

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: 123 «Комп'ютерна інженерія»

Освітньо-професійна програма: «Комп'ютерна графіка і Web-дизайн»

Група: 4КГ-08

# Дипломний проект

здобувача освіти денної форми навчання

КГ.08.03.000.ДП

**ВЕЛЬЧЕНКА**  
**КИРИЛА ІГОРОВИЧА**

м. Одеса  
2025 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: 123 «Комп'ютерна інженерія»

Освітньо-професійна програма: «Комп'ютерна графіка і Web-дизайн»


Група: 4КГ-08

**ПОЯСНЮВАЛЬНА ЗАПИСКА**

до дипломного проекту на тему:

**Розробка експертного веб-застосунку**  
**розрахунку екологічних слідів**

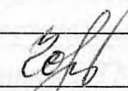
Проектний матеріал складається з пояснювальної записки на 77 сторінках та графічного (презентаційного) матеріалу на 15 аркушах (слайдах)

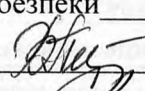
Дипломник  ( Вельченко К.І. )

Керівник  ( Жадан А.С. )

**Консультанти:**


з економічного розділу  ( Канський М.Ю. )

з розділу охорони праці та техніки безпеки  ( Чорновол Н.І. )

з нормоконтролю  ( Петрашова В.І. )

старший консультант  ( Кривченко Ю.В. )

**До захисту допущений**

Голова циклової комісії  ( Кривченко Ю.В. )

Завідувач відділення  ( Краснокутська К.Г. )

Захист « 21 » червня 2025 р.

Протокол ЕК № 2

Оцінка ЕК 4 (добре) / 80%

Секретар ЕК 

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Відділення комп'ютерних систем Комісія КТ та ПІ  
Спеціальність 123 «Комп'ютерна інженерія»  
Освітньо-професійна програма «Комп'ютерна графіка і web-дизайн»

ЗАТВЕРДЖУЮ:  
Заст. дир. з НВР Беркань І.В.  
“ 19 ” 08 2025 р.

**ЗАВДАННЯ**

на дипломний проект

Здобувачеві освіти Вельченку Кирилу Ігоровичу

(прізвище, ім'я, по батькові)

1. Тема проекту Розробка експертного веб-застосунку розрахунку екологічних слідів.

затверджена наказом по коледжу від “ 14 ” листопада 2024р. № 246

2. Термін здачі закінченого проекту \_\_\_\_\_

3. Вихідні данні до проекту \_\_\_\_\_

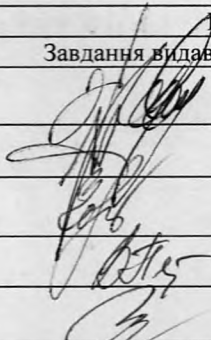
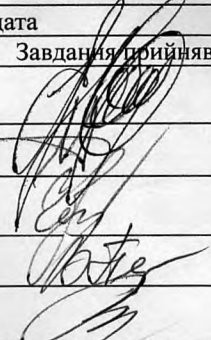
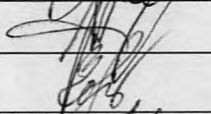
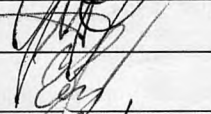
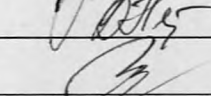
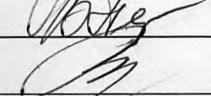
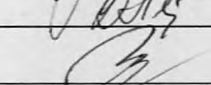
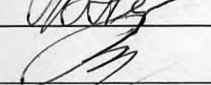
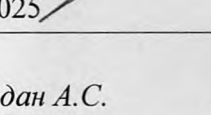

1. Передбачити структуру веб-застосунку відповідно до поставленої мети;
2. Застосовувати сучасні методи веб-розробки з використанням HTML, CSS і JavaScript;
3. Реалізувати обчислення екологічного сліду на основі встановлених коефіцієнтів;
4. Забезпечити адаптивність інтерфейсу для різних типів пристроїв;
5. Застосувати принципи UX/UI-дизайну для забезпечення зручності користувача.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які необхідно розробити)

1. Аналіз проблеми зміни клімату та екологічного;
2. Огляд існуючих веб-рішень для оцінки викидів CO<sub>2</sub>;
3. Вибір технологій реалізації веб-застосунку;
4. Архітектура та файлова структура калькулятора;
5. Розробка інтерфейсу користувача;
6. Реалізація функціоналу та збереження даних;
7. Алгоритми підрахунку викидів та візуалізація результатів;
8. Тестування, адаптація та висновки щодо ефективності рішення;
9. Економіка;
10. Охорона праці.

5. Перелік графічного (презентаційного) матеріалу (з точним зазначенням обов'язкових креслень, кількості слайдів)  
Титул; Вступ: Оцінка особистих викидів CO<sub>2</sub>; Використані технології; Архітектура застосунку; Файлова структура проекту; Ключовий функціонал; Алгоритм розрахунку CO<sub>2</sub>; UI/UX Дизайн Застосунку; Результат розрахунку; Кнопка «Редагувати відповіді»; Інформаційні Сторінки Застосунку; Тестування Застосунку; Оптимізація Застосунку; Публікація Застосунку; Висновки.

6. Консультанти по проекту, із зазначенням розділів проекту, що їх стосується

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Основний розділ	Жадан А.С.		
Економічний розділ	Канський М.Ю.		
Розділ охорони праці	Чорновол Н.І.		
Нормоконтроль	Петрашова В.І.		
Старший консультант	Кривченко Ю.В.		

7. Дата видачі завдання \_\_\_\_\_

12.05.2025

Керівник

Жадан А.С.

Завдання прийняв до виконання

Вельченко К.І.

(підпис)

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/р	Назва етапів дипломного проекту	Термін виконання етапів дипломного проекту (роботи)	Відмітка про виконання
1	Обґрунтування актуальності теми (вступ)	15.05.2025	Закрито
2	Аналіз екологічних проблем і предметної області	16.05.2025	Виконано
3	Огляд існуючих калькуляторів екологічного сліду	17.05.2025	Виконано
4	Вибір технологій реалізації	18.05.2025	Виконано
5	Розробка архітектури та структури проекту	22.05.2025	Виконано
6	Реалізація інтерфейсу та введення даних	26.05.2025	Виконано
7	Програмування функціоналу та логіки переходів	07.06.2025	Виконано
8	Створення алгоритму обчислення сліду	08.06.2025	Виконано
9	Візуалізація результатів через Chart.js	10.06.2025	Виконано
10	Додавання інформаційних блоків	11.06.2025	Виконано
11	Тестування адаптивності та функціональності	13.06.2025	Виконано
12	Оформлення пояснювальної записки	13.06.2025	Виконано
13	Економічний розрахунок	14.06.2025	Виконано
14	Розділ з охорони праці та безпеки	16.06.2025	Виконано

Дипломник

(підпис)

Керівник

(підпис)



# ЗМІСТ

Вступ.....	7
1 Основний розділ.....	8
1.1 Теоретичні основи розрахунку екологічного сліду .....	8
1.1.1 Аналіз проблеми та постановка задачі .....	8
1.1.2 Огляд існуючих рішень та підходів .....	12
1.2 Вибір та обґрунтування технологій реалізації.....	16
1.2.1 Вибір технологій для реалізації.....	16
1.2.2 Архітектура веб-застосунку.....	23
1.2.3 Файлова структура проєкту .....	26
1.2.4 Потенціал масштабування та подальшого розвитку веб-застосунку.....	30
1.3 Розробка та реалізація веб-застосунку .....	36
1.3.1 Інтерфейс користувача (UI/UX структура).....	36
1.3.2 Розробка функціоналу .....	41
1.3.3 Візуалізація результатів.....	46
1.3.4 Тестування, адаптація, перевірка .....	49
2 Економічний розділ .....	52
2.1 Резюме .....	52
2.2 Визначення трудомісткості розробки ПЗ.....	52
2.3 Розрахунок ціни програмного продукту .....	55
3 Розділ охорони праці та техніки безпеки.....	57
3.1 Основні положення.....	57
3.2 Аналіз умов праці й забезпечення безпеки при виконання основних видів робіт на об'єкті дипломного проєктування.....	57
3.3.1 Гігієнічні вимоги до виробничого середовища .....	57
3.3.2 Вимоги до приміщення експлуатації ПК.....	57
3.3.3 Гігієнічні вимоги до параметрів повітря приміщень із ПК.....	58
3.3.4 Виробниче освітлення .....	59
3.3.5 Організація робочих місць із ПК.....	59
3.3.6 Вимоги до режимів праці та відпочинку при роботі з ПК.....	60

					<i>КГ 08. 03 000. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		5

3.4 Пожежна безпека .....	60
Висновки.....	62
Перелік використаних інформаційних джерел .....	63
Додаток А. Фрагмент JavaScript коду розрахунків екологічного сліду .....	64
Додаток Б. Слайди мультимедійної презентації .....	69

					<i>КГ 08. 03 000. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		6

## ВСТУП

Екологічні проблеми вже давно посідають важливе місце серед глобальних викликів сучасності. Зростання викидів парникових газів, зміни клімату та надмірне споживання природних ресурсів змушують суспільство шукати нові підходи до свідомого ставлення до довкілля. Одним із важливих інструментів формування екологічної відповідальності є усвідомлення власного впливу на природу, зокрема – через поняття екологічного (вуглецевого) сліду. Метою даної дипломної роботи є створення інтерактивного веб-застосунку, який допоможе користувачеві оцінити свій екологічний слід на основі повсякденних дій: споживання електроенергії, використання транспорту, харчування тощо.

Для реалізації поставленої мети було сформульовано низку завдань: створити структуру веб-застосунку, розробити інтерфейс для введення даних, реалізувати логіку обчислення екологічного сліду за допомогою мови JavaScript, організувати виведення результатів у зручному форматі (у вигляді тексту або графіка), а також забезпечити базову стилізацію та адаптивність інтерфейсу.

У процесі роботи використовуються методи структурного проєктування, аналізу відкритих джерел даних з коефіцієнтами впливу, а також сучасні підходи до UX/UI-дизайну для зручності користувача. Розробка ведеться за допомогою HTML, CSS та JavaScript, з використанням бібліотеки Chart.js для візуалізації. Код пишеться у середовищі Visual Studio Code з можливістю розгортання на безкоштовних хостингових платформах, таких як GitHub Pages або Netlify.

У світі цифрових технологій особливої ваги набувають інструменти, які дозволяють поєднати інформаційні рішення з екологічною свідомістю. Саме завдяки таким веб-застосункам можна не лише інформувати користувачів, а й мотивувати їх до змін у поведінці, що є важливим кроком до сталого розвитку.

Результатом проєкту має стати доступний та зрозумілий інструмент для тих користувачів, які прагнуть краще усвідомити свій вплив на довкілля та зробити перші кроки до зменшення екологічного навантаження.

					<i>КГ 08. 03 000. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		7

# 1 ОСНОВНИЙ РОЗДІЛ

## 1.1 Теоретичні основи розрахунку екологічного сліду

### 1.1.1 Аналіз проблеми та постановка задачі

Питання екологічної стійкості стає все більш важливим у сучасному суспільстві. Збільшення викидів парникових газів, надмірне використання природних ресурсів та антропогенний вплив на навколишнє середовище – все це сприяє глобальній зміні клімату, що призводить до підвищення температури, танення льодовиків, підвищення рівня моря та інших руйнівних наслідків. Вчені та міжнародні організації визнають, що людство має розв'язувати проблему зміни методів використання ресурсів та мінімізації шкідливого впливу на навколишнє середовище [1].

Однією з головних причин порушення екологічної рівноваги є значні обсяги викидів вуглекислого газу (CO<sub>2</sub>) та інших парникових газів, що спричиняються людською діяльністю. Вже у ХХ столітті вчені дійшли висновку, що навіть помірне підвищення концентрації CO<sub>2</sub> спричиняє підвищення температури на планеті, що і є основним рушієм глобального потепління [2].

Харчовий сектор, сільське господарство, промисловість, енергетика та транспорт є одними з основних джерел цих викидів. На навколишнє середовище прямо чи опосередковано впливає щоденна діяльність кожної людини, яка керує автомобілем, користується електроенергією чи приймає рішення щодо раціону.

Часто люди недооцінюють свій персональний внесок у негативний вплив на довкілля, вважаючи його незначним у порівнянні з великими підприємствами. У порівнянні з величезними енергетичними компаніями чи промисловими підприємствами, люди часто думають, що їхній внесок невеликий. Проте очевидно, що навіть незначний щоденний вибір має значний вплив, якщо взяти до уваги сумарний вплив мільйонів людей по всьому світу [3].

Саме тому потрібно створювати доступні веб-інструменти для індивідуального оцінювання екологічного сліду. Одним з таких інструментів є веб-додаток, який дозволяє користувачам самостійно обчислювати кількість викидів

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		8

CO<sub>2</sub>, спричинених їхньою щоденною поведінкою. Ця стратегія не лише покращує розуміння громадськістю екологічних проблем, але й заохочує людей змінювати свою поведінку, щоб зменшити її шкідливий вплив на довкілля.

Дослідження, проведені Global Footprint Network, показують, що щороку людство використовує більше ресурсів, ніж Земля може відновити. «День екологічного боргу» – це дата, коли ми починаємо жити "в кредит" щодо природних ресурсів. У 2024 році цей день припав на 1 серпня, що свідчить про незначне, але позитивне зрушення порівняно з попередніми роками [4].

Середній екологічний слід на одну особу в Україні у 2024 році склав 2,0 глобальних гектара, тоді як біокапітет становив 3,1 гектара, що означає екологічний профіцит [4].

На рисунку 1.1 представлена глобальна карта екологічного сліду за даними Global Footprint Network. Вона демонструє країни з екологічним профіцитом (зелений колір) та дефіцитом (червоний колір), що дозволяє оцінити масштаби проблеми у світовому контексті.

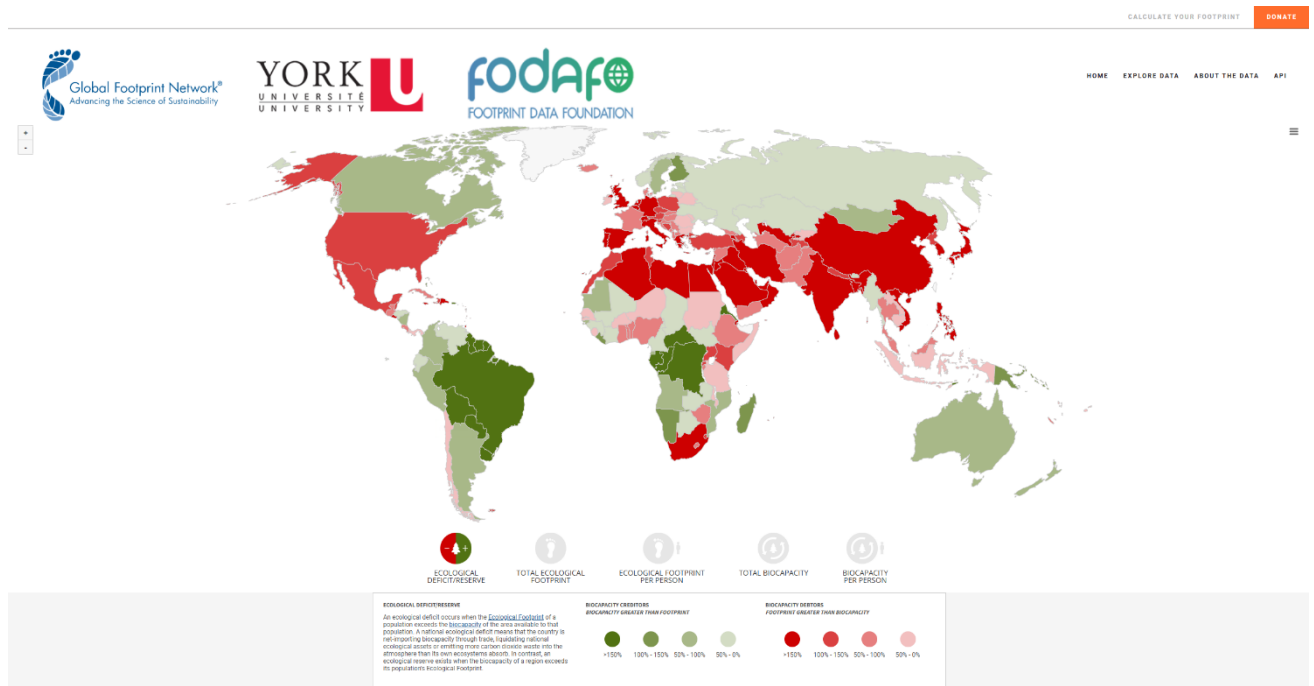


Рисунок 1.1. Глобальна карта екологічного дефіциту та резерву (Global Footprint Network, 2024)

Першим глобальним документом, що встановив чіткі цілі зі скорочення викидів, став Кіотський протокол, підписаний у 1997 році [5].

					<b>КГ 08. 03 001. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		9

Паризька кліматична угода 2015 року стала наступним важливим кроком у глобальній боротьбі зі зміною клімату, яка об'єднала більшість країн світу навколо спільної мети – обмежити глобальне потепління до 2°C, водночас намагаючись утримати його на рівні не вище 1,5°C [6]. Вперше до активного зобов'язання були залучені не лише індустріальні, а й країни, що розвиваються.

Ця угода стала каталізатором для державних політик, стратегій бізнесу та громадянських ініціатив, спрямованих на скорочення викидів, розвиток відновлюваної енергетики та формування екологічно відповідальної поведінки серед населення.

Повсякденні звички, такі як вибір транспорту, джерела енергії, харчові вподобання та підходи до споживання і переробки ресурсів, визначають індивідуальний екологічний слід кожної людини.

Наприклад, використання велосипеда замість авто зменшує викиди приблизно на тонну CO<sub>2</sub> за рік. Перехід на світлодіодні лампи дозволяє скоротити споживання електроенергії до 80%, а зменшення споживання м'яса хоча б на 1 кг скорочує викиди CO<sub>2</sub> на 27 кг.

Інтерактивний веб-застосунок, який дозволяє людям швидко оцінити власний екологічний слід, стане важливим елементом екологічної просвіти.

Згідно з даними Європейського дослідницького центру EDGAR, найбільшими джерелами викидів CO<sub>2</sub> у 2024 році залишаються Китай (понад 26%), США (приблизно 12%) та Європейський Союз (близько 7%) від загальносвітових обсягів [7].

Зростання концентрації парникових газів призводить до посилення екстремальних погодних явищ: частіших посух, повеней і лісових пожеж. В Україні кліматичні зміни проявляються у зменшенні кількості опадів у південних областях, що створює серйозні проблеми для сільського господарства.

У всьому світі реалізуються численні проекти, спрямовані на підвищення обізнаності громадськості з екологічними проблемами. Серед найбільш відомих ініціатив – Година Землі, яку щорічно організовує WWF, Європейський тиждень сталої енергетики, започаткований Єврокомісією, а також численні локальні

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
						10
Зм.	Арк.	№ докум.	Підпис	Дата		

кампанії з екопросвіти. За даними WWF, участь у Годині Землі беруть понад 190 країн та мільйони людей у всьому світі [8].

Вплив забруднення повітря на здоров'я населення є ще одним важливим аспектом. Всесвітня організація охорони здоров'я встановила, що при спалюванні палива утворюються дрібні частинки, які є шкідливими для здоров'я людини. Вони можуть проникати в дихальну та кровоносну системи, що призводить до респіраторних, серцево-судинних та інших стійких захворювань. Згідно з даними, забруднення повітря щороку спричиняє мільйони передчасних смертей у всьому світі. Це ще раз підкреслює необхідність впровадження заходів, спрямованих на зменшення впливу людської діяльності на навколишнє середовище [9].

Успішний досвід провідних європейських міст доводить, що поєднання екопросвіти та інфраструктурних рішень ефективно зменшує викиди. Наприклад, у Копенгагені понад половина мешканців щоденно користуються велосипедами, а у Стокгольмі впроваджено плату за в'їзд у центр, що знизило рівень заторів і забруднення повітря [10].

Досягнення Цілей сталого розвитку ООН, які містять наступні, є критично важливим компонентом всесвітньої боротьби зі зміною клімату:

- Ціль 11: забезпечити сталий розвиток міст і населених пунктів;
- Ціль 12: сприяти сталому споживанню та виробництву;
- Ціль 13: діяти швидко для розв'язання проблеми зміни клімату та її наслідків [11].

Залучення громадянського суспільства має вирішальне значення поряд із зобов'язаннями урядів. Формування екологічної культури можливе завдяки інформуванню населення, участі в просвітницьких ініціативах та створенню простих цифрових інструментів, що допомагають оцінити власний вплив на довкілля.

Ефективним кроком на шляху до екологічної свідомості є впровадження інтерактивного веб-застосунку, що дає змогу оцінити особистий вплив на довкілля та отримати персоналізовані поради для його зменшення.

У межах цього дослідження саме створення такого веб застосунку стало

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		11

ключовим завданням, що поєднує сучасні цифрові технології та екологічну просвіту. Це не лише відповідь на сучасні виклики, а й конкретний крок у напрямі сталого розвитку.

### 1.1.2 Огляд існуючих рішень та підходів

Інструменти оцінки вуглецевого сліду відомі як калькулятори вуглецевого сліду та допомагають людям визначити кількість парникових газів, що викидаються в результаті їхньої повсякденної діяльності. На глобальному рівні існує безліч подібних онлайн-рішень з різним функціоналом та аналітикою. В основному це веб-додатки, які забезпечують швидкий і зручний розрахунок приблизних викидів CO<sub>2</sub> на основі заповнення інтерактивних форм. Завдяки простоті використання та доступності, ці послуги стали значним інструментом для підвищення екологічної свідомості.

Одним з найвідоміших міжнародних прикладів є Калькулятор сліду WWF – інтерактивний онлайн-додаток від Всесвітнього фонду дикої природи. Користувач проходить анкетування про свої звички щодо транспорту, споживання енергії та харчування. На основі наданої інформації система визначає екологічний слід у вигляді кількості планет, необхідних для підтримки такого способу життя [12].

До основних переваг калькулятора WWF можна віднести наступні аспекти:

- Привабливий і зрозумілий дизайн інтерфейсу.
- зручний доступ з будь-якого сучасного гаджета (з комп'ютера до планшетів і смартфонів);
- графічне представлення даних для більш наочного сприйняття інформації.
- У той же час у додатку є і свої недоліки:
- Користувачі не мають можливості вводити конкретні числові значення і можуть вибирати тільки з наданих варіантів.
- Методика не є прозорою, не містить формул розрахунку або вказівки використовуваних коефіцієнтів.
- Результати представлені у вигляді «кількості планет», що може викликати плутанину у користувачів, які звикли до більш стандартного вимірювання в кілограмах або тоннах CO<sub>2</sub>.

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		12

Калькулятор на сайті CarbonFootprint.com розрахований на більш обізнану аудиторію, яка зацікавлена в отриманні детального аналізу свого впливу на навколишнє середовище [13]. Серед його переваг можна виділити:

- можливість надання точної та детальної інформації по кожній сфері споживання;
- надання детальних даних з чітким поділом на різні джерела викидів;
- компаніям надається можливість використовувати бізнес-версію для аналізу вуглецевого сліду своєї корпорації.

Однак у цього засобу є кілька істотних недоліків:

- Для простого користувача інтерфейс може здатися заплутаним і створити труднощі в спілкуванні з системою.
- Дизайн не оптимізований для мобільних пристроїв і здається застарілим.
- Обробка великого обсягу інформації забирає багато часу і сил, що може знизити бажання бути оціненим.

Нижче буде приведена таблиця 1.1 порівняння цих двох калькуляторів.

Таблиця 1.1. Порівняння популярних калькуляторів CO<sub>2</sub>

<b>Критерій</b>	<b>WWF Footprint Calculator</b>	<b>CarbonFootprint.com</b>
Цільова аудиторія	Широка публіка, молодь	Обізнані користувачі, представники бізнесу
Дизайн і доступність	Інтуїтивний інтерфейс, адаптований до мобільних пристроїв	Складний інтерфейс, неадаптований для смартфонів
Введення даних	Обмежений вибір варіантів	Введення точних числових даних
Деталізація результатів	Загальний індекс у вигляді кількості планет	Розподіл викидів за категоріями у кг або тоннах CO <sub>2</sub>
Прозорість методики	Відсутність чітких формул і пояснень	Частково описана методологія
Підтримка для бізнесу	Відсутня	Є бізнес-версія для компаній
Гейміфікація / мотивація	Є елементи дизайну, що заохочують	Відсутні

Крім глобальних рішень, існують також спеціалізовані калькулятори вуглецевого сліду на регіональному рівні, які розроблені різними країнами з урахуванням особливостей своєї території. Наприклад, у Великій Британії, Німеччині та Франції існують національні онлайн-сервіси на цю тему; Вони враховують місцеві нормативи споживання електроенергії та транспортну інфраструктуру при проведенні розрахунків впливу на довкілля. Це забезпечує більш точні оцінки для зацікавлених країн. Однак такі сервіси не завжди адаптовані для використання іноземними користувачами; Вони часто доступні лише місцевими мовами і не претендують на глобальну універсальність.

В Україні такі інструменти поки що не набули широкого поширення. Хоча деякі екологічні програми пропонують зручні калькулятори, їхні можливості обмежені, а дані розрахунків не завжди відображають реальні умови життя українців – такі як поточні тарифи, поведінкові моделі чи моделі споживання ресурсів.

Згідно з дослідженнями користувацького досвіду (UX), більшість людей вважають за краще використовувати інструменти, які є збалансованими – простими у використанні та аналітичними одночасно. Заповнення форм із занадто великою кількістю деталей може відлякати деяких користувачів через втому або брак часу. З іншого боку, занадто загальні результати без детальних пояснень і числових показників можуть викликати сумніви в достовірності оцінки.

Мобільні додатки, такі як JouleBug або Oroeco, заслуговують на особливу увагу за гейміфікований підхід до оцінки свого вуглецевого сліду та мотивацію користувачів змінювати свою поведінку через взаємодію з соціальними мережами, віртуальними винагородами та спільнотами. Така механіка активізує користувачів та сприяє формуванню екологічної спільноти. Школярі та студенти можуть використовувати різноманітні інструменти для навчання про вплив людської діяльності на навколишнє середовище на планету в рамках програм з виховання відповідального ставлення до ресурсів та навколишнього середовища.

З досвіду використання таких сервісів ми знаємо наступне: користувачі найбільш прихильно ставляться до додатків

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		14

- мати зрозумілий і зручний інтерфейс;
- забезпечують швидкий результат;
- включати пояснення та натяки;
- дати можливість поділитися своїми успіхами з друзями;
- Керуйте трансформацією за допомогою ігрових елементів або порад.

Крім раніше згаданих плюсів і мінусів, важливо також враховувати цільову аудиторію кожного калькулятора. Наприклад, калькулятор вуглецевого сліду від WWF орієнтований в першу чергу на молодь і студентів і широке коло користувачів без глибоких технічних знань. У той час як CarbonFootprint.com розрахований на більш підготовлених користувачів і включає представників бізнесу, яким потрібні точні дані для корпоративної звітності або екологічної сертифікації.

Разом з тим, інструменти оцінки екологічних показників мають значний потенціал у формуванні екологічної свідомості в нашій країні. Вони можуть бути застосовані в навчальних програмах шкіл та університетів для навчання учнів та студентів принципам сталого розвитку. Такі ресурси бізнес може використовувати для підвищення своєї відповідальності та створення екологічно соціально прийнятної іміджу.

Малі та середні підприємства в Україні можуть скористатися різними інструментами для оцінки свого впливу на довкілля, підвищення екологічної відповідальності та зміцнення ділової репутації компанії. Це також відкриває нові перспективи для сертифікації продукції за європейськими стандартами, що є важливим кроком на шляху інтеграції в європейський ринок.

Однак сучасні підходи мають свої недоліки: одними з них є поверховість або недостатня точність у поданні інформації, або надмірна складність і надмірно спрощене подання даних. Саме тому виникає потреба в калькуляторах нового покоління – вони повинні бути простими у використанні і доступними для користувачів з гнучкими налаштуваннями функціоналу і можливістю проведення більш глибокого аналізу з урахуванням локальних особливостей.

Одним з перспективних напрямків розвитку є використання штучного

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		15

інтелекту і машинного навчання для автоматичного аналізу поведінки користувачів в інтернеті. Нещодавні дослідження показують, що взаємодія з цифровими сервісами, такими як Google Fit або Apple Health, може допомогти створювати персоналізовані рекомендації на основі реальних даних про активність користувачів. Це не тільки спрощує роботу з калькулятором активності кроків і калорій користувача, але й робить пропозиції більш персоналізованими та актуальними для конкретної людини.

Крім того, можливість створення спільнот за інтересами може бути додана в майбутніх версіях калькуляторів. Користувачі зможуть об'єднатися в групи, щоб спільно зменшити негативний вплив на навколишнє середовище та поділитися досягненнями та взаємною мотивацією. Такий підхід сприяє більш соціальній та захоплюючій стороні процесу, що неодмінно підвищить інтерес населення до природоохоронних практик.

Тому створення нового українського калькулятора екологічного сліду видається розумним та актуальним рішенням. Такий інструмент має бути зручним у використанні для широкої аудиторії та адаптованим до українських умов. Також він повинен бути зручним для повсякденного використання, а також для освітніх і корпоративних цілей.

Вивчення існуючих екологічних калькуляторів дозволяє виявити як позитивні приклади їх використання на практиці, так і недоліки, які важливо враховувати при розробці нового рішення. Це хороша відправна точка для подальшого технічного та архітектурного проектування веб додатку.

## **1.2 Вибір та обґрунтування технологій реалізації**

### **1.2.1 Вибір технологій для реалізації**

Для створення дипломної роботи було використано текстовий редактор Visual Studio Code (VS Code) – один з найбільш зручних і потужних інструментів для розробки інтерфейсів веб-додатків (frontend). Він надає зручне середовище для написання коду HTML5, CSS3 та JavaScript ES6, який ідеально відповідає цілям мого проекту. Завдяки різноманітності плагінів, таких як Live Server, Prettier,

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		16

Emmet та інші, процес розробки помітно прискорився і став більш організованим [14].

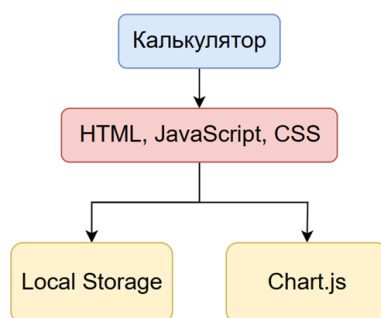


Рисунок 1.2. Схема використаних технологій у проєкті

Однією з вагомих переваг VS Code є його портативність. Середовище розробки може запускатися із зовнішнього носія інформації, що було особливо корисно при роботі над дипломним проєктом, коли потрібно було вносити зміни на різних пристроях: вдома, в університеті або в дорозі.

Ще одним корисним інструментом у процесі розробки стала функція живого попереднього перегляду через розширення Live Server. Вона дозволила миттєво бачити результат кожної зміни в браузері без необхідності вручну оновлювати сторінку. Це значно прискорило процес створення адаптивного дизайну та налаштування візуальних елементів.

Нижче продемонстровано приклад роботи Live Server у середовищі Visual Studio Code: зміни в HTML-коді одразу відображаються у вікні браузера, що значно пришвидшує процес верстки.

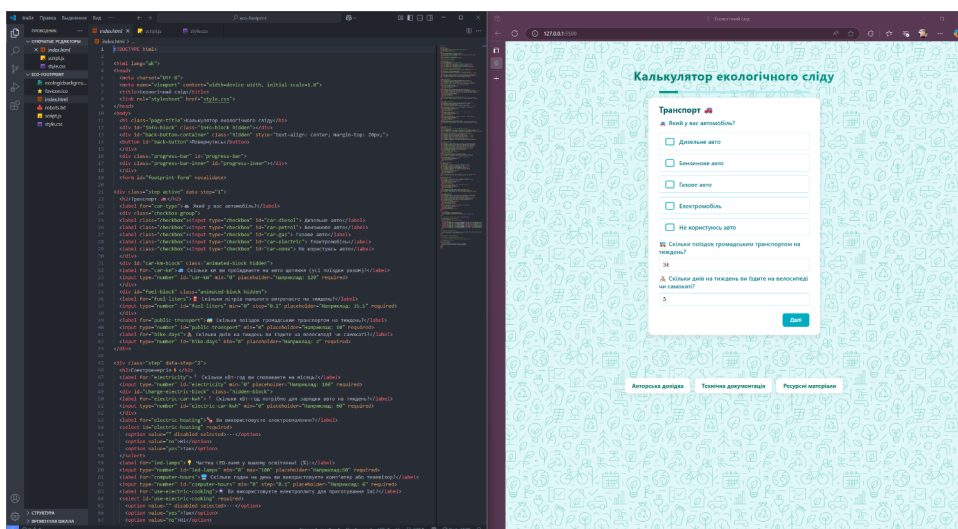


Рисунок 1.3. Робота в середовищі Visual Studio Code з розширенням Live Server

При виборі таких ключових технологій, як HTML, CSS, JavaScript, було чітко розуміння переваг їх використання: забезпечення прозорості, передбачуваності, повного розуміння архітектури та освітньої цінності. Такий підхід дозволив не тільки створити робочий продукт, а й глибше вивчити основи сучасної веб-розробки, минаючи зайві абстракції, характерні для фреймворків.

HTML5 використовує як базову основу для структури веб-сторінки. Ця мова розмітки широко використовується в сучасній веб-розробці та надає широкі можливості для створення логічно структурованого та семантично значущого контенту. На відміну від попередніх версій HTML, HTML5 підтримує значну кількість нових тегів, що дозволяє розробнику не тільки оформити сторінку візуально приємним, але і адекватно розділити її контент на семантичні блоки.

HTML використовує спеціальні мітки для організації контенту на веб-сторінках – тексту, фотографій, відео та іншого контенту. `<h1>`, `<title>`, `<body>`, `<header>`, `<footer>`, `<article>`, `<section>`, `<div>`, `<span>`, `<a>`, `<img>`, `<canvas>`, `<video>`, `<ul>` які визначають структуру контенту і керують відображенням в браузері.

Кожен елемент HTML складається з тега з назвою в кутових дужках і може бути як парним (з відкриваючим і закриваючим тегами), так і одиночним. Теги не враховують регістр, але сучасні стандарти рекомендують використовувати малі літери (наприклад: `<title>` замість `<TITLE>`) для поліпшення читабельності та відповідності стилю в HTML5.

Семантичні компоненти, такі як `<section>`, `<article>`, `<header>`, `<footer>`, `<main>`, `<nav>` та інші не тільки полегшують читання коду, але й сприяють кращій індексації сайту пошуковими системами. Наприклад, тег `<section>` використовується для логічного групування контенту в конкретні групи за тематикою; в той час як `<header>` містить заголовки і короткий виклад інформації; А блок `<footer>` може містити контактні дані або посилання для навігації сайтом. Це робить сторінку зручною для використання як звичайними користувачами, так і допоміжними технологіями, такими як скрінрідери для людей з обмеженими можливостями.

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		18

На рисунку 1.4 показано приклад типової семантичної структури сторінки.

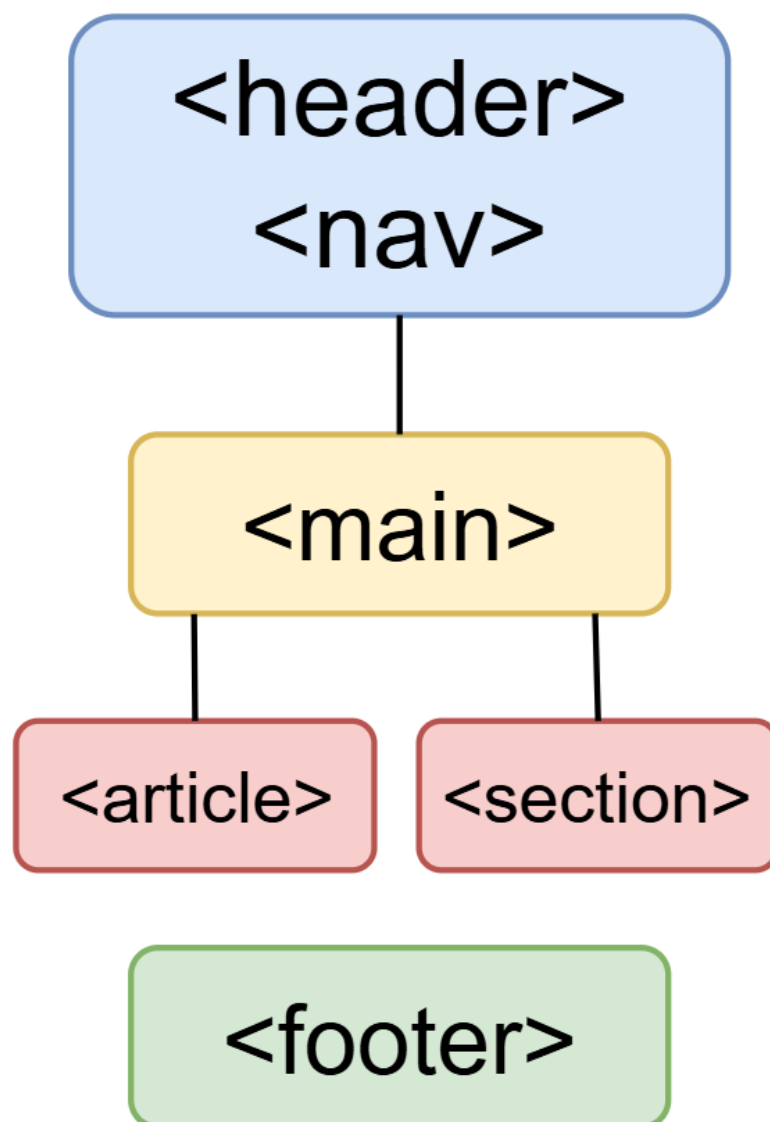


Рисунок 1.4. Семантична структура HTML5-компонентів

Фреймворк HTML5 дуже гнучкий у розробці та спрощує підтримку проектів у майбутньому. Такий підхід робить кожен блок коду більш зрозумілим і автономним, що грає важливу роль в командному розвитку або при розширенні сайту. Крім того, відповідність стандартам HTML5 сприяє кращій сумісності з різними браузерами і забезпечує коректне відображення сайту на більшості сучасних інтернет-браузерів.

CSS3 відіграє важливу роль у створенні сучасного та зручного користувацького досвіду для веб-додатків завдяки своїм можливостям візуального дизайну. У проекті для стилізації використовуються такі технології, як Flexbox і

CSS Grid – це сучасні підходи до верстки, які дозволяють без проблем розміщувати елементи на сторінці незалежно від розміру екрану пристрою користувача. Це сприяє адаптивності інтерфейсу програми і забезпечує коректне відображення як на великих моніторах, так і на мобільних пристроях.

CSS Grid та Flexbox – це два основних інструменти для створення сучасних макетів у веб-розробці. Flexbox працює як одновимірний модель, що дозволяє вирівнювати елементи в одному напрямку – по горизонталі або вертикалі. Він ідеально підходить для простих макетів, таких як вирівнювання кнопок у рядок або побудова вертикальних блоків. У Flexbox головним є вміст, який диктує структуру.

Натомість CSS Grid реалізує двовимірну модель, де елементи розміщуються по рядках і стовпцях одночасно. Це дозволяє будувати складні макети зі структурованою сіткою. Grid орієнтований насамперед на макет і є більш гнучким для створення складних інтерфейсів. Проте при великій кількості елементів Grid може мати нижчу продуктивність, у той час як Flexbox зазвичай працює швидше для простих структур.

На рисунку 1.5 наочно показано, як Flexbox розміщує елементи в один ряд, тоді як CSS Grid дозволяє формувати повноцінну двовимірну сітку. Це демонструє ключову різницю між цими підходами до верстки.

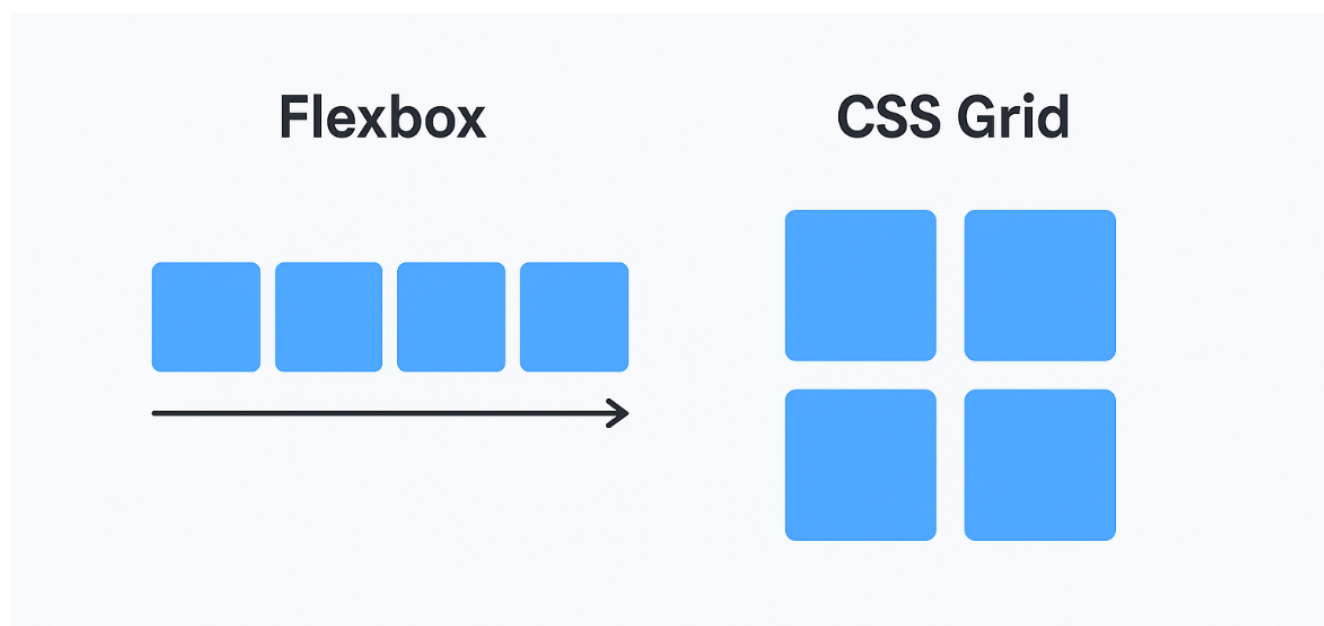


Рисунок 1.5. Порівняння принципів роботи Flexbox і CSS Grid

У проєкті також широко використовувалися CSS-змінні (custom properties), сучасний стандарт, який дозволяє централізовано керувати кольорами, шрифтами та відступами. Каскадні стилі також використовувалися для підтримки єдиного візуального стилю по всій структурі, дозволяючи елементам успадковувати властивості та централізовано керувати стилями з мінімальним повторенням коду. Такий підхід робить код більш читабельним і полегшує внесення змін до стилю – це швидше і з меншим ризиком виникнення помилок у процесі.

Медіа-запити використовувалися для адаптації дизайну під різні пристрої. На екранах розміром менше 600 пікселів були застосовані змінені відступи та розміри шрифтів для кращої читабельності. Цей метод узгоджується зі стратегією проєктування інтерфейсів з урахуванням мобільних пристроїв.

JavaScript в ES6 грає ключову роль в реалізації взаємодії користувача з веб-додатком. В рамках проєкту весь інтерактивний функціонал був реалізований на чистому JavaScript без використання сторонніх бібліотек. Це дозволило зберегти контроль над структурою, продуктивністю та чіткістю коду, а також зробило поведінку калькулятора більш передбачуваним.

Код був написаний з використанням сучасних можливостей мови ES6. Використовувалися змінні let та const, стрілкові функції (=>), обробники подій (addEventListener), а також методи роботи з елементами DOM через querySelector, classList, innerHTML та style. Такий підхід дозволив організувати калькулятор у вигляді односторінкового додатку (SPA) – всі етапи можна проходити без перезавантаження сторінки: контент оновлюється динамічно в міру переміщення між кроками.

Ця функціональність також дозволяє користувачам зберігати поточний стан на кожному етапі процесу та повертатися для внесення змін без необхідності повторного введення інформації. Навігація по етапах здійснюється автоматично програмно шляхом управління класами активності (active, hidden) елементів DOM для полегшення анімації та додавання нових кроків у майбутньому.

Особлива увага приділяється чистоті коду та повторному використанню функцій: у проєкті використовуються загальні обробники подій та допоміжні

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		21

методи, що робить проект більш масштабованим та підтримуваним. Стандарт ES6 також зменшив кількість коду і поліпшив його читабельність.

У межах односторінкового додатку важливо забезпечити збереження введених даних між етапами взаємодії. У проекті ця задача реалізована за допомогою localStorage – технології, яка дозволяє зберігати інформацію локально в браузері без підключення до сервера. Це забезпечує швидке завантаження, збереження конфіденційності користувача, а також можливість повернення до заповнених форм навіть після оновлення сторінки. Простота впровадження localStorage і його автономність зробили його ідеальним рішенням для освітнього проекту, націленого на демонстрацію основ веб-розробки.

Для прикладу, нижче наведено фрагмент коду, який реалізує зчитування збережених у localStorage даних при завантаженні сторінки.

```
window.addEventListener("DOMContentLoaded", () => {
  document.querySelectorAll("input, select").forEach(el => {
    const saved = localStorage.getItem(el.id);
    if (saved !== null) el.value = saved;
  });
});
```

Це дозволяє користувачам продовжувати роботу з калькулятором без повторного введення інформації, навіть після оновлення сторінки чи закриття вкладки, що особливо зручно для багатоетапних форм.

Використання такого стека технологій як HTML5, CSS3, JavaScript ES6, localStorage і середовища розробки Visual Studio Code – повністю відповідає цілям і обсягам дипломного проекту. Ці інструменти дозволяють створити ефективний і зручний веб-додаток без зайвих архітектурних складнощів або залежності від сторонніх рішень. Крім того, вони забезпечують надійну роботу калькулятора в різних умовах і сприяють формуванню практичних навичок веб-розробки.

Такий підхід допомагає зосередитися на логіці, дизайні та користувацькому досвіді – основних аспектах розробки сучасних цифрових рішень з акцентом на екологію. Простота використовуваних технологій у поєднанні з прозорістю та освітньою цінністю робить проект не тільки корисним з точки зору кінцевого результату, а й значним кроком у професійному зростанні.

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		22

## 1.2.2 Архітектура веб-застосунку

Проектування структури веб-додатку відіграє важливу роль у створенні надійної, масштабованої та придатної для використання системи роботи. В рамках дипломного проекту було приділено особливу увагу модульності архітектури додатку і послідовної логіці її компонентів, щоб чітко визначити обов'язки кожного функціонального елемента системи. Такий підхід дозволив забезпечити прозорість у кодї програми та зробив процес налагодження простішим та зрозумілішим для розробників. Крім того, додавання нових функцій стало менш ризикованим для існуючого функціоналу завдяки чіткому розмежуванню завдань між компонентами. Обрані архітектурні рішення зробили мій додаток стабільним, гнучким і простим в оновленні.

Основний акцент архітектури полягав у тому, щоб створити логічний шлях для користувача, який максимально наближений до реальних сценаріїв взаємодії. Кожна категорія екологічного сліду, починаючи від транспорту і закінчуючи споживанням ресурсів, представлена у вигляді окремого розділу з унікальним набором полів і логікою обробки даних. Це дозволило забезпечити зручність використання, поділ логіки всередині кожного блоку, а також можливість розширити інтерфейс без необхідності повної перебудови структури програми. З точки зору дизайну, інтерфейс виглядає як багатосторінковий додаток, але технічно він реалізований як односторінковий додаток з посиланнями і динамічним завантаженням контенту за допомогою JavaScript. Таке рішення допомогло знизити навантаження на мережу і забезпечити миттєву реакцію на дії користувача.

Кожен етап форми в додатку представлений у вигляді окремого HTML-блоку з класом "step", що робить його не тільки важливою частиною інтерфейсу для користувача, але і логічною одиницею архітектури. Такий підхід дозволив чітко структурувати різні тематичні категорії: транспорт, продукти харчування, покупки, вода, електрика, газ і повсякденні звички. Це не тільки сприяє впорядкованості коду, а й забезпечує зручність для розширення функціоналу: додати новий тематичний блок можна легко, вставивши ще один контейнер з

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		23

класом "step" з відповідними полями та інтегрувавши його в загальну логіку навігації.

Логіка взаємодії з цими блоками здійснюється шляхом застосування окремих функцій, незалежних від змісту конкретних кроків. У проекті використовувалися функції `showStep(index)`, `nextStep()` та `prevStep()`, які відповідають за перемикання активного розділу шляхом додавання або видалення CSS-класів `.active` та `.hid`. Важливо зазначити, що ці функції не працювали з даними у формі; вони лише діяли зі структурою ДОМ відповідно до принципу поділу концернів.

Окремі блоки були виділені для логіки виконання обчислень (`calculateResult()`, `updateChart()` та `sortChartData()`). Це полегшило розробку і налагодження формул, а також відображення результатів незалежно від основної форми програми. Наприклад, зміна розрахункових коефіцієнтів або порядку категорій на діаграмі не вимагала переписування логіки кроків або інтерфейсу.

Важливо відзначити, що структура форми проектувалася з розрахунком на повторне використання і адаптацію. Архітектура додатку розроблена так, що базові елементи – покрокові переходи, обробка подій, валідація та відображення даних – ізольовані один від одного. Це дозволяє легко реалізувати функціонал калькулятора в іншому проекті або змінити тему без необхідності повністю переробляти внутрішню логіку.

При створенні архітектури особлива увага була приділена управлінню станом елементів форми за допомогою CSS-класів `.active` і `.hided`. Ці класи використовувалися для контролю видимості та доступності різних етапів форми. Крім простого перемикання видимості, для всіх інтерактивних елементів інтерфейсу була розроблена гнучка система станів.

Кожен елемент форми в структурі може перебувати в різних станах для точного налаштування її поведінки. Наприклад, поля введення можуть бути порожніми або містити помилки, а також позначатися додатковими невірними класами під час перевірки даних користувача. Стан кожного елемента стає важливою частиною архітектури і не тільки відображає візуальні ефекти.

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		24

Для управління цими станами використовується загальний метод валідації: перед переходом до наступного кроку додаток викликає функцію валідації, яка проходить по всіх активних полях поточного блоку. При виявленні помилок відповідні елементи підсвічуються, прокручуються до першого з них і фіксуються перехід далі. Це робиться не шляхом прив'язки до індивідуальних ідентифікаторів, а за допомогою загального обходу активної блокової структури DOM для підвищення гнучкості коду.

Також була впроваджена функція «зворотного зв'язку в реальному часі»: якщо користувач виправить помилку в полі, клас `.invalid` буде автоматично видалений. Це допомагає знизити рівень роздратування і робить взаємодію з інтерфейсом більш природною. Всі ці зміни є частиною архітектурної стратегії, яка спрямована на покращення користувацького досвіду та стійкості архітектури без перевантаження логіки форми.

Важливо, що логіка збереження і відновлення даних реалізована незалежно від логіки вводу, що дозволяє масштабувати рішення в майбутньому – наприклад, додати імпорт, синхронізацію чи збереження форм між пристроями. Крім того, такий підхід повністю відповідає концепції архітектури клієнта: всі дії виконуються в браузері без необхідності доступу до сервера. Це забезпечує швидку реакцію системи, знижує навантаження на мережу та гарантує конфіденційність даних користувача за рахунок їх локального зберігання.

Окрім прогрес-бара, в архітектурі додатку є система контекстних підказок для поліпшення інтерфейсу. При певних діях користувача на екрані з'являється інформаційний блок з підказками. Наприклад, якщо користувач не заповнює потрібне поле або вибирає певний вид транспорту, система пропонує поради, як це може вплинути на кінцевий результат. Такий підхід не тільки покращує навігацію інтерфейсу, але і робить взаємодію більш доцільною. З цієї логічної точки зору, архітектура використовує окремий модуль `#info`-блоку та функції `showInfo(type)` та `hideInfo()` для управління ним. Це дає можливість централізовано керувати всіма текстовими повідомленнями, не впливаючи на базову логіку кроків калькулятора.

Одним з важливих поліпшень в адаптивній архітектурі є додавання умовних

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
						25
Зм.	Арк.	№ докум.	Підпис	Дата		

блоків, які з'являються тільки при певних умовах виконання. Наприклад, область для введення кількості газового палива, використаного транспортним засобом, стає видимою тільки після того, як користувач вибере транспортний засіб, що працює на газі. Для цього в коді використовуються логічні перевірки статусу чекбоксу та функції для відображення або приховування відповідних елементів інтерфейсу. Ця модульна реалізація дозволяє гнучко масштабувати сценарії взаємодії, не додаючи складності базовій структурі програми.

Отже, архітектура калькулятора відповідає сучасним стандартам, залишаючись гнучкою і зрозумілою для подальшого розвитку. Всі основні компоненти модульні та ізольовані один від одного, що дозволяє легко вносити зміни та тестувати кожен елемент окремо. Такий підхід дозволив створити не тільки зручний для користувача інструмент, але і надійну технічну базу з можливістю адаптації під майбутні потреби – від розширення функціоналу калькулятора до додавання нових функцій без повного переписування коду. Такий підхід робить розроблене рішення стабільним, гнучким і таким, що відповідає сучасним практикам сучасної веб-розробки.

### **1.2.3 Файлова структура проєкту**

Файлова структура веб-додатку відіграє важливу роль у полегшенні розробки та підтримки програми, а також у її потенційному масштабуванні. Працюючи над дипломним проєктом, було розроблено просту і логічно продуману структуру з акцентом на юзабіліті і швидку навігацію. Основна увага була приділена чіткому розподілу функціоналу між файлами: кожен з них відповідає за певний аспект – розмітку, стилізацію, логіку програми або додаткові елементи. Це дозволяє легко вносити зміни без ризику небажаного накладання або повторення коду.

Так як додаток працює повністю на стороні користувача і не вимагає підключення до сервера або виконання скриптів сервера, всі файли були поміщені в одну головну папку для зручності запуску проєкту в будь-яких умовах, включаючи роботу в автономному режимі без доступу в інтернет. Така організація також ідеально підходить для розміщення на платформах, що підтримують

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		26

статичні сайти, де можна розгорнути весь проєкт без необхідності додаткового налаштування.

Розглянемо на рисунку 1.6 та 1.7, як виглядає структура проєкту у різних середовищах.

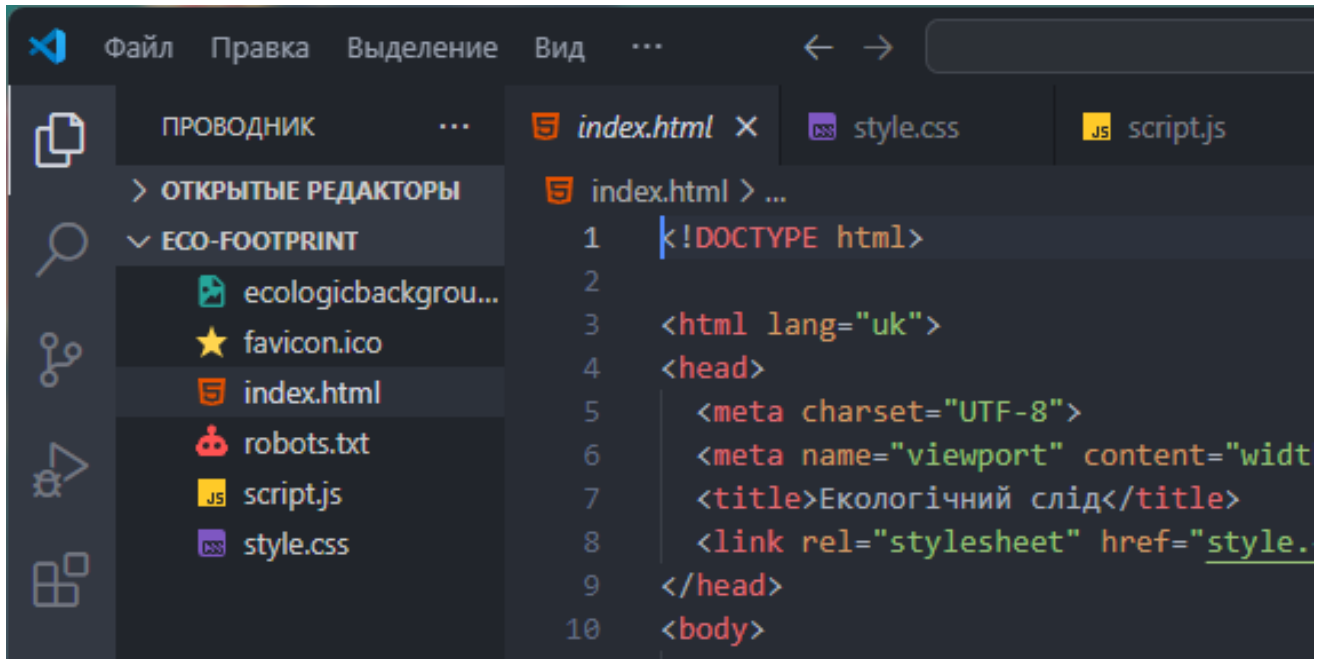


Рисунок 1.6. Структура проєкту у редакторі Visual Studio Code

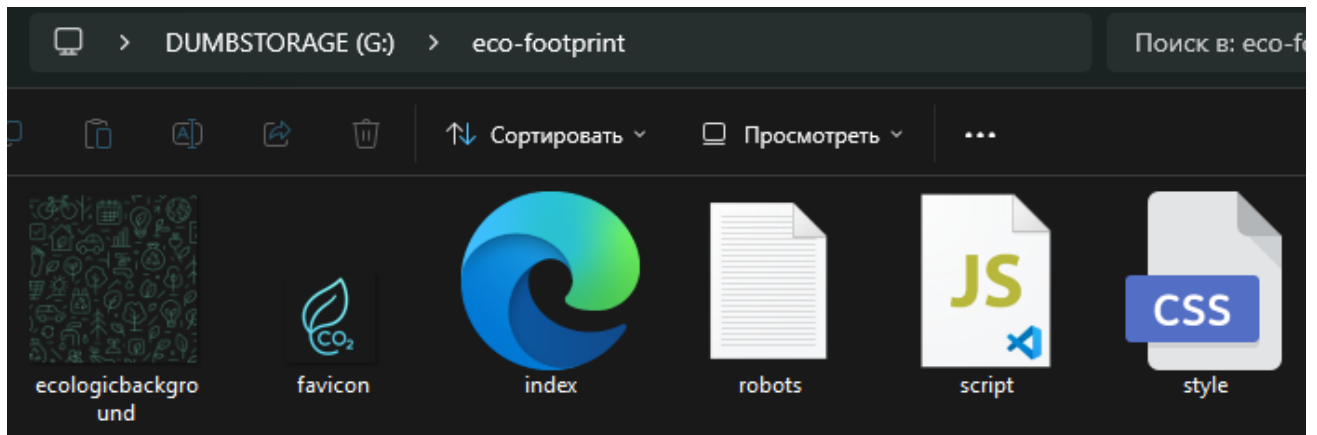


Рисунок 1.7. Вигляд проєкту у файловому менеджері Windows

Кожному файлу в проєктній організації призначається певне завдання і він розміщується в головному каталозі для зручної навігації та обслуговування.

index.html – основний файл входу в додаток. Цей HTML-документ містить всю необхідну розмітку: від покрокової форми калькулятора та інформаційних блоків до індикатора прогресу та кнопки розрахунку; Також є контейнер для діаграми та кнопок навігації. Зовнішні стилі та скрипти підключаються

					КГ 08. 03 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		27

безпосередньо до цього файлу, щоб забезпечити централізований контроль над усією логікою клієнта.

Файл `style.css` містить каскадну таблицю стилів для оформлення інтерфейсу користувача веб-сторінки. Стили тут визначаються не тільки для основної структури (відступи, шрифти, колірна схема), але і для інтерактивних елементів – наприклад, підсвічування помилок, покрокова анімація і адаптивний відступ для мобільних пристроїв з візуальними підказками. Змінні CSS і медіа-запити також використовуються для більш гнучких налаштувань відображення на різних пристроях.

Сценарій `script.js` реалізувати базову логіку проекту. Цей файл містить функції для модифікації активних блоків форми (`next step()`), попереднього `step()`), перевірки введених користувачем даних, збереження прогресу в `localStorage`, обчислення викидів та оновлення візуалізацій даних. У ньому також розміщені всі обробники подій, кнопки для взаємодії з користувачем і динамічний дисплей діаграми.

Іконка `favicon.ico` додана у вкладку браузера для візуального позначення екологічної спрямованості сайту. Цей невеликий елемент доповнює інтерфейс і допомагає користувачам швидко визначити потрібну вкладку серед інших.

`ecologicbackground.png` – це картинка, яка використовується в якості фону для всієї сторінки. Він надає візуальний стиль і атмосферу сталого розвитку, підкреслюючи тематику проекту і роблячи інтерфейс більш привабливим.

Файл `robots.txt` – це додатковий файл, який встановлює правила індексації сайту пошуковими системами або забороняє її виконання. У локальному проекті роль цього файлу може бути незначною; Однак його наявність свідчить про відповідність сучасним веб-стандартам.

Організація файлів у додатку ґрунтується на кількох продуманих принципах. Ці принципи допомагають створити зручне середовище для поточного процесу розробки та забезпечують гнучкість для майбутніх змін.

Простота і зрозумілість – ключові моменти цього проекту. Структура розробки спланована так, щоб навіть новачок у програмуванні міг швидко в ній

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		28

розібратися. Відсутність складних вкладених папок, просте іменування файлів і послідовні з'єднання файлів (спочатку HTML, потім CSS і JS) дозволяють легко вивчити код при першому відкритті проекту. Це особливо цінно в освітньому контексті: чітка структура полегшує розуміння рецензентами та запобігає можливим труднощам під час демонстрації роботи над проектом.

Структурування логіки програми на окремі файли є гарною практикою розробки: `index.html` містить структуру та контент сторінки; `style.css` відповідає за його візуальне оформлення та поведінку; `A script.js` включає в себе логіку і базовий функціонал програми. Цей метод відповідає принципу поділу концернів, що сприяє зменшенню зв'язків між компонентами та підвищенню їх незалежності один від одного. Це дозволяє вносити зміни в логіку без необхідності змінювати стилі і навпаки.

Готовність до масштабування завжди важлива. Незважаючи на компактну організацію, проект розроблений з можливістю подальшого розширення. При необхідності ви можете легко додати додаткові директорії (`/js`, `/css`, `/img`, `/components`) і перемістити в них відповідні файли без зміни загальної структури – просто оновіть шляхи підключення. Такий підхід також дозволяє легко підлаштовуватися під потреби масштабного проекту: додавання багатомовної підтримки або шаблонізації за потреби не викличе серйозних труднощів.

Також важливо відзначити, що структура була організована без використання асемблерів або модульних систем компіляції. Це був свідомий вибір – спрощення архітектури дозволило їй зосередитися на користувацькому досвіді (UX), покроковій логіці та візуальному дизайні. Однак важливим моментом, на який слід звернути увагу, є те, що поточна організація файлів повністю сумісна з можливістю інтеграції цих інструментів у майбутньому.

Для цього випускного проекту ми віддали перевагу використанню простого і зрозумілого архітектурного рішення без використання будівельників (наприклад, Webpack або Vite), препроцесорів CSS (таких як SASS або LESS) або зовнішніх бібліотек JavaScript. Такий підхід був обраний з урахуванням специфіки проекту і поставлених перед ним завдань.

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		29

По-перше, додаток функціонує повністю на стороні користувача і не вимагає бекенда для роботи. Усі розрахунки, валідація, візуалізація та навігація даних виконуються за допомогою JavaScript без участі сервера. Це дозволяє уникнути необхідності створення, компіляції та обслуговування серверної інфраструктури, що спрощує процес розгортання та зменшує потенційну кількість помилок.

По-друге, важливо враховувати простоту запуску та валідації проекту як одну з головних вимог. Оскільки сторонні інструменти не потрібні, програму можна запустити лише одним клацанням миші з Live Server або відкрити в браузері безпосередньо з локального диска. Також є можливість розміщувати на Netlify або GitHub Pages без необхідності додаткової підготовки. Це особливо корисно для презентації роботи вчителю або при захисті проекту.

По-третє, відсутність складної технічної інфраструктури дозволила сконцентруватися на головному – інтерфейсі для користувачів, логіці розрахунку екологічного сліду і забезпеченні приємного користувальницького досвіду (UX). В умовах обмеженого часу та з огляду на освітню мету проекту, це рішення виявилось найефективнішим: воно дало можливість не розпорозуватися на другорядні аспекти та зосередитися на цінностях та функціоналі програми.

Структурування проекту було простим та інтуїтивно зрозумілим. Кожен файл виконує свою певну функцію без зайвих ускладнень, а при необхідності структуру можна зручно розширити – наприклад, за рахунок додавання нових розділів або багатомовності та бази знань програми. Навіть найпростіша програма має простір для розробки без необхідності перепроєктувати її з самого початку.

В кінцевому підсумку такий підхід створює міцний фундамент для майбутніх проектів і дозволяє використовувати більш складні інструменти: фреймворки, модулі, компоненти і системи, збірки. Переходячи від простих конструкцій до більш складних рішень, розробник отримує послідовний і глибокий досвід, який відіграє ключову роль в процесі навчання і розвитку професійних навичок.

#### **1.2.4 Потенціал масштабування та подальшого розвитку веб-застосунку**

В рамках мого дипломного проекту був розроблений екологічний

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		30

калькулятор з наміром створити простий у використанні інструмент без необхідності підключення до сервера або складних систем авторизації. Основний акцент був зроблений на зручність для користувача, точність розрахунків і яскраве візуальне представлення результатів. При цьому спочатку були враховані архітектурні принципи для забезпечення гнучкості проекту, як з технічної точки зору (структура коду, зберігання даних і незалежність компонентів), так і з точки зору логіки взаємодії (модульність блоків і універсальність алгоритмів розрахунку). Цей проект не тільки стає об'єктом навчання, а й має потенціал стати основою для освітніх та бізнес-рішень. У цьому розділі ми розглянемо технічну та функціональну еволюцію програми, з можливими способами масштабування на різні платформи, аудиторії та типи даних.

Завдяки модульній структурі додатку і чіткому розподілу завдань між його частинами, екологічний калькулятор має великий потенціал для зростання і впровадження нових можливостей. Кожен блок в поточній версії організований у вигляді окремого розділу зі своєю логікою роботи, що дозволяє легко додавати нові категорії без зміни існуючих. У звичні розділи можна додати інформацію про водний транспорт та міжміські або міжнародні рейси. Також варто звернути увагу на використання будівельних матеріалів у побуті та онлайн-шопінгу. Не забудьте також обговорити цифрові технології – від перегляду відео до зберігання даних у хмарі – і важливість створення цифрового сліду.

Кожен конкретний блок функцій може бути створений у вигляді додаткового контейнера з класом `.step`, що містить унікальні параметри опису та формули для обчислень. При цьому базова структура навігації та логіка переходів між етапами залишаються незмінними, що спрощує процес освоєння нових компонентів. Такий підхід до масштабування не вимагає повної перебудови архітектури – досить ввести в логічну частину нові ідентифікатори, розрахункові коефіцієнти і пов'язані з ними події.

Крім того, в рамках розробки проекту ми можемо розглянути можливість впровадження функції умовного відображення блоків. Це означає, що певні розділи можуть з'являтися на основі відповідей користувача на попередні

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		31

запитання. Наприклад, якщо користувач вкаже на наявність приватного будинку, автоматично з'явиться блок з інформацією про опалення дровами або вугіллям. Такий підхід зробить користувацький досвід більш гнучким та індивідуальним відповідно до конкретних умов використання; Крім того, сам калькулятор можна використовувати як універсальний інструмент для самодіагностики навколишнього середовища.

Одним з найбільш перспективних напрямків для розширення функціональних можливостей додатку є інтеграція зовнішніх джерел даних і сервісів по API. Це дозволить не тільки оновлювати інформацію про розрахунки за потреби, але й покращувати користувацький досвід, надаючи більш точну та персоналізовану інформацію. Наприклад, можна використовувати відкриті бази даних про споживання енергії населенням або статистику викидів вуглекислого газу по регіонах, щоб розширити функціонал програми і підвищити її цінність для користувачів.

Наприклад, за допомогою інформації від місцевих енергетичних компаній калькулятор може автоматично розраховувати коефіцієнти для розрахунку викидів в залежності від енергетичної структури регіону (частка вугілля, газу, відновлювальних джерел тощо). Це дозволить не тільки враховувати введене користувачем значення в кіловат-годинах, але і коригувати його з урахуванням місця розташування користувача для максимально точного результату.

Ми також можемо вбудовувати картографічні сервіси в наш додаток, такі як OpenStreetMap або Google Maps API, щоб автоматично визначати маршрути для користувачів. Користувач зможе лише вказати початкову та кінцеву точки шляху; Після цього додаток самостійно розрахує відстань в кілометрах, транспортні витрати і передбачувану кількість викидів CO<sub>2</sub> в атмосферу. Це дасть можливість реалізувати більш складні сценарії – наприклад, порівняння різних маршрутів з точки зору екологічної ефективності.

Ще одним важливим кроком у розробці може стати додавання можливості експорту результатів роботи з калькулятором. Підтримка форматів PDF, Excel (XLSX) або JSON дозволяє користувачам зберігати та досліджувати дані зручним

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		32

способом. Це особливо корисно при використанні калькулятора в освітніх проектах, складанні корпоративних екологічних звітів або подачі заявок на гранти та екологічні програми. Інформацію можна доповнити графіками, таблицями та поясненнями, щоб підвищити її цінність для широкого кола користувачів – як для звичайної людини, так і для фахівців.

Наступним кроком у розробці мого додатку буде додавання функціоналу облікових записів користувачів та можливості зберігати дані у хмарі. Це дозволить значно розширити можливості проекту, перетворивши його з простого розрахункового інструменту в повноцінну платформу для аналізу та моніторингу вашого особистого впливу на навколишнє середовище в динаміці.

Після впровадження системи авторизації користувачі зможуть створювати власні облікові записи з можливістю зберігати історію своїх розрахунків і будувати в ній графіки прогресу і змін. Особливо це буде корисно для людей, які хочуть поліпшити свої екологічні звички і стежити за реальними змінами в них. Крім того, можна додати функцію перездачі тесту через певні проміжки часу (наприклад, раз на місяць) з автоматичним аналізом результатів і видачею рекомендацій і повідомлень.

Для реалізації технічної частини необхідно перейти від простої структури клієнта до моделі «клієнт-сервер». На стороні сервера буде використовуватися база даних (наприклад, Firebase, Supabase або PostgreSQL), в той час як клієнтська сторона буде займатися процесом авторизації (через форму або соціальні мережі) і взаємодією з API. При цьому поточна front-end структура залишиться недоторканою і легко розширюваною: етапи, форми і методи обчислень можуть бути адаптовані до нових умов без необхідності повного перепроєктування архітектури.

Використання сховища даних у хмарному середовищі дає можливості для розробки більш складних методів аналітики. Наприклад, ви можете створити інформаційну панель зі зведеними даними (для вчених, освітян та екологічних організацій), побудувати графік свого вуглецевого сліду на основі пори року або звичок і пропонувати рекомендації на основі поведінки користувачів. Маючи

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		33

достатньо інформації, стає можливим використовувати моделі штучного інтелекту для аналізу поведінки та створення персоналізованих рекомендацій.

Це також допоможе вам синхронізувати інформацію між різними пристроями та використовувати одне й те саме робоче середовище на телефоні та комп'ютері. Крім того, можна буде отримувати повідомлення про планові перевірки або зміни в розрахунках (наприклад, при зміні тарифів або цін на паливо в регіоні користувача).

Спочатку проєкт був орієнтований на аудиторію з україномовним контекстом сталого розвитку в Україні. Однак додаток з самого початку розроблений так, що є можливість підтримувати кілька мов одночасно без істотних змін в архітектурі. Для цього потрібно додати об'єкт-словник, в якому всі текстові складові інтерфейсу – заголовки, кнопки, підписи і спливаючі підказки – будуть зберігатися у вигляді ключів з відповідними перекладами на різні мови. Коли сторінка завантажується або змінює мову, дані зі словника автоматично вставляються у відповідні елементи DOM.

Технічно це можна зробити за допомогою функції JavaScript. Він прочитає обрану мову з локального сховища або параметр URL та оновить текстовий контент усіх елементів з певними атрибутами (наприклад, `data-i18n`). Такий підхід робить роботу з декількома мовами більш зручною: щоб додати нову мову, потрібно просто перекласти словник без необхідності змінювати HTML-код або логіку JavaScript.

На додаток до цього, щоб адаптуватися до аудиторії по всьому світу, необхідно також локалізувати контент: змінювати вимірювання (наприклад, кілограми → фунти), перераховувати тарифи та назви продуктів, і навіть приводити методики розрахунку викидів у відповідність до регіональних стандартів. Ви легко зможете налаштувати калькулятор для країн Європейського Союзу (EU), Сполучених Штатів Америки (USA), Канади або будь-якої іншої країни.

Це розширення відкриває нові можливості не тільки для особистого користування, а й для інтеграції в освітні та корпоративні проєкти з метою

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		34

зменшення вуглецевого сліду. Наприклад, компанія з іншої країни зможе використовувати таку ж адаптовану під свої потреби платформу і відстежувати інформацію про своїх співробітників в рамках екологічних програм.

Зі збільшенням кількості користувачів, які взаємодіють з веб-додатками через мобільні пристрої та планшети, логічним кроком для розширення функціоналу є покращення мобільної версії та додавання можливості роботи в автономному режимі. В даний час дизайн калькулятора вже адаптований з використанням медіа-запитів і гнучкої верстки. Однак в майбутньому можна перейти на використання повноцінної моделі PWA (Progressive Web App).

Додаток з прогресивним веб-підходом надає користувачам можливості взаємодії на рівні рідного рівня: додавання на головний екран пристрою, кешування інформації для офлайн-доступу та миттєве завантаження інтерфейсу при наступному запуску. Це особливо корисно для регіонів з нестабільним інтернет-з'єднанням і в тих випадках, коли тренування проводяться в місцях без надійного доступу до мережі, наприклад, на виїздах або в школах зі слабким з'єднанням.

Щоб перейти на PWA, потрібно впровадити Service worker, налаштувати маніфест додатку (manifest.json) та розробити стратегію кешування даних. Така структура дозволить використовувати калькулятор як повноцінний мобільний додаток незалежно від браузера або постійного інтернет-з'єднання. Це значно підвищить стійкість та доступність проекту та підвищить його шанси на успішне проникнення серед широкої аудиторії.

Одним з перспективних напрямків розвитку є взаємодія екологічного калькулятора з освітніми платформами. Таке партнерство дозволить використовувати калькулятор як практичний інструмент у навчальному процесі для школярів та студентів. Вони не лише зможуть дізнатися про свій вуглецевий слід, але й зможуть оцінити свій вплив на навколишнє середовище.

З практичного боку можна додати спеціальну опцію «для викладача», яка дозволить аналізувати дані по класах або групах студентів, порівнювати результати і відстежувати прогрес протягом семестру. Також є можливість

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
						35
Зм.	Арк.	№ докум.	Підпис	Дата		

розробити інструмент для збереження інформації про студентів з автоматичною генерацією PDF-звітів або сертифікатів про участь.

У майбутньому калькулятор може стати частиною національної програми екологічної освіти або брати участь у міжнародних конкурсах, спрямованих на розвиток екології у свідомості молоді. Це дозволить поліпшити репутацію проекту і розширити його вплив за межі сфери інформаційних технологій.

Архітектура та структура веб-додатку були розроблені з урахуванням майбутніх удосконалень на етапі дипломного дизайну. Завдяки модульній організації коду та гнучкості інтерфейсу, калькулятор можна легко адаптувати під нові сценарії використання, цільові аудиторії та технологічні вимоги. Будь то розширення функціоналу, перехід на клієнт-серверну архітектуру, інтеграція з освітніми платформами або адаптація до міжнародних ринків – проект готовий до майбутніх змін.

Потенціал масштабування робить цей екологічний калькулятор не тільки хорошим кейсом, а й перспективним проектом зі створення корисного продукту для суспільства та його екологічної свідомості.

## **1.3 Розробка та реалізація веб-застосунку**

### **1.3.1 Інтерфейс користувача (UI/UX структура)**

Інтерфейс калькулятора екологічного сліду був розроблений з акцентом на простоту використання, інтуїтивність і привабливий дизайн. Основна мета полягала в тому, щоб створити користувальницький досвід так, щоб будь-хто – незалежно від віку, технічної підготовки чи досвіду – міг легко пройти всі етапи калькулятора та отримати зрозумілі результати.

Інноваційним рішенням стало впровадження покрокового підходу: кожен етап представлений окремим блоком із запитаннями на певну тему – транспорт, електроенергія, газ, їжа, вода, покупки тощо.

Нижче розглянемо приклад HTML-структури одного з етапів калькулятора. Кожен блок містить заголовок, описові підписи та поля введення, згруповані логічно відповідно до тематики кроку.

```
<div class="step active" data-step="1">
```

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		36

```

<h2>Транспорт</h2>
<label for="public-transport">🚗 Скільки поїздок громадським транспортом на тиждень?</label>
<input type="number" id="public-transport" min="0" placeholder="Наприклад: 10" required>
</div>

```

У наведеному фрагменті HTML-структури використано клас `step` для позначення окремого етапу форми, який може бути активним (`.active`) або прихованим. Заголовок `<h2>` визначає тему кроку, елемент `<label>` підписує відповідне поле введення, а тег `<input type="number">` дозволяє користувачеві вводити числові значення (у даному випадку — кількість поїздок). Атрибут `placeholder` забезпечує зручну текстову підказку, що полегшує орієнтацію під час заповнення.

Завдяки поділу на тематичні блоки (`step-by-step`) користувач зосереджується лише на одній категорії за раз, не відволікається і проходить логічну послідовність з високим рівнем залучення. Це знижує бар'єр входу і створює ефект «освоєння з першого використання».

На рисунку 1.8 подано схему етапів взаємодії з калькулятором.

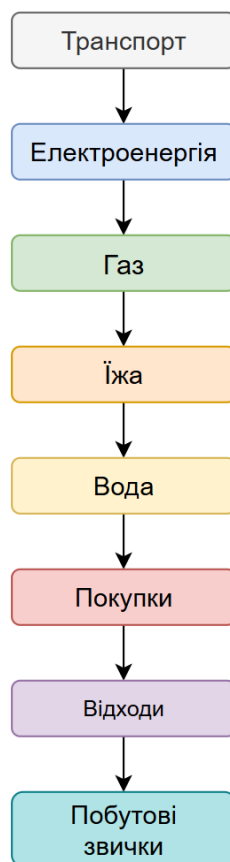


Рисунок 1.8. Схема поетапної взаємодії користувача з калькулятором

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		37

Для кожного розділу використовуються емодзі – яскраві піктограми, що символізують відповідну тему (наприклад, 🚗 для транспорту чи 🍌 для їжі). Крім цього, супровідні тексти подано у дружній, неформальній манері, що створює більш приємну атмосферу взаємодії та полегшує сприйняття інформації, особливо для молодіжної аудиторії. На рисунку 1.9 зображено як виглядає блок «Їжа».

Рисунок 1.9. Інтерфейс тематичного блоку «Їжа» у калькуляторі екологічного сліду

Кожне поле введення містить текст-підказку з прикладами значень (наприклад, «Наприклад: 1.5»), що полегшує заповнення даних і зменшує ймовірність помилок. Крім того, є індивідуальна перевірка введення даних для того, щоб виділити порожні поля тільки після натискання кнопки «Далі», щоб не створювати зайвого тиску на користувача при заповненні форми.

Інтерфейс розроблений з урахуванням адаптивної верстки – за допомогою медіазапитів CSS забезпечується коректне відображення як на великих екранах,

так і на мобільних пристроях. Всі блоки автоматично вирівнюються, а розміри шрифтів і відступів підлаштовуються під роздільну здатність екрана. Це дає можливість користуватися калькулятором в будь-яких умовах – від ноутбуків до смартфонів.

Щоб реалізувати адаптацію під різні розміри екранів, було використано медіазапити CSS. Вони дозволяють змінювати орієнтацію та поведінку блоків залежно від ширини вікна перегляду. Приклад одного з таких запитів наведено нижче:

```
@media (max-width: 600px) {  
  body {  
    padding: 0 12px 24px;  
  }  
}
```

Цей код задає відповідні відступи для тіла сторінки, якщо ширина вікна становить 600 пікселів або менше, що дозволяє коректно відображати вміст на мобільних пристроях.

На рисунку 1.10 подано приклад відображення калькулятора на смартфоні iPhone 15 Pro Max.

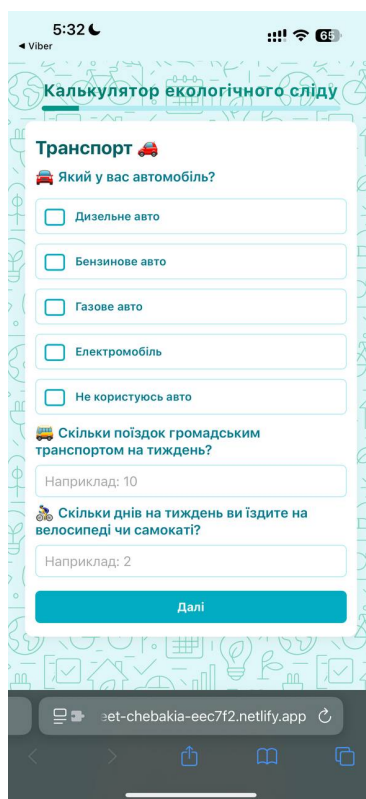


Рисунок 1.10. Приклад адаптивного відображення на мобільному пристрої

					КГ 08. 03 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		39

У верхній частині сторінки є індикатор прогресу у вигляді рядка статусу, який показує поточний прогрес користувача у заповненні форми. Реалізація цього елемента заснована на CSS і JavaScript: при переході до наступного кроку ширина індикатора збільшується, зберігаючи м'який відтінок і гармонійне поєднання з основною колірною гамою сторінки. Такий підхід не тільки створює приємне візуальне враження для користувача, а й допомагає йому контролювати процес заповнення форми.

Розміщення додаткових інформаційних розділів реалізовано у вигляді компактного блоку в нижній частині сторінки. Кожен елемент цього блоку – це інтерактивне текстове посилання, яке відкриває окремий контентний шар без оновлення всієї сторінки.

Розглянемо HTML-структуру цих кнопок.

```
<footer class="site-footer">
  <div class="footer-links">
    <a href="#" onclick="showInfo('author-ref')">Авторська довідка</a>
    <a href="#" onclick="showInfo('tech-doc')">Технічна документація</a>
    <a href="#" onclick="showInfo('resources')">Ресурсні матеріали</a>
  </div>
</footer>
```

Завдяки атрибуту `onclick` викликається функція `showInfo()`, що динамічно відображає відповідний інформаційний блок – довідку про автора, технічний опис чи корисні екологічні матеріали. Весь навігаційний блок оформлено так, щоб він не відволікав від основного функціоналу калькулятора, залишаючись водночас доступним та помітним для користувача.

Вся верстка призначеного для користувача інтерфейсу організована компактно: кроки форми, кнопки навігації, підказки та блоки з результатами розміщуються в єдиному HTML-документі. Це дозволяє легко вносити зміни в елементи, додавати нові розділи або змінювати їх порядок без необхідності переглядати основну логіку проекту.

Інтерфейс був розроблений з урахуванням потреб усіх користувачів. Тексти написані простою і зрозумілою мовою. Зручне розташування елементів на екрані, достатні відступи та адаптивний дизайн роблять використання калькулятора комфортним як для новачків, так і для досвідчених користувачів.

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		40

Отже, калькулятор враховує принципи зручності, доступності та інтуїтивної навігації. А завдяки візуальній привабливості та доброзичливому тону, він створює позитивний досвід взаємодії, мотивуючи користувачів звертати більше уваги на власний екологічний вплив.

### 1.3.2 Розробка функціоналу

При розробці основний акцент був зроблений на простоті логіки роботи і високій інтерактивності без використання сторонніх бібліотек для більшого контролю над кожним елементом і спрощення налагодження.

Ключові функціональні елементи взаємодії – це методи, які дозволяють користувачеві плавно переміщатися між етапами калькулятора. Основу цього механізму складають обробники подій, прив'язані до кнопок «Далі» та «Назад». Вони реалізовані через `addEventListener` і відповідають за логіку покрокового переходу.

```
nextBtn.addEventListener("click", () => {
  if (currentStep < steps.length - 1) {
    steps[currentStep].classList.remove("active");
    currentStep++;
    showStep(currentStep);
  }
});
```

При натисканні кнопки «Далі» перевіряється, чи не досягнуто останнього кроку. Якщо ні – поточний етап приховується шляхом видалення класу `active`, і показується наступний, якому додається цей клас. Такий підхід дозволяє формувати лінійний процес взаємодії, де користувач зосереджується лише на одному блоці за раз, що спрощує сприйняття та зменшує інформаційне навантаження.

```
prevBtn.addEventListener("click", () => {
  if (currentStep > 0) {
    steps[currentStep].classList.remove("active");
    currentStep--;
    showStep(currentStep);
  }
});
```

Кнопка «Назад» працює аналогічно, але в зворотному напрямку. Вона дає змогу повернутися до попереднього етапу, що забезпечує свободу навігації – користувач у будь-який момент може змінити введені дані або переглянути

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		41

попередню інформацію. Завдяки цьому процес заповнення стає більш гнучким та адаптивним до потреб користувача.

Перш ніж користувач зможе перейти до наступного етапу, дані на поточному кроці проходять локальну перевірку на коректність. Така покрокова валідація дозволяє уникнути введення некоректних або порожніх значень, зберігаючи простоту взаємодії на кожному етапі. Розглянемо логіку коду, який реалізовано через обробник події для кнопки «Далі»

```
nextBtn.addEventListener("click", () => {
  const currentStepEl = steps[currentStep];
  const inputs = currentStepEl.querySelectorAll("input, select");
  let isValid = true;
  for (const input of inputs) {
    if (input.closest(".hidden") || input.closest(".hidden-block")) {
      continue;
    }
    if (!input.checkValidity()) {
      input.classList.add("invalid");
      input.reportValidity();
      input.focus();
      isValid = false;
      break;
    }
  }
  if (isValid && currentStep < steps.length - 1) {
    steps[currentStep].classList.remove("active");
    currentStep++;
    showStep(currentStep);
  }
});
```

Цей код перевіряє тільки видимі поля на поточному етапі – тобто ті, що не мають класів `.hidden` або `.hidden-block`. Це дозволяє не змушувати користувача заповнювати поля, які йому не були показані (наприклад, залежно від обраного типу транспорту чи джерела енергії). Якщо якесь поле є обов'язковим і залишилось порожнім, воно підсвічується класом `invalid`, що зазвичай оформлено як червона рамка або інше візуальне підкреслення.

Після внесення правильного значення виділення зникає автоматично, що покращує досвід користувача.

Цей тип перевірки – локальний, лише для поточного кроку.

Під час натискання на кнопку «Розрахувати» виконується фінальна перевірка – тоді аналізуються всі кроки та всі поля (включно з тими, що були

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		42

приховані), щоб переконатися в повноті введених даних. Нижче наведено код, який реалізує цю перевірку.

```
form.addEventListener("submit", function(event) {
  event.preventDefault();
  let hasError = false;
  let firstInvalidInput = null;
  steps.forEach((stepEl, index) => {
    const inputs = stepEl.querySelectorAll("input, select");
    inputs.forEach(input => {
      if (input.closest(".hidden") || input.closest(".hidden-block")) {
        return;
      }
      if (input.hasAttribute("required") && !input.value.trim()) {
        input.classList.add("invalid");
        if (!firstInvalidInput) {
          firstInvalidInput = input;
          currentStep = index;
        }
        hasError = true;
      } else {
        input.classList.remove("invalid");
      }
    });
  });
  if (hasError && firstInvalidInput) {
    steps.forEach(el => el.classList.remove("active"));
    showStep(currentStep);
    setTimeout(() => {
      firstInvalidInput.scrollIntoView({ behavior: "smooth", block: "center" });
      firstInvalidInput.focus();
    }, 100);
    return;
  }
  showResult();
});
```

Цей фрагмент скрипта реалізує фінальну валідацію форми перед відображенням результатів розрахунку. Його головне завдання – забезпечити цілісність та коректність усіх введених даних, незалежно від порядку заповнення блоків чи можливих пропусків.

Насамперед, обробник події submit блокує стандартну поведінку браузера (event.preventDefault()), що унеможливорює передчасне завершення процесу без перевірки даних. Далі скрипт поетапно перебирає всі кроки форми, аналізуючи кожне текстове поле та випадаючий список. У процесі обробки ігноруються поля, приховані за допомогою класів .hidden та .hidden-block, оскільки вони не повинні брати участі у валідації.

Якщо хоча б одне з обов'язкових полів (із атрибутом required) залишено порожнім, воно виділяється візуально за допомогою класу invalid. Крім того,

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		43

скрипт запам'ятовує перше некоректне поле, щоб автоматично перемістити користувача до відповідного етапу форми. Після цього здійснюється плавна прокрутка до поля з помилкою та фокусування на ньому, що сприяє зручності виправлення.

Лише у випадку, якщо всі обов'язкові поля заповнено коректно, викликається функція `showResult()`, яка відповідає за генерацію підсумкового результату.

Реалізований підхід комбінує локальну покрокову перевірку даних (на етапі переходу кнопкою «Далі») із глобальною перевіркою перед завершенням, що дозволяє досягти високої надійності введених даних та позитивного досвіду користувача. керування поведінкою кнопками «Про автора», «Про застосунок» та «Базу знань» було реалізовано функції `showInfo()` та `hideInfo()`, які відповідають за динамічне відображення вмісту без перезавантаження сторінки. Розглянем на рисунку 1.11 розкритий блок однієї з кнопок.

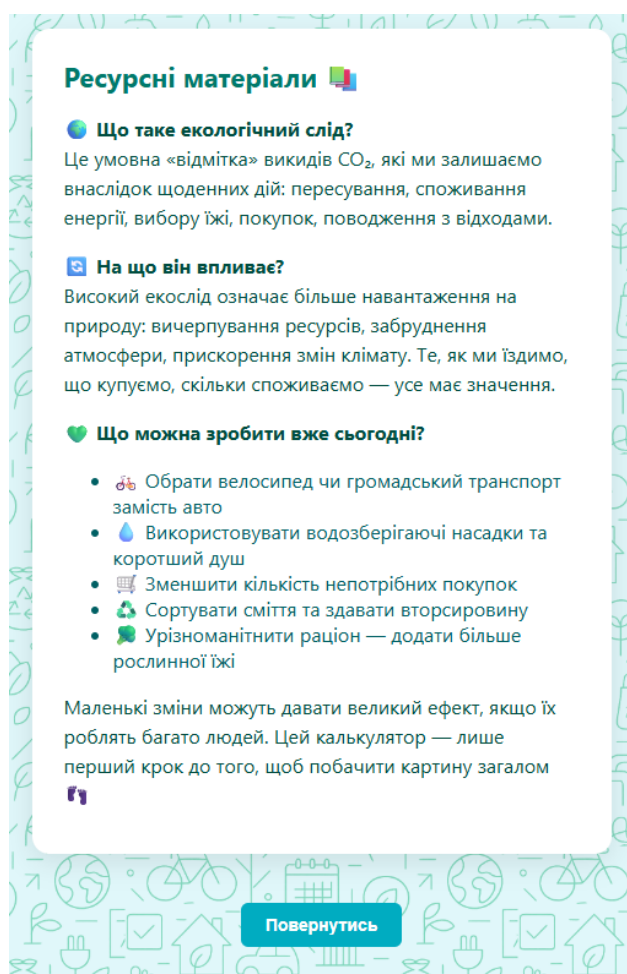


Рисунок 1.11. Блок «Ресурсні матеріали»

					КГ 08. 03 001. 00 ДП ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		44

Розглянемо код `showInfo()` та `hideInfo()`.

```
function showInfo(type) {
  const infoBlock = document.getElementById("info-block");
  let html = "";
  if (type === "author-ref") {
    html = `

## Авторська довідка

<p>...</p>`;
  } else if (type === "tech-doc") {
    html = `

## Технічна документація

<p>...</p>`;
  } else if (type === "resources") {
    html = `

## Ресурсні матеріали

<p>...</p>`;
  }
  infoBlock.innerHTML = html;
  infoBlock.classList.remove("hidden");
  document.getElementById("footprint-form").classList.add("hidden");
  document.getElementById("back-button-container").classList.remove("hidden");
}
```

Повернення до головної форми здійснюється натисканням кнопки «Повернутись», що активує функцію:

```
function hideInfo() {
  document.getElementById("info-block").classList.add("hidden");
  document.getElementById("footprint-form").classList.remove("hidden");
  document.getElementById("back-button-container").classList.add("hidden");
}
```

Узагальнена схема маршрутизації елементів сайту з урахуванням усіх можливих переходів та дій користувача, включно з інформаційними секціями та кнопками керування, наведена нижче на рисунку 1.12.

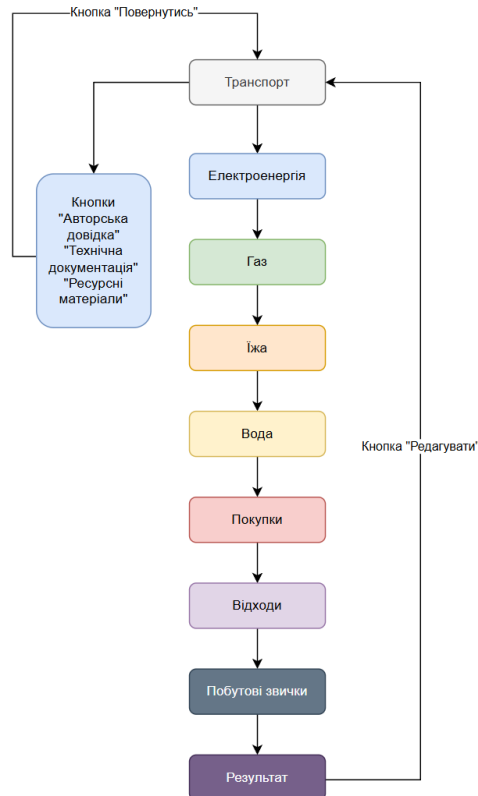


Рисунок 1.12. Схема маршрутизації інтерфейсу користувача калькулятора

Особлива увага приділяється гнучкості: при необхідності нові кроки, поля або блоки можна легко додати без проблем. Система побудована так, що будь-які додаткові розділи будуть оброблені коректно, не змінюючи загальну структуру тексту. Це особливо важливо для майбутнього розширення проекту – як в рамках освітньої сфери, так і при адаптації до інших ринків або категорій.

Додаток створює зручний і надійний користувальницький досвід, щоб в майбутньому ви могли легко розширювати і адаптувати його функціонал.

### 1.3.3 Візуалізація результатів

Завершальним етапом взаємодії користувача з калькулятором екологічного сліду є відображення результатів після натискання кнопки «Розрахувати». Після проходження усіх етапів форми запускається функція `showResult()`, яка здійснює підрахунок загального сліду та його розподіл за категоріями. Кожен з блоків калькулятора — транспорт, електроенергія, газ, вода, харчування тощо — має власну формулу для обчислення відповідного внеску у загальний обсяг викидів CO<sub>2</sub>. Отримані значення підсумовуються, формуються категорії, які потім відображаються на діаграмі та слугують основою для вибору екологічної поради.

Щоб краще зрозуміти логіку цього процесу, на рисунку 1.13 представлено узагальнену схему обчислення.

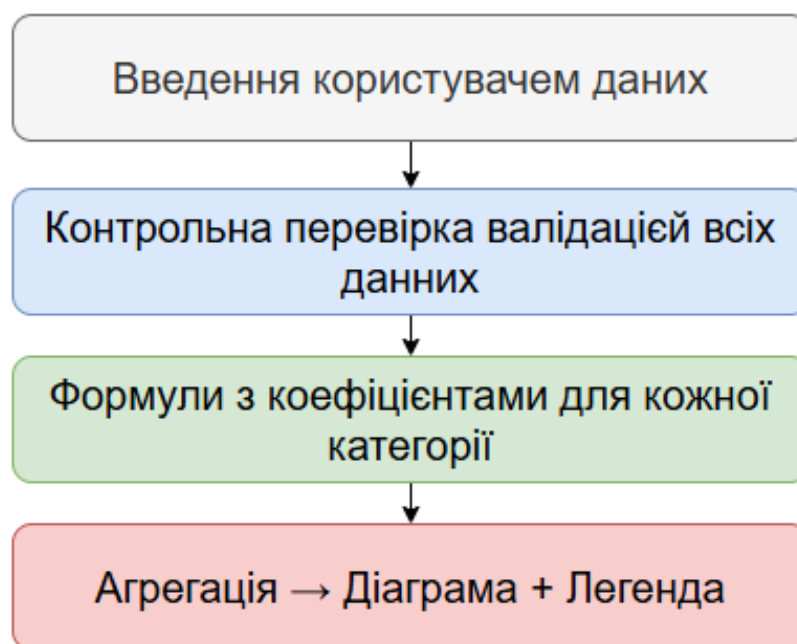


Рисунок 1.13. Алгоритм побудови фінального результату у калькуляторі CO<sub>2</sub>

Для візуального представлення результатів використовується діаграма типу «pie chart», створена на основі бібліотеки Chart.js. Всі обчислення відбуваються безпосередньо на клієнтській стороні у функції showResult(). Значення агрегуються за наступними категоріями: транспорт, електроенергія, газ, їжа, вода, покупки, відходи, побут. Нижче наведено фрагмент коду, який відповідає за побудову цієї діаграми:

```
const ctx = document.getElementById("chart").getContext("2d");
window.myChart = new Chart(ctx, {
  type: "pie",
  data: {
    labels: sortedLabels,
    datasets: [{ data: sortedValues, backgroundColor: sortedColors }]
  },
  options: {
    responsive: false,
    events: [],
    animation: {
      animateRotate: true,
      animateScale: true,
      duration: 1200,
      delay: 400
    },
    rotation: -Math.PI / 2,
    plugins: {
      tooltip: { enabled: false },
      legend: { display: false },
      datalabels: {
        color: "#fff",
        font: { weight: "bold", size: 14 },
        formatter: function(value, context) {
          const ds = context.chart.data.datasets[0].data;
          const tot = ds.reduce((a, b) => a + b, 0);
          return `$$$((value / tot) * 100).toFixed(1))%`;
        }
      }
    }
  }
});
```

Діаграма створюється динамічно на основі відсортованого набору даних. Спочатку категорії в масиві впорядковуються в порядку спадання, а потім візуалізуються у вигляді кругової діаграми від початку найбільш значущого відрізка. Найбільш значні джерела викидів виділяються у верхній частині діаграми, щоб привернути увагу. На основі цих даних створюється масив dataset, який потім сортується за величиною викидів (від більшого до меншого). Це дозволяє користувачу візуально побачити, яка сфера побуту має найбільший вплив на екологію.

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		47

Після сортування даних генерується легенда, яка відображається ліворуч від діаграми. Для кожної категорії зазначається назва, обсяг викидів у кілограмах та відсоткове співвідношення. Легенда з'являється з анімацією, що покращує сприйняття інформації.

Кожній категорії заздалегідь присвоюється свій унікальний колір:

- Транспорт – помаранчевий (#FFCBA4),
- Elekrika – фіолетовий (#CAB8FF),
- Газ – жовтий (#FFEVB4),
- Їжа – світло-зелений (#C1FBA4),
- Вода – блакитний (#A0E7E5),
- Шопінг – рожевий (#FFB5C2),
- Відходи – серо-зелёный (#B2C9A0),
- Побут жовто-бежевий (#fffd9e).

Ці відтінки підібрані так, щоб забезпечити чітке розмежування між різними сегментами і гармонійно вписатися в загальну колірну гамму ділянки. Кожен сегмент діаграми також супроводжується легендою збоку графіка, щоб користувачі могли швидко визначити відповідність кольору та категорії.

Такі кольори роблять діаграму інтуїтивно зрозумілою та візуально привабливою. Вони допомагають швидко орієнтуватися в даних без зайвих зусиль. Гармонійна палітра сприяє кращому сприйняттю інформації та знижує втому очей. Користувачі можуть легко порівнювати категорії та аналізувати їхній вплив. Завдяки чіткій легенді інтерпретація даних стає максимально зручною..

Під графіком та легендою буде показано фінальне повідомлення з наступною інформацією:

- Загальний підсумок (наприклад, "Ваш орієнтовний екологічний слід: 130.46 кг CO<sub>2</sub>"),
- Екологічна порада (в залежності від суми балів або домінуючої категорії).

Це дозволяє людині відчути завершеність процесу і отримати точні дані для розвитку змін у своєму житті. Нижче на рисунку 1.14 наведено приклад фінальної кругової діаграми, яка відображає розподіл вуглецевого сліду між категоріями.

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		48



Першочерговим завданням було тестування калькулятора в різних браузерах: Google Chrome, Mozilla Firefox, Microsoft Edge і Safari. Особлива увага була приділена правильному відображенню форми, працездатності кнопок, анімації переходів, точності прогрес-бару і сумісності з JavaScript і стилями. Для максимального охоплення аудиторії враховуються як сучасні версії браузерів, так і старі пристрої.

Перед цим він був розміщений на платформі Netlify для подальшого тестування в публічному середовищі з метою перевірки коректності роботи завантаження файлів, підвищення швидкості інтерфейсу і стабільності з'єднання

Тестування адаптивного дизайну веб-сторінки також було важливим кроком. Тест проводився на самих різних пристроях – від великих моніторів до планшетів і смартфонів з різними екранами. Це допомогло виявити та виправити візуальні дефекти: обрізаний текст, неправильну ширину кнопок, зміщення блоків на маленьких екранах. Для цього були змінені CSS-медіа-запити і логіка розміщення компонентів.

Під час тестування функціоналу акцент був зроблений на стабільності багатоетапної форми: перевірявся перехід між етапами процесу заповнення даних та функціонування кнопок «Назад», «Далі» та «Розрахувати», коректність відображення та приховування розділів за допомогою класів `.active.`, `.hidden.`, а також збереження цілісності локальних даних у `localStorage`.

Ручне тестування було проведено на поширених випадках використання користувачами:

- Рух вперед і назад без необхідності вводити інформацію.
- Розрахунок у разі неповного заповнення форми;
- Перезапустить програму після закриття вкладки.

На цьому етапі було виявлено кілька недоліків: відсутність порожніх заповнених обов'язкових полів, випадкове зникнення кнопки «Назад» при поверненні на попередню сторінку, неочевидна функціональність блоку з підказками і один випадок помилкового округлення результату розрахунку. Всі знайдені проблеми були усунені вручну після аналізу відгуків тестувальника і

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		50

переробки відповідного скрипта.

Для фінальної перевірки проект був завантажений на GitHub, також було перевірено доступність сайту за прямим посиланням, щоб продемонструвати проект вчителям та членам комітету без необхідності авторизації чи додавання нових сервірів.

Нижче на рисунку 1.15 наведено приклад застосунок у браузері.

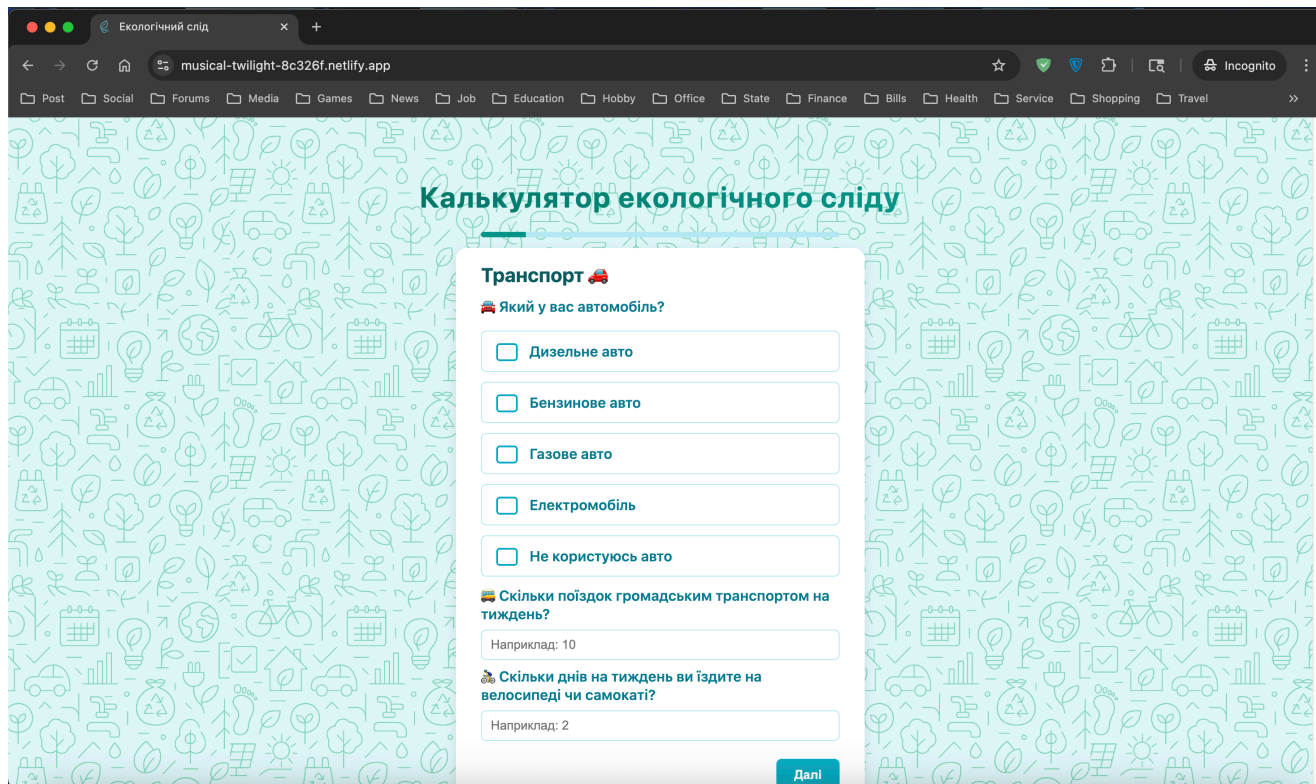


Рисунок 1.15. Веб-застосунок у браузері

					<i>КГ 08. 03 001. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		51

## 2 ЕКОНОМІЧНИЙ РОЗДІЛ

### 2.1 Резюме

У дипломному проєкті розроблено експертний веб-застосунок для розрахунку екологічних слідів, який дозволяє аналізувати вплив споживчих звичок на навколишнє середовище. Ефективність рішення оцінюється за критеріями точності розрахунків, інтерактивності візуалізації даних та корисності для користувачів.

### 2.2 Визначення трудомісткості розробки ПЗ

Тривалість розробки програмного продукту залежить від його обсягу, складності, кваліфікації розробників і встановлених ринком термінів. Метод структурної аналогії дозволяє оцінити обсяг у тисячах умовних машинних команд на основі подібного програмного забезпечення.

Табл. 2.1 містить аналоги ПЗ з подібними функціями; обраний варіант виділено сірим.

Таблиця 2.1. Каталог аналогів

Найменування ПП	Обсяг функції ПП – $V_0$ , умовних. машинних командах
1. ПП автоматизації засобів по каталогу	680 – 7000
2. ПП автоматизованих розрахунків	1300 – 8600
3. ПП введення інформації	1060 – 5750

Після вибору аналога з обсягом  $V_0$  (умовні машинні команди), трудомісткість визначається за табл. 2.2.

Таблиця 2.2. Обсяг ПП

Обсяг ПП, тис.умов.машинних команд	Норма часу, люд/год
1.00	229
2.00	244
3.00	262
4.00	283
5.00	306
6.00	330
7.00	357

Обсяг ПП, тис.умов.машинних команд	Норма часу, люд/год
8.00	385
9.00	414
10.00	445

На основі отриманого значення за довідником визначають укрупнену норму часу, скориговану коефіцієнтом  $K_k = 0,7-0,8$  для умов розробки на комп'ютері:

$$T_{ap} = 306 \times 0,8 = 244,8 \text{ (люд/годин)} \quad (2.1)$$

Трудомісткість визначається для кожного етапу окремо, з урахуванням складності, новизни та використання стандартних модулів, за відповідними формулами:

$$T_{T3} = T^a \rho \times L_1 \times K_H \quad (2.2)$$

$$T_{TP} = T^a \rho \times L_2 \times K_H \quad (2.3)$$

$$T_{PP} = T^a \rho \times L_3 \times K_H \times K_T \quad (2.4)$$

Для розрахунку використовуються такі коефіцієнти:

- $L_i$  – частка  $i$ -го етапу (табл. 2.3);
- $K_H$  – коефіцієнт новизни (табл. 2.4);
- $K_T$  – коефіцієнт використання типових програм (табл. 2.5).

Наш варіант виділено сірим.

Таблиця 2.3. Питомі коефіцієнти трудомісткості стадії у загальній трудомісткості розробки ПП

Код стадії	Ступінь новизни		
	А	Б	В
ТЗ ( $L_1$ )	0,15	0,12	0,12
ТП ( $L_2$ )	0,16	0,15	0,11
РП ( $L_3$ )	0,55	0,58	0,61

Таблиця 2.4. Значення коефіцієнта новизни

Код ступеня новизни	Ступінь новизни	Значення $K_H$
А	Принципово новий ПП	1,75 – 1,2
Б	ПП – розвиток визначеного параметричного ряду	1,0 – 0,8

Код ступеня новизни	Ступінь новизни	Значення $K_n$
В	ПП, що має аналог	0,7

Таблиця 2.5. Значення коефіцієнта використання типових програм

Ступінь охоплення реалізованих функцій розробленого ПП типовими програмами, %	Значення $K_T$
60 і вище	0,6
40-60	0,7
20-40	0,8
До 20	0,9

Тепер розраховуємо трудомісткість для всіх етапів і зводимо у табл. 2.6:

Трудомісткість технічного завдання:

$$T_{tz} = Ta * L_1 * K_n = 244,8 * 0,12 * 0,8 = 23,50 \text{ (люд/годин)} \quad (2.2)$$

Трудомісткість розробки технічного проекту:

$$T_{tp} = Ta * L_2 * K_n = 244,8 * 0,15 * 0,8 = 29,37 \text{ (люд/годин)} \quad (2.3)$$

Трудомісткість розробки робочого проекту:

$$T_{rp} = Ta * L_3 * K_n * K_m = 244,8 * 0,58 * 0,8 * 0,7 = 79,51 \text{ (люд/годин)} \quad (2.4)$$

Для розрахунків визначили обсяг документації по етапах:

- технічне завдання  $N_{tz}=2$  (стор);
- розробка ТП  $N_{tp}=43$  (стор);
- розробка робочого проекту  $N_{rp}=9$  (стор);
- пояснювальна записка відповідно  $N_{pz}=15$  (стор).

Таблиця 2.6. Розрахунок трудомісткості ПП

Найменування етапів	Розрахунок, годин		
	1.ТЗ	$T_{РТЗ}=23,50$	$T_{кк}=0,7*N_{ТЗ}= 0,7*2=1,4$
2.Розробка ТП	$T_{РТП}=29,37$	$T_{кк}=0,7*N_{ТП}=0,7*43=30,1$	$T_{нк}=0,15*N_{ТП}=0,15*43=6,45$
3.Розробка РП	$T_{Ррп}= 79,51$	$T_{кк}=0,7*N_{рп}=0,7*9=6,3$	$T_{нк}=0,15*N_{рп}=0,15*9=1,35$
4.Розробка ПЗ	$T_{Пз}=1,5*N_{Пз}=1,5*15=22,5$	$T_{кк}=0,7*N_{Тз}=0,7*15=10,5$	$T_{нк}=0,15*N_{Пз}=0,15*15=2,25$

Усього, в т.ч.:	213,53		
- на розробку	Тр=154,88		
- контроль керівника		Ткк=48,3	
- нормоконтроль			Тнк=10,35

### 2.3 Розрахунок ціни програмного продукту

Розраховуємо основну зарплату виконавців, матеріальні та загальні витрати на розробку ПП. Зарплата наведена в табл. 2.7. З 1 квітня 2024 мінімальна місячна зарплата – 8000 грн, погодинна ставка – 46 грн (згідно зі ст. 8 Закону про Держбюджет України).

Таблиця 2.7. Розрахунок основної заробітної плати виконавців

Найменування робіт	Трудомісткість робіт, години	Погодинна тарифна ставка, грн.	Розрахунок, грн.
1.Розробка ПП	154,88	55,35	8573,48
2.Контроль керівника	48,3	120,28	5810,00
3.Нормоконтроль	10,35	120,28	1245,00
Усього	-	-	З <sub>о</sub> = 15628,48

Розрахуємо матеріальні витрати на розробку ПП та наведемо їх у табл. 2.8.

Таблиця 2.8. Розрахунок матеріальних витрат на розробку

Найменування матеріальних витрат	Тип, модель	Кількість	Ціна одиниці, грн.	Вартість, грн.
Папір	Лист А4	70	4.0	280,0
Разом	-	-	-	$V_{M1}=280,0$
Транспортно – заготівельні Витрати (10%)				$V_{тр-з} = 0.1 * V_{M1} = 0,1*280 = 28,00$
Усього				$V_M = V_{M1} + V_{тр-з} = 308,00$

За отриманими даними складена калькуляція планової собівартості ПП, наведена в табл. 2.9.

					<b>КГ 08. 03 002. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		55

Таблиця 2.9. Розрахунок статей витрат планової собівартості

Стаття витрат	Значення, грн.	Формула розрахунку
1. Матеріали	308,00	$V_M$ (див. табл. 2.8)
2. Основна заробітна плата	15628,48	$Z_o$ (див. табл. 2.7)
3. Додаткова заробітна плата	1562,84	$Z_d = 0.1 * Z_o = 15628,48 * 0,1$
4. Відрахування до єдиного фонду соціального внеску	3864,37	$V_{с.с.в.} = 0.22 * (Z_o + Z_d) = 0,22 * (15628,48 + 1562,84)$
5. Накладні витрати	6251,39	$V_{нак.} = 0.4 * Z_o = 0.4 * 15628,48$
6. Повна собівартість	27681,08	$C_{пов} = V_M + Z_o + Z_d + V_{с.с.в.} + V_{нак.} = 308,00 + 15628,48 + 1562,84 + 3864,37 + 6251,39$

Розмір прибутку розраховується за формулою:

$$P = (C_n * P) / 100 = (27681,08 * 10) / 100 = 2768,11 \text{ грн.} \quad (2.5)$$

Де  $p$  – плановий рівень рентабельності (10-15%).

Оптова ціна розраховується за формулою:

$$C_o = C_n + P = 27681,08 + 2768,11 = 30449,19 \text{ грн.} \quad (2.6)$$

За отриманими даними, ціна реалізації ПП за формулою становить:

$$C_p = C_o + ПДВ = 30449,19 + 30449,19 * 0.2 = 36539,03 \text{ грн.} \quad (2.7)$$

## **3 РОЗДІЛ ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ**

### **3.1 Основні положення**

Однією з ключових рис сучасного суспільного розвитку є стрімке розширення сфер людської діяльності, у яких застосовуються інформаційні технології. Активне використання персональних комп'ютерів загострює питання збереження індивідуального та суспільного здоров'я, що потребує вдосконалення наявних підходів і створення нових методів організації робочих місць, а також проведення профілактичних заходів для запобігання негативному впливу ПК на здоров'я користувачів. В розділі охорони праці дипломного проєкту розглядається питання безпеки праці веб-розробника.

### **3.2 Аналіз умов праці й забезпечення безпеки при виконання основних видів робіт на об'єкті дипломного проєктування**

Оператори та програмісти зазнають впливу низки небезпечних і шкідливих виробничих чинників фізичного характеру, зокрема підвищеного рівня шуму, високої температури навколишнього середовища, недостатнього або відсутнього освітлення робочого місця, дії електричного струму, статичної електрики та інших.

#### **3.3.1 Гігієнічні вимоги до виробничого середовища**

Для забезпечення безпечної та ефективної роботи програміста необхідно організувати робоче місце відповідно до вимог чинних нормативних і законодавчих документів.

#### **3.3.2 Вимоги до приміщення експлуатації ПК**

Розміщення робочих місць із персональними комп'ютерами у підвальних або цокольних приміщеннях не допускається. Мінімумально допустима площа на одне робоче місце повинна становити не менше 6,0 м<sup>2</sup>, а об'єм — щонайменше 20,0 м<sup>3</sup>.

Об'ємно-планувальні характеристики будівель і приміщень для роботи за ПК мають відповідати вимогам ДСанПіН 3.3.2.007–98. Освітлення в таких

					<i>КГ 08. 03 003. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		57

приміщеннях повинно включати як природне, так і штучне освітлення згідно з ДБН В.2.5-28-2006. Природне світло має надходити через вікна, орієнтовані переважно на північ або північний схід, при цьому коефіцієнт природної освітленості (КПО) не повинен бути нижчим за 1,5%.

Робочі приміщення повинні бути оснащені шафами для зберігання документації, магнітних носіїв, а також полицями, тумбами, стелажми — з урахуванням норм площі. Крім того, у приміщеннях із ПК обов'язково мають бути аптечки першої медичної допомоги..

### 3.3.3 Гігієнічні вимоги до параметрів повітря приміщень із ПК

У приміщеннях, де розміщені робочі місця з персональними комп'ютерами, необхідно дотримуватись оптимальних показників мікроклімату — температури, відносної вологості та швидкості руху повітря — згідно з вимогами ДСН 3.3.6.042-99 «Державні санітарні норми мікроклімату виробничих приміщень».

У таблиці 3.1 наведено норми мікроклімату для приміщень.

Таблиця 3.1. Норми мікроклімату для приміщень

Пора року	Категорія робіт	Температура повітря, °С, не більше	Відносна вологість повітря %	Швидкість руху повітря, м/с
Холодна	Легка-1а	22-24	40-60	0,1
	Легка-1б	21-23	40-60	0,1
Тепла	Легка-1а	23-25	40-60	0,1
	Легка-1б	22-24	40-60	0,1

Рівні позитивних і негативних іонів у повітрі приміщень з ВДТ мають відповідати санітарно-гігієнічним нормам № 2152-80.

У таблиці 3.2 наведено санітарно-гігієнічні нормам № 2152-80.

Таблиця 3.2. Санітарно-гігієнічні нормам № 2152-80

Рівні	Число іонів в 1 см <sup>3</sup> повітря	Число іонів в 1 см <sup>3</sup> повітря
	n+	n-
Мінімально необхідні	400	600
Оптимальні	1500-3000	3000-5000
Максимально допустимі	50000	50000

### 3.3.4 Виробниче освітлення

У приміщеннях з робочими місцями, обладнаними персональними комп'ютерами, штучне освітлення повинно бути організоване за принципом загального рівномірного освітлення. Якщо робота здебільшого пов'язана з обробкою паперових документів, дозволяється використання комбінованої системи освітлення, яка передбачає встановлення локальних світильників у доповнення до загального освітлення.

Освітленість робочої зони на столі, де розміщені документи, повинна складати 300–500 лк. Якщо таких показників неможливо досягти за допомогою загального освітлення, допускається застосування місцевих джерел світла, які слід розміщувати так, щоб уникати появи бликів на екрані. При цьому яскравість освітлення екрана не повинна перевищувати 300 лк. Основними джерелами світла рекомендується використовувати люмінесцентні лампи типу ЛБ.

### 3.3.5 Організація робочих місць із ПК

Обладнання й організація робочого місця програміста мають забезпечувати відповідність конструкції всіх елементів робочого місця та їх взаємного розташування ергономічним вимогам, з урахуванням особливостей трудової діяльності.

Конструкція робочого місця користувача ПК має забезпечити підтримання оптимальної робочої пози..

При розміщенні робочих столів з ПК слід дотримуватись таких відстаней: між бічними поверхнями БДТ – 1,2 м; від тильної поверхні одного ПК до екрана іншого – 2,5 м.

Екран ПК має розташовуватися на оптимальній відстані тещ очей користувача, що становить 600...700 мм, але не ближче ніж за 600 мм з урахуванням розміру літерно-цифрових знаків і символів.

У конструкції клавіатури має передбачатися опорний пристрій (виготовлений із матеріалу з високим коефіцієнтом тертя, що перешкоджає мимовільному її зсуву), який дає змогу змінювати кут нахилу поверхні клавіатури в межах 5... 15°.

					<i>КГ 08. 03 003. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		59

### 3.3.6 Вимоги до режимів праці та відпочинку при роботі з ПК

Для збереження здоров'я працюючих, запобігання професійним захворюванням і підтримки працездатності при організації праці, пов'язаної з використанням ПК передбачаються внутрішньо змінні регламентовані перерви для відпочинку.

Впродовж робочої зміни мають передбачатися:

- перерви для відпочинку і вживання їжі (обідні перерви);
- перерви для відпочинку й особистих потреб (згідно з трудовими нормами);
- додаткові перерви, що вводяться для окремих професій з урахуванням особливостей трудової діяльності для зниження нервово-емоційного напруження, втомлення зорового аналізатора, поліпшення мозкового кровообігу, запобігання втомі.

Працюючі з ПК підлягають обов'язковим медичним оглядам: попереднім – при влаштуванні на роботу і періодичним – протягом трудової діяльності, відповідно до наказу МЗ України N45 від 31.03.94 р.

### 3.4 Пожежна безпека

До первинних засобів пожежогасіння належать вогнегасники (вуглекислотні та порошкові), а також пожежний інвентар: протипожежні покривала з негорючих матеріалів, ящики з піском, бочки з водою тощо. Ці засоби призначені для оперативної ліквідації загорянь на ранніх стадіях їх виникнення силами персоналу до прибуття пожежно-рятувальних підрозділів.

Зазвичай інвентар та необхідні інструменти розміщуються на спеціально облаштованих пожежних щитах. Один такий щит має встановлюватися на кожні 5000 м<sup>2</sup> території об'єкта. Для полегшення орієнтування на території обов'язково мають бути розміщені інформаційні знаки, що вказують на місцезнаходження щитів чи вогнегасників.

У приміщеннях, де використовуються персональні комп'ютери, встановлюють вуглекислотні вогнегасники з урахуванням допустимої концентрації вогнегасної речовини в повітрі. Такі вогнегасники вирізняються

					<b>КГ 08. 03 003. 00 ДП ПЗ</b>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		60

високою ефективністю та швидкою дією.

Розрахунок кількості засобів пожежогасіння проводиться окремо для кожного приміщення, поверху чи відкритої установки. Якщо в одному приміщенні розміщено декілька виробництв з різними рівнями пожежної небезпеки без протипожежних перегородок між ними, то все приміщення забезпечується засобами пожежогасіння відповідно до вимог для найнебезпечнішого з виробництв.

На рисунку 3.1 зображено розміщення пожежного інвентарю.



Рисунок 3.1. Розміщення пожежного інвентарю

На рисунку 3.2 зображено засоби пожежогасіння.



Рисунок 3.2. Розміщення пожежного інвентарю

					<i>КГ 08. 03 003. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		61

## ВИСНОВКИ

Дипломний проєкт був спрямований на створення сучасного інтерактивного веб застосунку для розрахунку екологічного сліду людини, що дає змогу легко і доступно оцінити вплив щоденних дій користувача на навколишнє середовище. Актуальність цього завдання визначається глобальними екологічними викликами сучасності: зростанням рівня парникових газів у атмосфері, виснаженням природних ресурсів та негативними змінами клімату, що загрожують стабільності екосистем та якості життя людей.

Проведений аналіз існуючих рішень для обчислення екологічного сліду виявив, що більшість з них або недостатньо зручні й зрозумілі для широкої аудиторії, або не адаптовані до українських реалій. Новий застосунок враховує ці проблеми, пропонуючи інтуїтивний інтерфейс, адаптивний дизайн для мобільних пристроїв, використання точних коефіцієнтів для України, а також персоналізовані рекомендації, які допомагають зменшити негативний вплив на природу.

З технічної сторони проєкт базується на сучасних веб технологіях: HTML5, CSS3 та JavaScript. Такий технологічний стек забезпечує високу швидкість роботи, легкість у масштабуванні й повну автономність функціонування на стороні клієнта, що гарантує користувачам конфіденційність і безпеку їхніх даних.

Крім того, було враховано перспективи подальшого розвитку проєкту, зокрема можливість інтеграції зовнішніх джерел даних через API, створення мобільного додатку у форматі PWA та реалізацію функцій авторизації для більш персоналізованого використання. Такий комплексний підхід дозволяє значно розширити функціонал застосунку і зробити його не лише просвітницьким, але й практичним інструментом для регулярного моніторингу та аналізу екологічного впливу в динаміці.

Створений калькулятор має значні перспективи для впровадження в освітніх програмах, корпоративних екологічних ініціативах, а також може слугувати основою для майбутніх інформаційних кампаній, спрямованих на формування екологічної культури та сталого розвитку в українському суспільстві.

					<i>КГ 08. 03 000. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		62

## ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1. Мороз, О. В. Основи веб-програмування: HTML, CSS, JavaScript, Angular. – К. : Академвидав, 2021. – 312 с.
2. Коваль, П. М. Сучасна веб-розробка: від HTML до Angular. – Львів : ЛНУ імені Івана Франка, 2020. – 276 с.
3. Krill M. Angular Development with TypeScript. – Manning Publications, 2021. – 400 p.
4. Fain Y., Moiseev A. "Modern Web Development with Angular". – Addison-Wesley, 2020. – 416 p.
5. Leron P. "Efficient Frontend Development with Angular and TypeScript". – Apress, 2022. – 320 p.
6. Витяг з «Стандарти якості електропостачання та надання компенсацій» з сайту національної комісії, яка здійснює державне регулювання у сферах енергетики та комунальних послуг України (НКРЕКУ). [Веб-сайт] URL: <https://www.nerc.gov.ua/sferi-diyalnosti/elektroenergiya/yakist-elektropostachannya/standarti-yakosti-elektropostachannya-ta-nadannya-kompensacij>
7. Про відповідність Енергетичної стратегії України на період до 2030 року сучасним викликам і загрозам у сфері енергетичної безпеки. [Веб-сайт]. URL: <https://niss.gov.ua/doslidzhennya/nacionalna-bezpeka/pro-vidpovidnist-energetichnoi-strategii-ukraini-na-period-do-2030>
8. Офіцій портал YASNO. [Веб-сайт] URL: <https://yasno.com.ua/>
9. Офіцій портал DTEK Одеса. [Веб-сайт] URL: <https://www.dtek-oem.com.ua/ua>

					<i>КГ 08. 03 000. 00 ДП ПЗ</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		63

# ДОДАТОК А. Фрагмент JavaScript коду розрахунків

## екологічного сліду

```
const form = document.getElementById("footprint-form");
const steps = document.querySelectorAll(".step");
const nextBtn = document.getElementById("next-button");
const prevBtn = document.getElementById("prev-button");
const submitBtn = document.getElementById("submit-button");

let currentStep = 0;

window.addEventListener("DOMContentLoaded", () => {
  document.querySelectorAll("input, select").forEach(el => {
    const saved = localStorage.getItem(el.id);
    if (saved !== null) el.value = saved;
  });
});

document.querySelectorAll("input, select").forEach(el => {
  el.addEventListener("input", () => {
    localStorage.setItem(el.id, el.value);
  });
});

["car-diesel", "car-petrol", "car-gas", "car-electric", "car-none"].forEach(id => {
  const el = document.getElementById(id);
  el.addEventListener("change", () => {
    if (id === "car-none" && el.checked) {
      ["car-diesel", "car-petrol", "car-gas", "car-electric"]
        .forEach(other => document.getElementById(other).checked = false);
    } else if (id !== "car-none" && el.checked) {
      document.getElementById("car-none").checked = false;
    }
    updateTransportVisibility();
  });
});

function updateTransportVisibility() {
  const diesel = document.getElementById("car-diesel").checked;
  const petrol = document.getElementById("car-petrol").checked;
  const gas = document.getElementById("car-gas").checked;
  const electric = document.getElementById("car-electric").checked;

  const fuelBlock = document.getElementById("fuel-block");
  if (fuelBlock) {
    fuelBlock.classList.toggle("hidden", !(diesel || petrol));
    fuelBlock.classList.toggle("show", (diesel || petrol));
  }

  const chargeBlock = document.getElementById("charge-electric-block");
  if (chargeBlock) {
    chargeBlock.classList.toggle("show", electric);
    chargeBlock.classList.toggle("hidden-block", !electric);
  }

  const gasAutoBlock = document.getElementById("gas-auto-block");
  if (gasAutoBlock) {
    gasAutoBlock.classList.toggle("show", gas);
    gasAutoBlock.classList.toggle("hidden-block", !gas);
  }
}
```

```

const carKmBlock = document.getElementById("car-km-block");
if (carKmBlock) {
  const anyCar = diesel || petrol || gas || electric;
  carKmBlock.classList.toggle("hidden", !anyCar);
  carKmBlock.classList.toggle("show", anyCar);
}
}

const showIfYes = (selectId, blockId) => {
  const selectEl = document.getElementById(selectId);
  const block = document.getElementById(blockId);

  if (selectEl && block) {
    block.classList.add("animated-block", "hidden");

    const toggle = () => {
      const shouldShow = selectEl.value === "yes";
      block.classList.toggle("hidden", !shouldShow);
      block.classList.toggle("show", shouldShow);
    };

    selectEl.addEventListener("change", toggle);
    toggle();
  }
};

showIfYes("use-electric-cooking", "electric-cooking-block");
showIfYes("use-gas-cooking", "gas-cooking-block");
showIfYes("use-dishwasher", "dishwasher-frequency-block");

const backBtn = document.getElementById("back-button");
if (backBtn) {
  backBtn.addEventListener("click", hideInfo);
}

showStep(currentStep);

nextBtn.addEventListener("click", () => {
  const currentStepEl = steps[currentStep];
  const inputs = currentStepEl.querySelectorAll("input, select");
  let isValid = true;

  for (const input of inputs) {
    if (input.closest(".hidden") || input.closest(".hidden-block")) {
      continue;
    }

    if (!input.checkValidity()) {
      input.classList.add("invalid");
      input.reportValidity();
      input.focus();
      isValid = false;
      break;
    }
  }
}

const checkboxGroups = currentStepEl.querySelectorAll(".checkbox-group");
checkboxboxGroups.forEach(group => {
  const checkboxes = group.querySelectorAll("input[type='checkbox']");
  const oneChecked = Array.from(checkboxboxes).some(cb => cb.checked);
  if (!oneChecked) {
    isValid = false;
    group.classList.add("invalid");
  } else {

```

```

        group.classList.remove("invalid");
    }
});

    if (isValid && currentStep < steps.length - 1) {
        steps[currentStep].classList.remove("active");
        currentStep++;
        showStep(currentStep);
    }
});

prevBtn.addEventListener("click", () => {
    if (currentStep > 0) {
        steps[currentStep].classList.remove("active");
        currentStep--;
        showStep(currentStep);
    }
});

function showStep(index) {
    const progressInner = document.getElementById("progress-inner");
    if (progressInner) {
        const percent = ((index + 1) / steps.length) * 100;
        progressInner.style.width = percent + "%";
    }
    steps.forEach((stepEl, i) => {
        stepEl.classList.toggle("active", i === index);
    });
    prevBtn.style.display = index === 0 ? "none" : "inline-block";
    nextBtn.style.display = index === steps.length - 1 ? "none" : "inline-block";
    submitBtn.style.display = index === steps.length - 1 ? "inline-block" : "none";
}

form.addEventListener("submit", function(event) {
    event.preventDefault();

    let hasError = false;
    let firstInvalidInput = null;

    steps.forEach((stepEl, index) => {
        const inputs = stepEl.querySelectorAll("input, select");
        inputs.forEach(input => {
            if (input.closest(".hidden") || input.closest(".hidden-block")) {
                return;
            }
            if (input.hasAttribute("required") && !input.value.trim()) {
                input.classList.add("invalid");
                if (!firstInvalidInput) {
                    firstInvalidInput = input;
                    currentStep = index;
                }
                hasError = true;
            } else {
                input.classList.remove("invalid");
            }
        });
    });

    if (hasError && firstInvalidInput) {
        steps.forEach(el => el.classList.remove("active"));
        showStep(currentStep);
        setTimeout(() => {
            firstInvalidInput.scrollIntoView({ behavior: "smooth", block: "center" });
            firstInvalidInput.focus();
        });
    }
});

```

```

    }, 100);
    return;
}

showResult();
});

function showResult() {
    form.style.display = "none";
    document.getElementById("final-result").style.display = "block";

    const get = id => parseFloat(document.getElementById(id).value) || 0;
    const val = id => document.getElementById(id).value;

    // ==== Транспорт ====
    let carKm = get("car-km");
    let transport = 0;
    if (document.getElementById("car-diesel").checked) {
        transport += carKm * 0.21 * 1.15;
    }
    if (document.getElementById("car-petrol").checked) {
        transport += carKm * 0.21 * 1.0;
    }
    if (document.getElementById("car-gas").checked) {
        transport += get("gas-auto-m3") * 2.02;
    }
    if (document.getElementById("car-electric").checked) {
        transport += carKm * 0.21 * 0.3;
        transport += get("electric-car-kwh") * 0.09;
    }
    transport += get("public-transport") * 0.105;
    transport -= get("bike-days") * 0.05;

    // ==== Электрика ====
    let electric = get("electricity") * 0.5 / 4;
    electric += get("electric-cooking") * 0.25;
    electric += get("computer-hours") * 0.2;
    if (val("electric-heating") === "yes") electric *= 1.5;
    electric *= 1 - (get("led-lamps") / 100 * 0.3);
    const electricCooking = val("use-electric-cooking") === "yes" ? get("electric-cooking")
* 0.25 : 0;

    // ==== Газ ====
    const gasM3 = get("gas-m3");
    let gas = 0;
    if (val("gas-use-heat") === "yes") {
        gas += gasM3 * 2.02 * 0.6;
    }
    if (val("gas-use-water") === "yes") {
        gas += gasM3 * 2.02 * 0.2;
    }
    if (val("use-gas-cooking") === "yes") {
        gas += get("gas-cooking-hours") * 0.2;
    }
    if (gas === 0 && gasM3 > 0) {
        gas = gasM3 * 2.02;
    }

    // ==== Ыжа ====
    const food = get("meat") * 27 +
        get("dairy") * 6 +
        get("eggs") * 0.5 +
        get("vegetables") * 2 +
        get("food-delivery") * 5;

```

```

// ==== Вода ====
let water = get("shower-minutes") * get("shower-frequency") * 0.05 +
    get("washing-machine") * 0.3 +
    get("dishwasher-uses") * 1.5;
if (val("water-saving") === "yes") water *= 0.8;

// ==== Покупки ====
let shopping = (get("clothes") * 25 + get("shoes") * 30 + get("gadgets") * 200) / 52 +
    get("plastic-bags") * 0.1;
const secondhand = val("secondhand");
if (secondhand === "often") shopping *= 0.6;
else if (secondhand === "sometimes") shopping *= 0.8;

// ==== Відходи ====
let waste = get("waste") * (val("recycling") === "yes" ? 0.3 : 0.6);
waste -= get("bottles") * 0.05;
waste *= 1 - (get("trash-percent") / 100);
if (val("organic-compost") === "yes") waste *= 0.9;


// ==== Побут ====
let habits = get("tv-hours") * 0.2 +
    Math.max(0, get("heating-temp") - 21) * 2 +
    get("ac-use") * 0.8;
const lights = val("lights-off");
if (lights === "always") habits *= 0.9;
else if (lights === "sometimes") habits *= 0.95;
const insulation = val("home-insulation");
if (insulation === "yes") habits *= 0.8;
else if (insulation === "partial") habits *= 0.9;

// ==== Ітоги ====
const values = [transport, electric, gas, food, water, shopping, waste, habits];
const total = values.reduce((a, b) => a + b, 0);
const labels = ["Авто", "Електрика", "Газ", "Їжа", "Вода", "Покупки", "Відходи",
"Побут"];
const colors = ["#FFCBA4", "#CAB8FF", "#FFE7B4", "#C1FBA4", "#A0E7E5", "#FFB5C2",
"#B2C9A0", "#fffd9e"];

// === Сортуюмо від більшого до меншого ===
const dataset = labels.map((label, i) => ({
label,
value: values[i],
color: colors[i],
originalIndex: i
}));
dataset.sort((a, b) => b.value - a.value);

// Перезаписуємо labels, values, colors у новому порядку
const sortedLabels = dataset.map(item => item.label);
const sortedValues = dataset.map(item => item.value);
const sortedColors = dataset.map(item => item.color);
const sortedOriginalIndices = dataset.map(item => item.originalIndex);

// === Вивід результатів ===
const resultBlock = document.getElementById("result");
resultBlock.innerHTML = `Ваш орієнтовний екологічний слід: ${total.toFixed(2)} кг CO2`;

const adviceBox = document.getElementById("advice-box");
const maxIndex = values.indexOf(Math.max(...values));
const advices = [
// 0 – Транспорт
" Спробуйте хоча б один день на тиждень залишити авто вдома спробуйте поїхати велосипедом, пішки або громадським транспортом. Це корисно і для планети, і для здоров'я!";

```

# ДОДАТОК Б. Слайди мультимедійної презентації

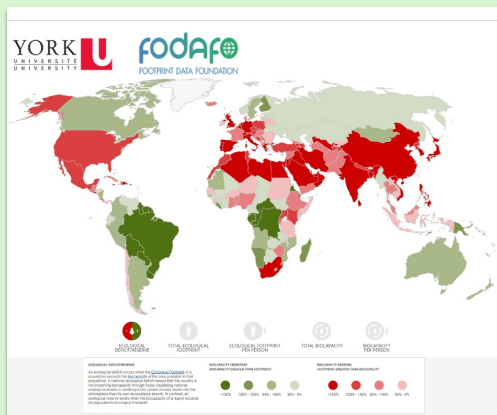
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

КГ.08.04.000.ДП

## Дипломний проєкт

На тему:  
**Розробка експертного веб-застосунку розрахунку екологічних слідів**

Вельченка Кирила Ігоровича  
Спеціальність: 123 «Комп'ютерна інженерія»  
Освітньо-професійна програма: «Комп'ютерна графіка і Web-дизайн»



### Вступ: Оцінка особистих викидів CO<sub>2</sub>



#### Вебзастосунок

Інтуїтивний онлайн-інструмент для оцінки особистого екологічного сліду на основі повсякденних дій користувача.



#### Екологічний слід

Відображає сукупний вплив життєдіяльності людини на довкілля, зокрема викиди парникових газів.



#### Викиди CO<sub>2</sub>

Фокус на вуглецевому сліді як ключовому індикаторі впливу на клімат.

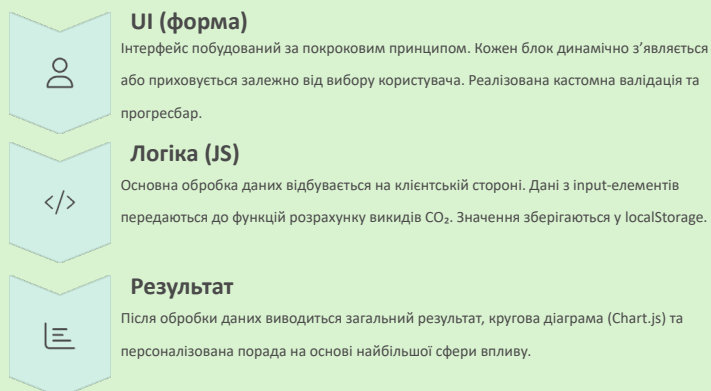
Застосунок дозволяє наочно оцінити вплив звичних побутових дій на навколишнє середовище та підвищити екологічну обізнаність користувачів.

## Використані технології

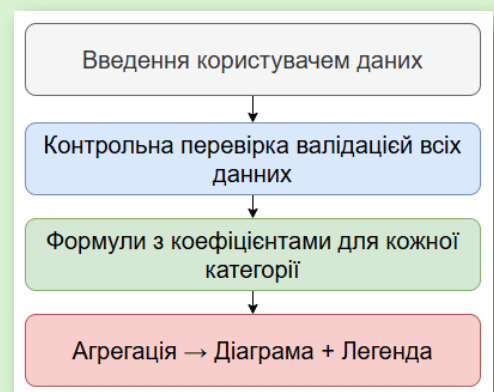
- HTML, CSS, JS**  
Базові вебтехнології для створення структури, стилю та логіки застосунку.
- Chart.js**  
Бібліотека для побудови інтерактивної кругової діаграми.
- localStorage**  
Механізм локального зберігання даних у браузері користувача.
- GitHub Pages**  
Хостинг для публікації застосунку з відкритим доступом.



## Архітектура застосунку

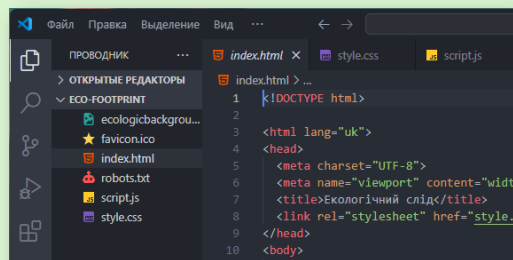
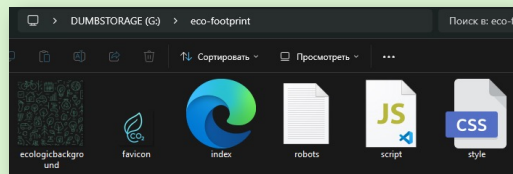


Така модульна архітектура забезпечує гнучкість застосунку та легкість підтримки коду.



## Файлова структура проекту

- 1 index.html**  
Основна HTML-структура, покрокова форма, контейнер результатів калькулятора.
- 2 style.css**  
Відповідає за адаптивність, стилізацію та візуальну логіку застосунку.
- 3 script.js**  
Функціональна логіка, розрахунок CO<sub>2</sub>, генерація діаграми.
- 4 favicon.ico**  
Іконка, що відображається на вкладці браузера користувача.
- 5 ecologicbackground.png**  
Фонове зображення, що підкреслює екологічну тематику проекту.



Чітка структура файлів спрощує підтримку коду, робить проект зрозумілим та масштабованим.

## Ключовий функціонал

### Покрокова форма

Форма розбита на окремі тематичні блоки з динамічним відображенням вмісту залежно від вибору користувача.

### Редагування

Користувач може повернутися до попередніх кроків та змінити введені дані перед отриманням результату.

### Валідація

Кастомна перевірка правильності введення лише активних полів без використання зовнішніх бібліотек.

### Прогресбар

Візуальний індикатор поточного етапу заповнення, що допомагає орієнтуватись у процесі.

Який у вас автомобіль?

- Дизельне авто
- Бензинове авто
- Газове авто
- Електромобіль
- Не користуюсь авто

Оберіть хоча б один варіант

Редагувати відповіді



Реалізований функціонал забезпечує зручний, зрозумілий і безперервний досвід взаємодії користувача із застосунком.

## Алгоритм розрахунку CO<sub>2</sub>

### Збір даних

Користувач вводить дані про транспорт, електрику, їжу тощо.

### Застосування коефіцієнтів

Введені значення множаться на коефіцієнти викидів CO<sub>2</sub>.

### Сумарний розрахунок

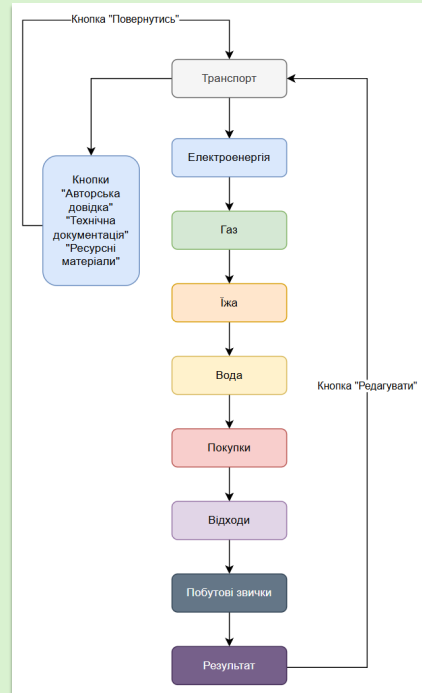
Всі обчислені значення додаються до загального сліду.

### Визначення домінантної сфери

Знаходиться категорія з найбільшими викидами для поради.

### Генерація результату

Виводиться діаграма, деталі та персоналізована порада.



## UI/UX Дизайн Застосунку



### Чистий та Адаптивний

Спрощений дизайн з чіткими блоками, великими полями. Повністю адаптується до будь-якого екрану.



### Екологічна Тематика

Кольорова палітра виконана у стилі "еко", із зеленими акцентами. Доступна світла та темна теми.



### Покрокова Взаємодія

Форма розбита на логічні етапи. Кожен крок має заголовок та чіткі вказівки для зручності.



### Анімації та Підказки

Плавне з'явлення елементів та підказки при введенні даних. Прогресбар допомагає відстежувати процес.

## Результат розрахунку



### Загальний результат:

Користувач бачить суму викидів CO<sub>2</sub> на тиждень у кілограмах.

### Візуалізація (діаграма):

Кругова діаграма показує розподіл викидів за категоріями.

### Персоналізована порада:

Система генерує індивідуальну рекомендацію на основі найбільшої сфери впливу.

### Можливість редагування:

Користувач може змінити відповіді та перерахувати результат.

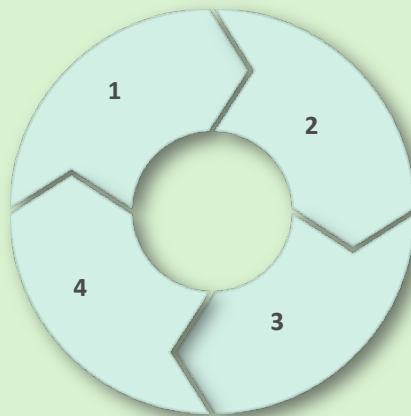
## Кнопка «Редагувати відповіді»

### Гнучкість Взаємодії

Користувач може легко повернутися до будь-якого кроку форми.

### Зручність та Контроль

Це підвищує довіру до результатів та допомагає оцінити вплив.



### Збереження Даних

Введені дані автоматично зберігаються, уникаючи повторного введення.

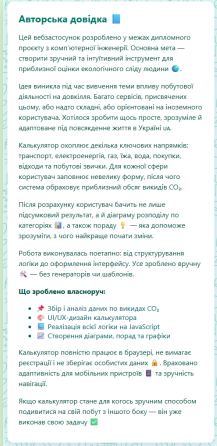
### Миттєве Оновлення

Після редагування система одразу перераховує новий екологічний слід.

# Інформаційні Сторінки Застосунку

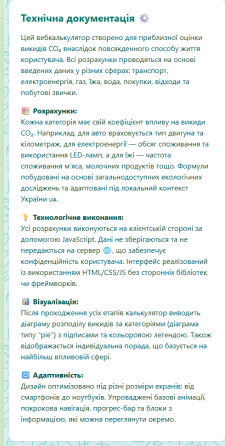
## Авторська довідка

Опис розробника, мети створення та освітнього контексту проекту.



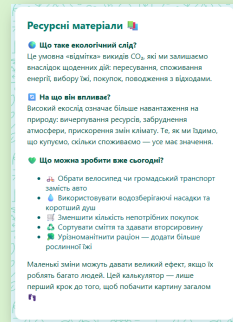
## Технічна документація

Технічна інформація про функціонал, логіку та використані технології.



## Ресурсні матеріали

Посилання на джерела даних, коефіцієнти викидів та екологічні калькулятори.



# Тестування Застосунку

## 1 Крос-платформене тестування:

Перевірено у Chrome, Firefox, Safari на ПК та мобільних пристроях.

## 2 Адаптивний дизайн:

Всі елементи коректно відображаються на екранах будь-якого розміру.

## 3 Обробка помилок:

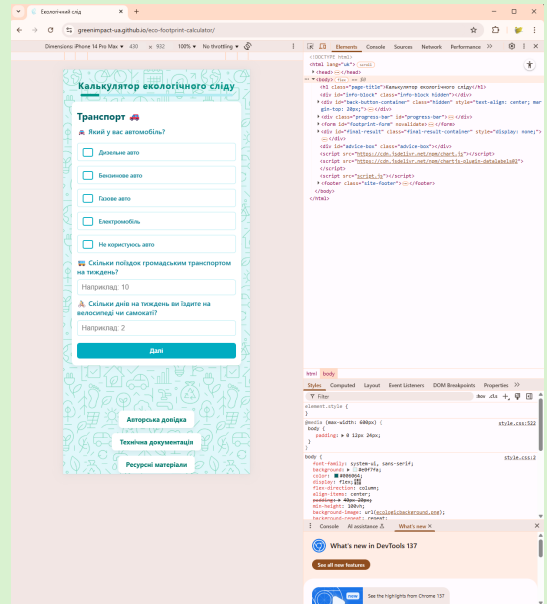
Реалізовано валідацію обов'язкових полів для запобігання невірному введенню.

## 4 Збереження даних:

Введені значення залишаються після перезавантаження завдяки localStorage.

## 5 Обробка граничних випадків:

Система коректно працює з нульовими, великими чи пустими значеннями.



# Оптимізація Застосунку



## Оптимізований Розмір

Усі файли мінімізовані, без зайвих залежностей, що забезпечує надзвичайно швидке завантаження.



## Ефективна Валідація

Валідація через JavaScript перевіряє лише активні поля, значно зменшуючи системне навантаження.



## Чітке Розділення Логіки

CSS, HTML та JavaScript чітко розділені, що спрощує підтримку та сприяє майбутньому масштабуванню проєкту.



## Асинхронне Завантаження

Ресурси завантажуються асинхронно, що покращує взаємодію з користувачем та забезпечує безперебійну роботу інтерфейсу під час завантаження даних.

# Публікація Застосунку

## 1 Доступність

Розміщено на GitHub Pages, доступний за посиланням для всіх користувачів.

## 2 Немає встановлення

Застосунок не вимагає реєстрації чи встановлення, запускається прямо у браузері.

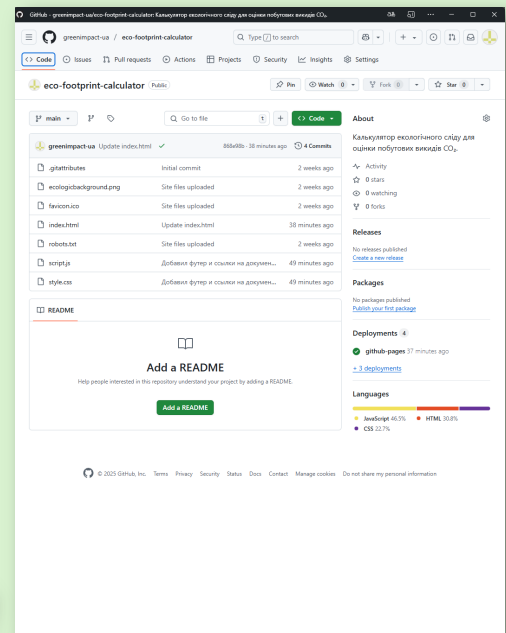
## 3 Постійний доступ

Доступний 24/7 онлайн з будь-якого пристрою: смартфона, планшета чи ПК.

## 4 Швидкий доступ

Швидкий доступ до сайту через QR-код на сторінці презентації.

Сайт застосунку:  
<https://greenimpact-ua.github.io/eco-footprint-calculator/>



# Висновки

## Повноцінний Застосунок

Було реалізовано повнофункціональний інструмент, що дозволяє кожному користувачу оцінити свій особистий екологічний слід на основі звичних дій. Проект охоплює всі етапи взаємодії: від введення до отримання персоналізованого результату.

## Інтуїтивний Інтерфейс

Інтерфейс спроектовано у покроковій структурі, що робить його простим у користуванні навіть для новачків. Адаптивність, валідація, прогресбар та редагування відповідей — усе спрямоване на зручність користувача.

## Оптимізована Архітектура

Проект створено без використання фреймворків — лише HTML, CSS та JavaScript. Такий підхід забезпечує швидке завантаження, легку підтримку та незалежність від сторонніх бібліотек.

## Цінність та Доступність

Застосунок розміщено на GitHub Pages і доступний з будь-якого пристрою. Це дозволяє не тільки демонструвати розробку, але й реально використовувати її як просвітницький інструмент.

## РЕЦЕНЗІЯ

на дипломний проект здобувача (здобувачки) освіти  
відділення комп'ютерних систем

*Вельченка Кирила Ігоровича*

(прізвище, ім'я та по батькові)

Спеціальність 123 «Комп'ютерна інженерія»

Освітня програма «Комп'ютерна графіка і Web-дизайн»

Керівник дипломного проекту (роботи) Жадан Артур Сергійович

(прізвище, ім'я та по батькові)

Тема дипломного проекту (роботи) Розробка експертного веб-застосунку розрахунку екологічних слідів

Обсяг розрахунково-пояснювальної записки 77 сторінок

Обсяг графічної (презентаційної) частини 15 аркушів (слайдів)

### ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ (РОБОТИ)

а) заключення про ступінь відповідності виконаного дипломного проекту завданню

*Представлений на рецензію дипломний проект відповідає затвердженій темі та виконаний відповідно технічному завданню. Дипломний проект присвячений проблемі екології та складається з пояснювальної записки, додатку з програмним кодом та мультимедійної презентації, що містить приклади роботи програми.*

б) характеристика виконання кожного розділу дипломного проекту

*Пояснювальна записка складається з основного розділу (аналізу предметної області, проектування застосунку, реалізації застосунку, тестування застосунку), економічного розділу, розділу охорони праці та додатків. Перелічені розділи поетапно охоплюють розробку, виконані докладно та обґрунтовано. Розділ охорони праці містить загальну інформацію та вимоги до техніки безпеки оператора КТ. Економічний розділ проекту містить розрахунок витрат на НДР та реалізацію проекту.*

в) оцінка якості виконання пояснювальної записки та графічної частини дипломного проекту

*Графічна частина складається з 15 слайдів мультимедійної презентації, виконаної у програмному продукті MS PowerPoint, які містять ілюстративні схеми, скріншоти роботи програмного застосунку, передбачені технічним завданням. Пояснювальна записка виконана акуратно та у відповідності до норм. Якість виконання графічної частини проекту та пояснювальної записки добра, розробку виконано у повному обсязі.*

г) перелік позитивних якостей дипломного проекту SPA-підхід з покроковою навігацією, зручно для користувача та добре організовано для розробника.

Використання localStorage для збереження прогресу, підказки в реальному часі, можливість редагування відповідей. Застосовано CSS Grid/Flexbox, адаптивний дизайн, підтримка мобільних пристроїв.

д) основні недоліки дипломного проекту \_\_\_\_\_

Без системи автентифікації й централізованого зберігання користувацьких даних неможлива синхронізація між пристроями та аналітика історії; проект перевірено тільки вручну - відсутні unit- або e2e-тести для критичних компонентів; використання React/Vue могло б забезпечити кращу масштабованість проекту; деякі недоліки оформлення пояснювальної записки

Оцінка розрахункової частини \_\_\_\_\_ добре

Оцінка графічної частини \_\_\_\_\_ добре

Загальна оцінка \_\_\_\_\_ добре

Прізвище, ім'я, по батькові рецензента \_\_\_\_\_ к.т.н. Рудніченко Микола Дмитрович

Місце роботи і посада рецензента \_\_\_\_\_ Національний університет «Одеська політехніка», доцент кафедри інформаційних технологій

Підпис \_\_\_\_\_

« 20 \_\_\_\_\_ 2025 р.



**ВІДГУК**

керівника на дипломний проект здобувача (здобувачки) освіти  
відділення комп'ютерних систем

*Вельченка Кирила Ігоровича*

(прізвище, ім'я та по батькові)

Спеціальність: 123 «Комп'ютерна інженерія»

Освітня програма: «Комп'ютерна графіка і Web-дизайн»

Тема дипломного проекту: Розробка експертного веб-застосунку розрахунку екологічних слідів

**ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ**

- а) обсяг і якість виконання проекту (графічного матеріалу і розрахунково-пояснювальної записки) Дипломний проект виконано відповідно технічному завданню. Пояснювальна записка до дипломного проекту містить 77 сторінки. У пояснювальній записці описано етапи розробки експертного веб-застосунку розрахунку екологічних слідів засобами HTML, CSS та JavaScript. Графічна частина складається з окремих 15 слайдів, оформлених у вигляді презентації, передбачених технічним завданням. Якість виконання пояснювальної записки та слайдів добра.
- б) самостійність роботи над проектом: Протягом виконання дипломного проекту здобувач освіти Вельченко Кирил поступово та послідовно виконував всі етапи, проявляв ініціативу в створенні загальної концепції та реалізації роботи. Всі роботи здобувач освіти виконував самостійно, з оглядом на рекомендації керівника.
- в) теоретична підготовка випускника (випускниці): Здобувач освіти Вельченко Кирил під час роботи над дипломним проектом вивчив достатньо багато літературних та інтернет-джерел за даною тематикою. Вважаю, що теоретична підготовка дипломника достатня і він готовий до захисту проекту.

г) вміння розв'язувати виробничі та конструкторські питання Під час виконання дипломного проекту здобувач освіти Вельченко Кирил показав вміння організовано працювати над поставленим завданням, застосовувати знання у галузі програмування та математики, розробляти, встановлювати та налаштовувати спеціалізоване програмне забезпечення, оформлювати слайди та складати презентації, користуючись сучасними комп'ютерними програмними засобами, такими як MS VS Code, HTML, CSS, JavaScript.

Оцінка розрахункової частини Добре  
Оцінка графічної частини Відмінно  
Загальна оцінка Відмінно

Прізвище, ім'я, по батькові керівника дипломного проекту Жадан Артур Сергійович

Місце роботи і посада керівника дипломного проекту ВСП «Одеський технічний фаховий коледж ОНТУ», викладач спецдисциплін циклової комісії комп'ютерної техніки та програмної інженерії

Підпис

  
«16» серпня 2025 р.

**ДОЗВІЛ  
НА РОЗМІЩЕННЯ  
ВИПУСКНОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ  
(ДИПЛОМНОГО ПРОЕКТУ)  
В ЕЛЕКТРОННОМУ РЕПОЗИТАРІЇ ВСП «ОТФК ОНТУ»**

Ми, що нижче підписалися,

**Вельченко Кирил Ігорович,**  
здобувач освіти гр. 4КГ-08, та

**Жадан Артур Сергійович,**  
керівник дипломного проекту,

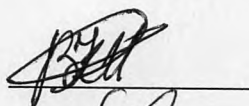
не заперечуємо щодо розміщення електронного варіанту пояснювальної записки до дипломного проекту фахового молодшого бакалавра на тему:

**«Розробка експертного веб-застосунку розрахунку екологічних слідів»  
(автор роботи – Вельченко К.І., керівник роботи – Жадан А.С.)**

виконаного у ВСП «Одеський технічний фаховий коледж Одеського національного технологічного університету» в 2025 році, у повному обсязі в електронному репозитарії ВСП «ОТФК ОНТУ» для вільного доступу через мережу Інтернет.

Несемо відповідальність за ідентичність електронного та друкованого варіантів випускної кваліфікаційної роботи і даємо згоду на обробку персональних даних.

Виконавець



/ Вельченко К.І. /

Керівник



/ Жадан А.С. /

«16» червня 2025 р.

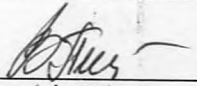
# Д О В І Д К А

циклової комісії КТ та ПП  
про допуск до захисту дипломного проекту  
здобувача (здобувачки) освіти IV курсу  
відділення комп'ютерних систем групи 4КГ-08

*Вельченка Кирила Ігоровича*

на тему Розробка експертного веб-застосунку  
розрахунку екологічних слідів

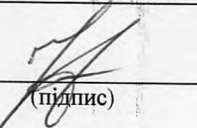
Висновок відповідальної особи за проведення нормоконтролю:  
пояснювальна записка до дипломного проекту виконана з деякими  
порушеннями ДСТУ та оформлена відповідно до вимог Положення про  
дипломне проектування

  
(підпис)

16.06.2025  
(дата)

Петрашова В.І.  
(П.І.Б.)

Висновок відповідальної особи за перевірку роботи на наявність академічного  
плагиату згідно звіту про перевірку від 16.06.2025 р. значення коефіцієнту  
подібності в роботі становить 12,84%, коефіцієнт цитування – 3,33%.

  
(підпис)

16.06.2025  
(дата)

Краснокутська К.Г.  
(П.І.Б.)

**Попередня експертиза (малий захист) дипломного проекту**

здобувача (здобувачки) освіти

Вельченка К.І.  
(П.І.Б.)

проведена « 16 » червня 2025 р.

Висновки Пояснювальна записка до дипломного проекту виконана у повному  
обсязі. Випускна кваліфікаційна робота (дипломний проект) відповідає  
вимогам Положення про дипломне проектування та рекомендована до  
захисту.

Голова ЦК КТ та ПП

  
(підпис)

Кривченко Ю.В.  
(П.І.Б.)

## Звіт подібності

## метадані

Назва організації

Odesa Technical Professional College of Odesa National University of Technology

Заголовок

Розробка експертного веб-застосунку розрахунку екологічних слідів

Автор

Науковий керівник / Експерт

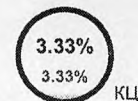
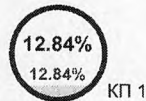
Вельченко Кирил ІгоровичЖадан Артур Сергійович

підрозділ

Відокремлений структурний підрозділ "Одеський технічний фаховий коледж Одеського національного технологічного університету"

## Обсяг знайдених подібностей

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.



25

Довжина фрази для коефіцієнта подібності 2

15670

Кількість слів

128813

Кількість символів

## Тривога

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про МОЖЛИВІ маніпуляції в тексті. Спотворення в тексті можуть мати навмисний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв	Ⓡ	16
Інтервали	A→	0
Мікропробіли		0
Білі знаки	Ⓡ	309
Парафрази (SmartMarks)	a	70

## Подібності за списком джерел

Нижче наведений список джерел. В цьому списку є джерела із різних баз даних. Колір тексту означає в якому джерелі він був знайдений. Ці джерела і значення Коефіцієнту Подібності не відображають прямого плагіату. Необхідно відкрити кожне джерело і проаналізувати зміст і правильність оформлення джерела.

## 10 найдовших фраз

ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	Копію тексту
		КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	<a href="https://card-file.ontu.edu.ua/bitstreams/53ed22ad-8700-4162-b97a-082a1ad472d6/download">https://card-file.ontu.edu.ua/bitstreams/53ed22ad-8700-4162-b97a-082a1ad472d6/download</a>	69 0.44 %
2	<a href="https://www.stud24.ru/management/ognchn-vimogi/511190-2182185-page1.html">https://www.stud24.ru/management/ognchn-vimogi/511190-2182185-page1.html</a>	55 0.35 %
3	<a href="https://card-file.ontu.edu.ua/bitstreams/9908b7a9-6b3e-46f5-a46e-84d83787cfd4/download">https://card-file.ontu.edu.ua/bitstreams/9908b7a9-6b3e-46f5-a46e-84d83787cfd4/download</a>	50 0.32 %
4	<a href="https://card-file.ontu.edu.ua/bitstreams/9908b7a9-6b3e-46f5-a46e-84d83787cfd4/download">https://card-file.ontu.edu.ua/bitstreams/9908b7a9-6b3e-46f5-a46e-84d83787cfd4/download</a>	45 0.29 %
5	<a href="https://card-file.ontu.edu.ua/bitstreams/9908b7a9-6b3e-46f5-a46e-84d83787cfd4/download">https://card-file.ontu.edu.ua/bitstreams/9908b7a9-6b3e-46f5-a46e-84d83787cfd4/download</a>	41 0.26 %

6	<a href="https://card-file.ontu.edu.ua/server/api/core/bitstreams/995bdcec-4e4d-4321-8070-4d6badcb8e49/content">https://card-file.ontu.edu.ua/server/api/core/bitstreams/995bdcec-4e4d-4321-8070-4d6badcb8e49/content</a>	40 0.26 %
7	<a href="https://studfile.net/preview/9260123/">https://studfile.net/preview/9260123/</a>	39 0.25 %
8	<a href="https://card-file.ontu.edu.ua/bitstreams/82a6d375-2b69-4233-b80f-fbfd149b7747/download">https://card-file.ontu.edu.ua/bitstreams/82a6d375-2b69-4233-b80f-fbfd149b7747/download</a>	37 0.24 %
9	<a href="http://referatiks.ru/oxorona-prac-koristuvach-v-pk_3923.html">http://referatiks.ru/oxorona-prac-koristuvach-v-pk_3923.html</a>	34 0.22 %
10	<a href="https://card-file.ontu.edu.ua/bitstreams/549ee9fe-7574-4ae5-b500-9fe2711f33e6/download">https://card-file.ontu.edu.ua/bitstreams/549ee9fe-7574-4ae5-b500-9fe2711f33e6/download</a>	32 0.20 %

### з домашньої бази даних (0.40 %)

ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	Розробка 3D-гри у жанрі survival-horror з налаштуваннями рівнів складності 6/12/2025 Odesa Technical Professional College of Odesa National University of Technology (Відокремлений структурний підрозділ "Одеський технічний фаховий коледж Одеського національного технологічного університету")	63 (4) 0.40 %

### з програми обміну базами даних (0.00 %)

ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
------------------	-----------	--

### з Інтернету (12.44 %)

ПОРЯДКОВИЙ НОМЕР	ДЖЕРЕЛО URL	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	<a href="https://card-file.ontu.edu.ua/server/api/core/bitstreams/44c16132-5f53-48e2-b6c0-61e9a2f0fd75/content">https://card-file.ontu.edu.ua/server/api/core/bitstreams/44c16132-5f53-48e2-b6c0-61e9a2f0fd75/content</a>	536 (44) 3.42 %
2	<a href="https://card-file.ontu.edu.ua/bitstreams/1dff552d-7200-49b8-ae1d-ba76a1335685/download">https://card-file.ontu.edu.ua/bitstreams/1dff552d-7200-49b8-ae1d-ba76a1335685/download</a>	213 (13) 1.36 %
3	<a href="https://card-file.ontu.edu.ua/bitstreams/53ed22ad-8700-4162-b97a-082a1ad472d6/download">https://card-file.ontu.edu.ua/bitstreams/53ed22ad-8700-4162-b97a-082a1ad472d6/download</a>	170 (9) 1.08 %
4	<a href="http://referatiks.ru/oxorona-prac-koristuvach-v-pk_3923.html">http://referatiks.ru/oxorona-prac-koristuvach-v-pk_3923.html</a>	141 (8) 0.90 %
5	<a href="https://card-file.ontu.edu.ua/bitstreams/9908b7a9-6b3e-46f5-a46e-84d83787cfd4/download">https://card-file.ontu.edu.ua/bitstreams/9908b7a9-6b3e-46f5-a46e-84d83787cfd4/download</a>	136 (3) 0.87 %
6	<a href="https://card-file.ontu.edu.ua/bitstreams/62baa43e-b968-4993-bb54-8cf8761a89b2/download">https://card-file.ontu.edu.ua/bitstreams/62baa43e-b968-4993-bb54-8cf8761a89b2/download</a>	86 (7) 0.55 %
7	<a href="https://card-file.ontu.edu.ua/bitstreams/55e2b8f2-7d3c-4235-99fc-2be51199b96d/download">https://card-file.ontu.edu.ua/bitstreams/55e2b8f2-7d3c-4235-99fc-2be51199b96d/download</a>	61 (5) 0.39 %
8	<a href="https://card-file.ontu.edu.ua/server/api/core/bitstreams/a141b658-5fa7-4f90-b0bd-7f0ccaed21e5/content">https://card-file.ontu.edu.ua/server/api/core/bitstreams/a141b658-5fa7-4f90-b0bd-7f0ccaed21e5/content</a>	59 (6) 0.38 %
9	<a href="https://card-file.ontu.edu.ua/bitstreams/549ee9fe-7574-4ae5-b500-9fe2711f33e6/download">https://card-file.ontu.edu.ua/bitstreams/549ee9fe-7574-4ae5-b500-9fe2711f33e6/download</a>	57 (3) 0.36 %
10	<a href="https://www.stud24.ru/management/ggnchn-vimogi/511190-2182185-page1.html">https://www.stud24.ru/management/ggnchn-vimogi/511190-2182185-page1.html</a>	55 (1) 0.35 %
11	<a href="https://card-file.ontu.edu.ua/bitstreams/82a6d375-2b69-4233-b80f-fbfd149b7747/download">https://card-file.ontu.edu.ua/bitstreams/82a6d375-2b69-4233-b80f-fbfd149b7747/download</a>	55 (4) 0.35 %
12	<a href="https://card-file.ontu.edu.ua/bitstreams/035f6436-20b4-4ee6-8e99-bede670e308b/download">https://card-file.ontu.edu.ua/bitstreams/035f6436-20b4-4ee6-8e99-bede670e308b/download</a>	53 (5) 0.34 %
13	<a href="https://card-file.ontu.edu.ua/server/api/core/bitstreams/995bdcec-4e4d-4321-8070-4d6badcb8e49/content">https://card-file.ontu.edu.ua/server/api/core/bitstreams/995bdcec-4e4d-4321-8070-4d6badcb8e49/content</a>	47 (2) 0.30 %
14	<a href="https://card-file.ontu.edu.ua/bitstreams/bbed74c8-2ea7-44c5-8d00-0fe3fd9790ee/download">https://card-file.ontu.edu.ua/bitstreams/bbed74c8-2ea7-44c5-8d00-0fe3fd9790ee/download</a>	45 (4) 0.29 %
15	<a href="https://studfile.net/preview/9260123/">https://studfile.net/preview/9260123/</a>	39 (1) 0.25 %
16	<a href="https://card-file.ontu.edu.ua/bitstreams/bbaf3f38-16a8-4070-bead-5562769b7c71/download">https://card-file.ontu.edu.ua/bitstreams/bbaf3f38-16a8-4070-bead-5562769b7c71/download</a>	37 (2) 0.24 %
17	<a href="https://card-file.ontu.edu.ua/bitstreams/e4afae26-0a7e-4a4d-afc2-94341838de2a/download">https://card-file.ontu.edu.ua/bitstreams/e4afae26-0a7e-4a4d-afc2-94341838de2a/download</a>	24 (1) 0.15 %

18	<a href="https://card-file.ontu.edu.ua/bitstreams/29489599-0561-4ce6-8890-c3b13d9f2e0e/download">https://card-file.ontu.edu.ua/bitstreams/29489599-0561-4ce6-8890-c3b13d9f2e0e/download</a>	19 (3) 0.12 %
19	<a href="https://studopedia.org/5-118043.html">https://studopedia.org/5-118043.html</a>	16 (1) 0.10 %
20	<a href="https://card-file.ontu.edu.ua/bitstreams/34a6756b-592f-4b77-a805-183aa03a6a26/download">https://card-file.ontu.edu.ua/bitstreams/34a6756b-592f-4b77-a805-183aa03a6a26/download</a>	16 (1) 0.10 %
21	<a href="https://card-file.ontu.edu.ua/bitstreams/11562741-24e6-4201-bc41-a00c8013fca1/download">https://card-file.ontu.edu.ua/bitstreams/11562741-24e6-4201-bc41-a00c8013fca1/download</a>	16 (1) 0.10 %
22	<a href="https://card-file.ontu.edu.ua/bitstreams/6cf43324-8f08-4031-ba42-f80b18efbbc8/download">https://card-file.ontu.edu.ua/bitstreams/6cf43324-8f08-4031-ba42-f80b18efbbc8/download</a>	15 (2) 0.10 %
23	<a href="https://os.nure.ua/wp-content/uploads/2019/05/met_zao_keruvannya_2016.pdf">https://os.nure.ua/wp-content/uploads/2019/05/met_zao_keruvannya_2016.pdf</a>	15 (2) 0.10 %
24	<a href="https://kosatka.media/uk/category/elektroenergiya/news/nkreku-utverdila-licenzionnye-usloviya-pohraneniyu-energii">https://kosatka.media/uk/category/elektroenergiya/news/nkreku-utverdila-licenzionnye-usloviya-pohraneniyu-energii</a>	11 (1) 0.07 %
25	<a href="https://ela.kpi.ua/bitstream/123456789/58194/1/Storozhuk_bakalavr.pdf">https://ela.kpi.ua/bitstream/123456789/58194/1/Storozhuk_bakalavr.pdf</a>	8 (1) 0.05 %
26	<a href="https://card-file.ontu.edu.ua/bitstreams/5240e379-7721-49f0-8ee8-27140b0b473a/download">https://card-file.ontu.edu.ua/bitstreams/5240e379-7721-49f0-8ee8-27140b0b473a/download</a>	7 (1) 0.04 %
27	<a href="https://studopedia.org/5-118045.html">https://studopedia.org/5-118045.html</a>	7 (1) 0.04 %
28	<a href="https://card-file.ontu.edu.ua/server/api/core/bitstreams/ead3fa83-2e3d-4cd7-bfbd-1d5ed04c1ce4/content">https://card-file.ontu.edu.ua/server/api/core/bitstreams/ead3fa83-2e3d-4cd7-bfbd-1d5ed04c1ce4/content</a>	5 (1) 0.03 %

### Список прийнятих фрагментів (немає прийнятих фрагментів)

ПОРЯДКОВИЙ НОМЕР	ЗМІСТ	КІЛЬКІСТЬ ОДНАКОВИХ СЛІВ (ФРАГМЕНТІВ)
------------------	-------	---------------------------------------

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: 123 «Комп'ютерна інженерія»  
Освітньо-професійна програма: «Комп'ютерна графіка і Web-дизайн» Група: 4КГ-08

Дипломний проект  
здобувача освіти денної форми навчання  
КГ.08.03.000.ДП.

ВЕЛЬЧЕНКА  
КИРИЛА ІГОРОВИЧА

м. Одеса  
2025 р. МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: 123 «Комп'ютерна інженерія»  
Освітньо-професійна програма: «Комп'ютерна графіка і Web-дизайн»  
Група: 4 КГ-08

ПОЯСНЮВАЛЬНА ЗАПИСКА до дипломного проекту на тему:

\_\_\_\_\_ Проектний матеріал складається з  
пояснювальної записки на \_\_\_\_\_ сторінках та графічного (презентаційного) матеріалу на \_\_\_\_\_ аркушах (слайдах) Дипломник  
\_\_\_\_\_ ( Вельченко К.І. )

Керівник \_\_\_\_\_ ( Жадан А.С. )

Консультанти:

з економічного розділу \_\_\_\_\_ ( Канський М.Ю. )

з розділу охорони праці та техніки безпеки \_\_\_\_\_ ( Чорновол Н.І. ) з нормоконтролю

\_\_\_\_\_ ( Петрашова В.І. ) старший консультант \_\_\_\_\_ ( Кривченко Ю. В. )

До захисту допущений

Голова циклової комісії \_\_\_\_\_ ( Кривченко Ю.В. )

Завідувач відділення \_\_\_\_\_ ( Краснокутська К.Г. )