,5

Порядок критеріїв, за важливістю:

- чутливість до відсутності деяких даних;
- стабільність результатів до різних наборів даних;
- швидкість виконання алгоритму;
- можливість реалізації різними способами;
- простота реалізації методу.

Визначення коефіцієнтів:

- чутливість до відсутності деяких даних:

$$\frac{5}{1 + 2 + 3 + 4 + 5} = \frac{5}{15} = \frac{1}{3}$$

- стабільність результатів до різних наборів даних:

$$\frac{4}{1 + 2 + 3 + 4 + 5} = \frac{4}{15}$$

- швидкість виконання алгоритму:

$$\frac{3}{1 + 2 + 3 + 4 + 5} = \frac{3}{15} = \frac{1}{5}$$

- можливість реалізації різними способами:

$$\frac{2}{1 + 2 + 3 + 4 + 5} = \frac{2}{15}$$

- простота реалізації методу:

$$\frac{1}{1 + 2 + 3 + 4 + 5} = \frac{1}{15}$$

Розрахунки корисності:

- метод суми добутків:

$$\frac{1}{3} * 0 + \frac{1}{15} * 1 + \frac{2}{15} * 1 + \frac{1}{5} * 1 + \frac{4}{15} * 0 = 0,4$$

- метод комбінації:

$$\frac{1}{3} * 0,5 + \frac{1}{15} * 0,75 + \frac{2}{15} * 0 + \frac{1}{5} * 0,571 + \frac{4}{15} * 0,5 = 0,45$$

- метод виключення всіх невідомих:

$$\frac{1}{3} * 0,75 + \frac{1}{15} * 0,375 + \frac{2}{15} * 0 + \frac{1}{5} * 0,285 + \frac{4}{15} * 1 = 0,59$$

- метод прогнозування невідомих:

$$\frac{1}{3} * 1 + \frac{1}{15} * 0 + \frac{2}{15} * 0 + \frac{1}{5} * 0 + \frac{4}{15} * 0,5 = 0,46$$

Отже, за результатами проведених експериментів та порівняльного аналізу результатів, визначено, що для реалізації методів оцінки моделі на основі параметрів на back-end частині, в загальному випадку краще використовувати метод виключення всіх невідомих. Хоча, метод має одну з найнижчих швидкостей виконання та складну реалізацію, однак має високі показники за найважливішими

критеріями.

Але, наприклад, коли не треба враховувати вплив невідомих даних та стабільність результату не потрібна, а потрібна швидкість виконання методу та простота реалізації, то в такому разі краще, ґрунтуючись на результатах дослідження, використати метод суми добутків.

ВИСНОВКИ

В результаті дослідження методи оцінки моделі на основі параметрів, були протестовані на різних наборах даних та визначені сильні сторони та обмеження кожного методу. За результатами дослідження був проведений порівняльний аналіз та зроблені обґрунтовані висновки.

До сильних сторін методу комбінації, методу виключення всіх невідомих, методу прогнозування невідомих відноситься стабільність до відсутності деяких даних та стабільність результатів до різних даних. До сильних сторін методу суми добутоків відноситься швидкість виконання, можливість створення алгоритму різними способами та простота виконання методу, а до слабких сторін значна чутливість до втрати даних параметрів та вагових коефіцієнтів та відносно малий показник стабільності.

Отримані результати відповідають задачам дослідження. В результаті виконання цього дослідження було виконано задачі, які були сформовані у постановці задачі, а саме:

- створено та проаналізовано завдання для дослідження;
- надано основну інформацію стосовно розглядаємої моделі;
- опрацьовано та розібрано інформацію для дослідження моделі та галузі її застосування – екології природніх об'єктів;
- надано основну інформацію стосовно методів оцінки моделі;
- надано основну інформацію стосовно способів застосування;
- обрано технології back-end для проведення дослідження;
- надано інформацію про технології проведення дослідження;
- реалізовані методи оцінки моделі на back-end частині;
- створений деякий функціонал на back-end частині та веб-застосунку;
- проведений порівняльний аналіз методів та сформовані результати тестування методів, які були оцінені критеріально;
- сформовано обґрунтований висновок результату проведеного дослідження.

Також, в результаті виконання роботи було, для більшої наглядності, наведено графічний матеріал схем компонентів та діаграм. Був проведений аналіз предметної галузі. Були наведені, проаналізовані та розібрані методи та різні способи розрахунку загальної оцінки моделі на основі параметрів за допомогою вагових коефіцієнтів.

Було проведено дослідження та реалізований необхідний функціонал. В якості платформи для серверної частини, під час критеріального аналізу, було обрано Node.js. Дослідження було успішно проведено з використанням обраних технологій.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Кожешкурт І.В., Валенда Н.А. Проблеми розпізнавання іменованих сутностей комп'ютерними системами // Abstracts of the 9th International scientific and practical conference. Sofia, Bulgaria С. 444-447.
2. Шуляк С.М., Валенда Н.А. Задача аналізу тональності тексту // Abstracts of the 9th International scientific and practical conference. Sofia, Bulgaria С. 951-956.
3. Валенда Н. А. Метод автоматичної обробки паралельних корпусів // Міжвузівський збірник "Наукові нотатки". Луцьк, 2019 С. 19-23.
4. Павленко Я. В., Валенда Н.А. Методи оцінки моделі на основі параметрів // Modernization of science and its influence on global processes, 2024. С. 102-104 URL: <https://previous.scientia.report/index.php/archive/issue/view/03.05.2024> (дата звернення 01.06.2024)
5. Mayer G. E. T., Awesomeness J. JavaScript: JavaScript Awesomeness Book. CreateSpace Independent Publishing Platform, 2017. С. 68.
6. Документація MySQL URL: <https://dev.mysql.com> (дата звернення 10.02.2024)
7. Martin R. Clean Coder: A Code of Conduct for Professional Programmers. Pearson Education, Limited.
8. Документація Express.js URL: <https://expressjs.com> (дата звернення 10.02.2024)
9. Acetozi J. The Spring Framework. Pro Java Clustering and Scalability. Berkeley, CA, 2017. С. 47–53. URL: https://doi.org/10.1007/978-1-4842-2985-9_8 (дата звернення 10.02.2024)
10. Документація Node.js URL: <https://nodejs.org> (дата звернення 1.02.2024)
11. Miller D. Pareto principle. Routledge Encyclopedia of Philosophy. London. URL: <https://doi.org/10.4324/9780415249126-s097-1> (дата звернення 10.02.2024)
12. Fowler M. Refactoring: Improving the Design of Existing Code. Pearson Education, Limited, 2018.