

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Чорноморський національний університет  
імені Петра Могили  
Факультет комп'ютерних наук  
Кафедра інтелектуальних інформаційних систем**

**ДОПУЩЕНО ДО ЗАХИСТУ**

Завідувач кафедри інтелектуальних  
інформаційних систем, д-р техн. наук, проф.

\_\_\_\_\_ Ю. П. Кондратенко

« \_\_\_\_ » \_\_\_\_\_ 2024 р.

**КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА**

**АВТОМАТИЗОВАНА ІНФОРМАЦІЙНА СИСТЕМА  
ІМПЕРАТИВА СТАЛОГО РОЗВИТКУ**

Спеціальність 122 «Комп'ютерні науки»

**122 – КРБ – 402.22010224**

*Виконав студент 4-го курсу, групи 402*

\_\_\_\_\_ *І. С. Стрельченко*

«20» червня 2024 р.

*Керівник: д-р пед. наук, професор*

\_\_\_\_\_ *О. П. Мещанінов*

«20» червня 2024 р.

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Чорноморський національний університет ім. Петра Могили**  
**Факультет комп'ютерних наук**  
**Кафедра інтелектуальних інформаційних систем**

Рівень вищої освіти **бакалавр**  
Спеціальність **122 «Комп'ютерні науки»**  
*(шифр і назва)*  
Галузь знань **12 «Інформаційні технології»**  
*(шифр і назва)*

**ЗАТВЕРДЖУЮ**

Завідувач кафедри інтелектуальних  
інформаційних систем, д-р техн. наук, проф.  
\_\_\_\_\_ Ю. П. Кондратенко  
« \_\_\_\_ » \_\_\_\_\_ 2024 р.

**З А В Д А Н Н Я**  
**на виконання кваліфікаційної роботи**

Видано студенту групи 402 факультету комп'ютерних наук Стрельченку Івану Сергійовичу.

1. Тема кваліфікаційної роботи «Автоматизована інформаційна система визначення імператива сталого розвитку».

Керівник роботи Мещанінов Олександр Павлович, д-р пед. наук, професор.

Затв. наказом Ректора ЧНУ ім. Петра Могили від «28» грудня 2023 р. № 271

2. Строк представлення кваліфікаційної роботи студентом «21» червня 2024 р.

3. Вхідні (початкові) дані до роботи: розробка автоматизованої інформаційної системи з фронтенд і бекенд частинами для визначення імператива сталого розвитку; створення чату для можливості респондентами дискутувати; можливість респондентами доповнювати інформацію на відмінні погляди.

Очікуваний результат: повноцінна автоматизована інформаційна система для ознайомлення з імперативом сталого розвитку та його обговоренням.

4. Перелік питань, що підлягають розробці (зміст пояснювальної записки):

- огляд існуючих методів розробки вебзастосунку;
- наявність двох поглядів на сталий розвиток;

– можливість респондентами доповнювати інформацію на відмінні погляди;

– можливість респондентами дискутувати. Чат.

5. Перелік графічного матеріалу: презентація, **24** рисунки та **10** таблиць.

6. Завдання до спеціальної частини: «Освітлення та вентиляція приміщень, аналіз та планування евакуації»

7. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис
Спеціальна частина з охорони праці	Алексеева А. О., доцент кафедри екології	

Керівник роботи д-р пед. наук, проф. Мещанінов О. П.  
(наук. ступінь, вчене звання, прізвище та ініціали)

\_\_\_\_\_  
(підпис)

Завдання прийнято до виконання Стрельченко І. С.  
(прізвище та ініціали)

\_\_\_\_\_  
(підпис)

Дата видачі завдання « 14 » січня 2024 р.

**КАЛЕНДАРНИЙ ПЛАН**  
**виконання кваліфікаційної роботи**

Тема: Автоматизована інформаційна система визначення імператива сталого розвитку

№	Найменування роботи	Початок	Закінчення	Примітки
1	Подання заяви на затвердження теми та керівників КРБ	10.11.2023	15.11.2023	Виконано
2	Отримання завдання на виконання КРБ	10.01.2024	15.01.2024	Виконано
3	Складання календарного плану роботи на весь період виконання КРБ	16.01.2024	30.01.2024	Виконано
4	Отримання завдання на переддипломну практику	15.04.2024	29.04.2024	Виконано
5	Проходження переддипломної практики, збір та аналіз матеріалів до КРБ	29.04.2024	11.05.2024	Виконано
6	Розробка звіту з переддипломної практики	12.05.2024	15.05.2024	Виконано
7	Виконання КРБ: Аналіз сфери продажу екотоварів, огляд та аналіз існуючих вебзастосунків, розробка ПЗ	13.05.2024	22.06.2024	Виконано
8	Перший попередній захист КРБ на засіданні комісії кафедри	27.05.2024	27.05.2024	Виконано
9	Доробка та остаточне оформлення КРБ	28.05.2024	09.06.2024	Виконано
10	Другий попередній захист КРБ на засіданні комісії кафедри	10.06.2024	10.06.2024	Виконано
11	Подання КРБ рецензенту	13.06.2024	13.06.2024	Виконано
11	Подання КРБ, її електронної копії та інших документів (відгуку, рецензії) до захисту	17.06.2024	21.06.2024	
12	Захист КРБ перед екзаменаційною комісією (ЕК)	24.06.2024	28.06.2024	

Розробив студент Стрельченко І. С.  
*(прізвище, ім'я, по батькові студента)*

\_\_\_\_\_ *(підпис)*

Керівник роботи д-р пед. наук, проф. Мещанінов О. П.  
*(посада, прізвище, ім'я, по батькові)*

\_\_\_\_\_ *(підпис)*

« 29 » \_\_\_\_\_ 01 \_\_\_\_\_ 2024 р

## АНОТАЦІЯ

кваліфікаційної роботи студента групи 402 ЧНУ ім. Петра Могили  
Стрельченка Івана Сергійовича

Тема: «Автоматизована інформаційна система визначення імператива  
сталого розвитку»

**Актуальність теми** полягає у необхідності вирішення глобальних екологічних, економічних та соціальних проблем через принципи сталого розвитку. Автоматизовані інформаційні системи відіграють важливу роль у цьому процесі, забезпечуючи доступ до інформації, аналізу та прийнятті обґрунтованих рішень для досягнення сталого розвитку в умовах технологічного прогресу та цифрового прогресу.лізації.

**Об'єкт дослідження** – імператив сталого розвитку.

**Предмет дослідження** – методи і технології розробки автоматизованих інформаційних систем.

**Мета кваліфікаційної роботи** полягає у систематизації та аналізі концепцій та підходів до сталого розвитку з різних перспектив, таких як екологічна, економічна та соціальна. Це включає вивчення поглядів провідних авторів у цій області та їхніх внесків у розуміння проблематики сталого розвитку. Додатковою метою є розробка автоматизованої інформаційної системи, яка допоможе у визначенні імператива сталого розвитку та сприятиме обміну інформацією та дискусіям у цій сфері..

Пояснювальна записка складається зі вступу, трьох розділів, висновків, списку використаних джерел і додатків.

У першому розділі розглядається визначення та теоретичні аспекти імператива сталого розвитку.

У другому розділі описано проектування та реалізація поставленої задачі.

Третій розділ є рекомендаційний, в якому розглядається можливі покращення автоматизованої інформаційної системи.

В результаті розроблено автоматизовану інформаційну систему визначення імператива сталого розвитку.

Кваліфікаційна робота містить 55 сторінок, 34 рисунки, 1 таблиць, 30 використаних джерел та 0 додатків.

Ключові слова: автоматизована інформаційна система, методи і технології розробки АІС, моделі та теорії сталого розвитку, імператив сталого розвитку, концепція сталого розвитку.

## **ABSTRACT**

**to the qualification work by the student of group 402 of  
Petro Mohyla Black Sea National University**

**Strelchenko Ivan**

**« Automated Information System for Determining the Imperative of Sustainable  
Development »**

The relevance of the topic lies in the necessity to address global ecological, economic, and social problems through the principles of sustainable development. Automated information systems play an essential role in this process, providing access to information, analysis, and informed decision-making to achieve sustainable development in the context of technological and digital progress.

The object of the work is the imperative of sustainable development..

The subject of the research is the methods and technologies for developing automated information systems.

The purpose of the qualification work is to systematize and analyze concepts and approaches to sustainable development from various perspectives, such as ecological, economic, and social. This includes studying the views of leading authors in this field and their contributions to understanding the issues of sustainable development. An additional objective is to develop an automated information system that helps in determining the imperative of sustainable development and facilitates information exchange and discussions in this area. The explanatory note consists of an introduction, three chapters, conclusions, a list of references, and appendices.

The explanatory note consists of an introduction, three chapters, conclusion, and appendices.

In the first chapter, discusses the definition and theoretical aspects of the imperative of sustainable development.

In the second chapter describes the design and implementation of the task.

In the third chapter is a recommendation section, which considers possible improvements to the automated information system.

As a result, an automated information system for determining the imperative of sustainable development has been developed.

The qualification work contains 55 pages, 34 figures, 1 tables, 30 references, and 0 appendices.

**Keywords:** automated information system, methods and technologies for developing AIS, models and theories of sustainable development, imperative of sustainable development, concept of sustainable development.

## ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	3
ВСТУП.....	4
1 ВИЗНАЧЕННЯ ТА ТЕОРЕТИЧНІ АСПЕКТИ ІМПЕРАТИВА СТАЛОГО РОЗВИТКУ .....	6
1.1 Вступ до поняття сталого розвитку .....	6
1.2 Моделі та теорії сталого розвитку .....	8
1.3 Імператив сталого розвитку: сутність та значення .....	11
1.4 Автоматизована інформаційна система: сутність та значення .....	14
Висновки до розділу 1 .....	15
2 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ АВТОМАТИЗОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ .....	16
2.1 Аналіз та вибір методів і технологій розробки.....	16
2.2 Розробка сайту.....	18
2.3 Розробка чату та можливості редагувати.....	27
Висновки до розділу 2 .....	39
3 РЕКОМЕНДАЦІЇ ТА ПЕРСПЕКТИВИ ВИКОРИСТАННЯ АВТОМАТИЗОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	41
3.1 Оновлення та модернізація вебсайту.....	41
3.2 Використання нових технологій .....	46
3.3 Впровадження соціальних логінів .....	48
Висновки до розділу 3 .....	51
ВИСНОВКИ.....	53
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	54

## **ПЕРЕЛІК СКОРОЧЕНЬ**

- PHP – Hypertext Preprocessor
- HTML – Hypertext Markup Language
- JS – JavaScript



## ВСТУП

У сучасному світі людство стикається з безпрецедентною складністю екологічних, економічних та соціальних викликів. Забруднення навколишнього середовища, вичерпання природних ресурсів, економічна нерівність та соціальні конфлікти є лише деякими з проблем, що потребують негайного вирішення. Концепція сталого розвитку, яка передбачає збалансоване задоволення потреб нинішнього покоління без шкоди для можливостей майбутніх поколінь задовольняти свої власні потреби, стає ключовою для забезпечення довгострокового процвітання та гармонійного співіснування людини і природи. Сталий розвиток передбачає інтеграцію трьох основних компонентів: економічного зростання, соціальної справедливості та екологічної стійкості. Важливість цього підходу полягає у його здатності забезпечити збереження природних ресурсів, підтримання екологічного балансу та створення умов для справедливого економічного зростання і соціального добробуту. Однак, досягнення цієї мети вимагає нових підходів та інструментів, які можуть забезпечити ефективне управління ресурсами, аналіз даних та прийняття обґрунтованих рішень. Одним із таких інструментів є автоматизовані інформаційні системи.

Аналіз досліджень та публікацій. Для отримання глибшого розуміння теми сталого розвитку, було важливо провести аналіз попередніх досліджень та публікацій. У процесі роботи були використані праці провідних авторів, що дозволило оцінити поточний стан досліджень та здобуті знання у цій області. Серед цих авторів можна виокремити: Брауна Лестера, Мещанінова Олександра, Форестера Деніела, Медоуз Донелли, Фрідмена Мілтона, Дейлі Гермена, Мітчелла Джеймса, Сена Амартії, Нельсона Нела та Фока Френсіса.

Завдання дослідження. Поставлена мета обумовила необхідність вирішення наступних завдань:

- 1) проаналізувати основні концепції сталого розвитку з екологічної, економічної та соціальної перспектив;
- 2) вивчити роботи провідних авторів у галузі сталого розвитку, зокрема їхні погляди та внесок у розвиток цієї області;
- 3) оцінити поточний стан досліджень та знань щодо проблематики сталого розвитку, визначити ключові виклики та тенденції;
- 4) провести аналіз вимог до функціоналу та можливостей інформаційної системи для забезпечення оптимального обміну інформацією та дискусій у галузі сталого розвитку;
- 5) здійснити впровадження та валідацію розробленої інформаційної системи, оцінити її ефективність та корисність для цільової аудиторії.

# 1 ВИЗНАЧЕННЯ ТА ТЕОРЕТИЧНІ АСПЕКТИ ІМПЕРАТИВА СТАЛОГО РОЗВИТКУ

## 1.1 Вступ до поняття сталого розвитку

Перевизначення поняття сталого розвитку в контексті сучасних викликів та наукових досягнень є критичною складовою для розуміння сутності цього концепту. Сучасне тлумачення сталого розвитку виходить за межі простої гармонії між екологічними, економічними та соціальними аспектами, адже воно враховує не лише поточні потреби, а й інтереси майбутніх поколінь. Перевизначення поняття сталого розвитку передбачає перехід від концепції "зростання за будь-яку ціну" до "розвитку з урахуванням меж природних ресурсів та потреб людства". Це означає, що сталий розвиток має забезпечувати насамперед потреби сучасного суспільства, не ушкоджуючи можливостей майбутніх поколінь задовольняти свої потреби. Одним із ключових аспектів перевизначення поняття сталого розвитку є перехід від простого застосування концепції до практичної реалізації через розробку конкретних стратегій, політик та програм, спрямованих на досягнення сталого розвитку на всіх рівнях - від місцевого до глобального.

Історичний огляд концепції сталого розвитку є ключовим етапом для розуміння її еволюції та визначення головних тенденцій у цьому напрямку. Концепція сталого розвитку виникла внаслідок рефлексії над екологічними, економічними та соціальними проблемами, що виникли в другій половині 20-го століття. Перші згадки про необхідність збалансованого підходу до розвитку з'явилися ще в середньовіччі, коли суспільства відчували перші наслідки експлуатації природних ресурсів. Проте концепція сталого розвитку як науковий та практичний напрямок набула значного розвитку у другій половині 20-го століття. Одним з перших кроків у становленні концепції сталого розвитку була публікація доповіді Клубу Римського в 1972 році, в якій вперше висловлено занепокоєння щодо обмеженості природних ресурсів та погіршення стану навколишнього середовища. Пізніше ця доповідь увійшла в історію як "Доповідь

Міжнародного клубу Рима про межі зростання" або "Межі зростання", де вперше було використано термін "сталий розвиток". У наступні десятиліття концепція сталого розвитку набула все більшої популярності, зокрема під впливом діяльності ООН та проведення конференцій із збереження навколишнього середовища. Згодом стало очевидним, що сталий розвиток не може бути досягнутий лише за допомогою заходів, спрямованих на екологічний захист, а потребує комплексного підходу, що враховує економічні та соціальні аспекти. Розуміння історичних контекстів формування концепції сталого розвитку дозволяє краще оцінити її потенційні можливості та перспективи у майбутньому, а також надає базовий фундамент для подальших досліджень та розробки стратегій у цій області.

Ключові складові сталого розвитку: економічні, екологічні та соціальні аспекти. Сталий розвиток є концепцією, яка визначає стратегію розвитку, спрямовану на задоволення потреб сучасного суспільства, не зашкоджуючи при цьому можливостям майбутніх поколінь. Ця концепція виходить за рамки економічного зростання та враховує екологічні та соціальні аспекти.

*Економічні аспекти* сталого розвитку зосереджені на створенні умов для стабільного та сталого економічного зростання, яке забезпечує підвищення рівня життя населення без шкоди для навколишнього середовища та природних ресурсів. Це охоплює впровадження ефективних механізмів управління ресурсами, розвиток інноваційних технологій та бізнес-моделей, спрямованих на збереження ресурсів та зменшення викидів, а також стимулювання економічної активності, яка сприяє сталому розвитку.

*Екологічні аспекти* сталого розвитку передбачають збереження природних ресурсів та біорізноманіття, зменшення викидів та забруднення, а також створення умов для здорового та стійкого середовища проживання. Це охоплює заходи з охорони природи, використання відновлюваних джерел енергії, розвиток екологічно чистих технологій та зелених інфраструктурних проєктів.

*Соціальні аспекти* сталого розвитку спрямовані на забезпечення справедливості, рівності та добробуту для всіх членів суспільства. Це охоплює забезпечення доступу до освіти та охорони здоров'я, боротьбу з бідністю та нерівністю, створення умов для активної участі громадян у прийнятті рішень та підтримки прав людини. Особливе значення в цьому контексті має освіта, яка є ключовим фактором у формуванні свідомого, відповідального та активного громадянського суспільства. Освіта сприяє підвищенню рівня обізнаності населення про важливість сталого розвитку, допомагає розвивати навички та знання, необхідні для вирішення складних екологічних, економічних та соціальних проблем. Інвестиції в освіту забезпечують довгостроковий позитивний вплив, стимулюючи інновації, економічне зростання та соціальну згуртованість. Загальна мета сталого розвитку полягає в досягненні гармонії між економічними, екологічними та соціальними аспектами, що забезпечить стабільний та стійкий розвиток суспільства без шкоди для навколишнього середовища та майбутніх поколінь.

## **1.2 Моделі та теорії сталого розвитку**

Екологічний підхід до сталого розвитку базується на принципах збереження природних ресурсів, збалансованого використання екосистем та зменшення негативного впливу людської діяльності на навколишнє середовище. Цей підхід враховує взаємозв'язок між людиною та природою, спираючись на наукові дані та принципи екологічної стійкості.

Основні принципи екологічного підходу включають наступне:

1) збереження біорізноманіття. Це означає захист екосистем та різноманітності видів, що є важливим для стабільності екологічних систем та підтримки життєздатності екосистем;

2) ефективне використання ресурсів. Екологічний підхід спрямований на зменшення споживання ресурсів та мінімізацію відходів шляхом оптимізації виробничих процесів та використання відновлюваних джерел енергії;

3) зменшення викидів та забруднення. Цей принцип передбачає обмеження викидів та забруднення атмосфери, водних ресурсів та ґрунтів шляхом впровадження чистих технологій та зелених ініціатив;

4) сприяння відновленню природних екосистем. Екологічний підхід включає заходи з відновлення та охорони природних екосистем, що допомагають підтримувати екологічну стійкість та зберігати біорізноманіття;

5) створення екологічно чистих та здорових середовищ. Цей принцип передбачає розробку інфраструктури та технологій, спрямованих на забезпечення здорового та безпечного для життя середовища для всіх членів суспільства.

Екологічний підхід до сталого розвитку враховує важливість збереження природних ресурсів та екосистем для майбутніх поколінь, сприяючи створенню життєздатного та екологічно стійкого суспільства.

Економічний підхід до сталого розвитку спрямований на створення ефективної економіки, яка забезпечує підвищення благополуччя суспільства, не шкодячи природному середовищу та забезпечуючи його збереження для майбутніх поколінь. Цей підхід враховує важливість економічного зростання, але не за рахунок виснаження ресурсів чи пошкодження навколишнього середовища, а за допомогою інноваційних технологій, раціонального використання ресурсів та стимулювання стійкого розвитку.

Основні принципи економічного підходу до сталого розвитку включають:

1) ефективне використання ресурсів. Це передбачає оптимізацію виробничих процесів та раціональне використання ресурсів з максимальним економічним ефектом;

2) стимулювання інновацій. Економічний підхід до сталого розвитку сприяє розвитку інноваційних технологій та підтримує інноваційну активність у сферах, що сприяють сталому розвитку;

3) забезпечення справедливості та рівності. Це означає створення умов для рівного доступу до економічних можливостей та ресурсів для всіх членів суспільства;

4) підтримка сталого споживання та виробництва. Економічний підхід передбачає перехід до моделей споживання та виробництва, які не виснажують природні ресурси та не завдають шкоди навколишньому середовищу;

5) розвиток зеленої економіки. Це означає перехід до економіки, що базується на використанні відновлюваних джерел енергії, використанні чистих технологій та зелених ініціатив для зменшення викидів та забруднення.

Економічний підхід до сталого розвитку спрямований на створення економічно стійкого та соціально справедливого суспільства, що забезпечує збалансований розвиток та збереження природних ресурсів для майбутніх поколінь.

Соціальний підхід до сталого розвитку акцентує увагу, в першу чергу, на розвиток системи освіти, а також на гармонії між соціальними, культурними та економічними аспектами суспільства з метою забезпечення всебічного розвитку та підвищення якості життя всіх його членів. Цей підхід враховує потреби та інтереси людей, їхню культурну та соціальну ідентичність, а також створює умови для їхнього розвитку та самореалізації.

Основні принципи соціального підходу до сталого розвитку включають:

1) забезпечення соціальної справедливості. Це передбачає створення умов для всебічного розвитку всіх верств суспільства, уникнення виникнення соціальних нерівностей та дискримінації;

2) забезпечення освіти та культурного розвитку. Освіта є найважливішим аспектом соціального підходу, оскільки вона виступає рушієм до переходу до сталого розвитку. Якісна освіта забезпечує інформованість населення про екологічні, економічні та соціальні аспекти сталого розвитку, сприяє формуванню свідомого та активного громадянського суспільства. Розвиток освіти, культури та мистецтва є ключовими складовими гармонійного розвитку суспільства.

3) створення сприятливих умов для здоров'я та добробуту. Це включає доступ до якісної медичної допомоги, житла, безпеки та інфраструктури, які сприяють здоровому та щасливому життю;

4) забезпечення участі громадськості та демократії. Соціальний підхід передбачає активну участь громадськості у процесах прийняття рішень та розвитку суспільства;

5) підтримка гендерної рівності та прав людини. Це означає визнання прав кожної людини незалежно від її статі, раси, віросповідання чи інших ознак.

Соціальний підхід до сталого розвитку спрямований на створення справедливого, гуманного та гармонійного суспільства, в якому всі його члени можуть реалізувати свій потенціал та жити щасливо і безпечно.

### **1.3 Імператив сталого розвитку: сутність та значення**

Поняття імперативу сталого розвитку. Імператив сталого розвитку представляє собою принцип або необхідність досягнення балансу між соціальними, економічними та екологічними аспектами людської діяльності з метою забезпечення як поточних, так і майбутніх поколінь.

Визначення поняття імперативу сталого розвитку. Імператив сталого розвитку визначається як стратегічний пріоритет для суспільства, спрямований на збереження природних ресурсів та екосистем, забезпечення соціальної справедливості та підвищення економічного добробуту на сучасному етапі розвитку та для майбутніх поколінь.

Ключові аспекти імперативу сталого розвитку включають екологічну, економічну та соціальну складові. Екологічний аспект передбачає збереження природних ресурсів та біорізноманіття. Економічний аспект включає раціональне використання ресурсів та забезпечення сталого економічного зростання. Соціальний аспект орієнтований на забезпечення соціальної справедливості, рівності та доступу до основних потреб для всіх членів суспільства.



Значення імперативу сталого розвитку. Імператив сталого розвитку має величезне значення для сучасного світу, оскільки він допомагає забезпечити баланс між економічними, екологічними та соціальними аспектами розвитку. Це важливо для збереження природних ресурсів, забезпечення життєздатності екосистем, підтримки гідного життя для всіх людей та мінімізації негативного впливу на клімат та довкілля.

Підходи до реалізації імперативу сталого розвитку. Реалізація імперативу сталого розвитку вимагає комплексного підходу та спільних зусиль усіх сфер суспільства. Серед ключових підходів до реалізації можна виділити створення ефективної екологічної політики, сприяння інноваційним технологіям, розвиток альтернативних джерел енергії та підвищення екологічної свідомості суспільства.

Незважаючи на важливість імперативу сталого розвитку, його впровадження не завжди є простим завданням. Серед основних викликів можна виділити:

1) економічні обмеження. Багато заходів, необхідних для досягнення сталого розвитку, можуть вимагати значних фінансових витрат, що ставить під сумнів готовність бізнесу та урядових структур інвестувати в ці цілі;

2) політична нестабільність. Впровадження стратегій сталого розвитку часто потребує довгострокових планів, які можуть бути піддані ризику через політичні зміни та недостатню підтримку з боку урядових структур;

3) відсутність усвідомлення. Не всі члени суспільства, включаючи бізнес та громадськість, можуть розуміти або приймати необхідність прийняття заходів для сталого розвитку, що ускладнює масштабну мобілізацію та підтримку;

4) культурні та соціальні відмінності. Реалізація стратегій сталого розвитку може зіткнутися з різними культурними та соціальними контекстами, що потребує індивідуального підходу та врахування місцевих особливостей.

Перехрестя між цими викликами вимагає комплексного підходу та стратегій, спрямованих на подолання перешкод та забезпечення успішного впровадження імперативу сталого розвитку.

Роль імперативу сталого розвитку в сучасному світі не може бути переоцінена через її визначний вплив на економічний, екологічний та соціальний ландшафт суспільства. Вона виступає як керівний принцип, що спрямовує дії урядових органів, бізнесу та громадян у напрямку сталого розвитку. Реалізація імперативу сталого розвитку допомагає забезпечити екологічну стійкість, збереження природних ресурсів та біорізноманіття, що є ключовими складовими здоров'я планети. Крім того, імператив сприяє підвищенню якості життя людей, забезпеченню соціальної справедливості та рівності, а також стимулює інновації та розвиток нових технологій, що сприяють ефективному використанню ресурсів та зменшенню викидів. У сучасному світі, де проблеми забруднення навколишнього середовища, зміна клімату та соціальна нерівність є актуальними, імператив сталого розвитку стає не лише моральним імперативом, але й стратегічним пріоритетом для всього суспільства.

Взаємозв'язок імперативу сталого розвитку з економічними, екологічними та соціальними аспектами є ключовим для досягнення збалансованого та стійкого розвитку. Економічний аспект імперативу сталого розвитку полягає в створенні стійкої економіки, яка базується на ефективному використанні ресурсів та інноваціях. Це сприяє забезпеченню економічного зростання, підвищенню рівня життя та зменшенню економічної нерівності. Екологічний аспект імперативу сталого розвитку передбачає збереження природних ресурсів, охорону довкілля та зменшення викидів забруднюючих речовин. Це важливо для забезпечення життєздатності екосистем, збереження біорізноманіття та мінімізації негативного впливу на клімат. Соціальний аспект імперативу сталого розвитку орієнтований на забезпечення соціальної справедливості, рівності та доступу до основних потреб для всіх членів суспільства. Це включає в себе забезпечення доступу до освіти, охорони здоров'я, житла та розвитку інфраструктури. Взаємодія цих аспектів утворює цілісний підхід до сталого розвитку, що сприяє забезпеченню гармонії між економічними, екологічними та соціальними цілями суспільства.

## 1.4 Автоматизована інформаційна система: сутність та значення

Автоматизована інформаційна система— це комплекс програмних та апаратних засобів, призначених для збору, обробки, зберігання, передачі та відображення інформації. Основною метою таких систем є забезпечення ефективного управління інформаційними потоками, що дозволяє спростити та прискорити процеси обробки даних. Автоматизована інформаційна система, спрямована на аналіз та тлумачення імперативу сталого розвитку, є надзвичайно важливим інструментом у сучасному світі. Така система може інтегрувати та аналізувати інформацію з різних джерел, надаючи користувачам можливість отримувати всебічне розуміння концепції сталого розвитку з різних точок зору. Однією з головних функцій такої автоматизованої системи є збір та інтеграція різних тлумачень імперативу сталого розвитку, що надаються різними джерелами. Наприклад, в одному дослідженні може бути акцент на економічні аспекти, тоді як в іншому — на екологічні або соціальні. Система дозволяє зібрати всі ці різні точки зору, об'єднати їх та представити в зручній для користувача формі.

Автоматизована інформаційна система надає можливість не тільки збирати, але й аналізувати дані, що дозволяє проводити порівняння різних підходів та тлумачень імперативу сталого розвитку. Це сприяє більш глибокому розумінню різних аспектів сталого розвитку та допомагає виявити найбільш ефективні стратегії та практики.

Важливою функцією автоматизованої інформаційної системи є візуалізація інформації. Користувачі можуть легко зрозуміти складні дані та побачити взаємозв'язки між різними аспектами сталого розвитку. Це сприяє більш легкому засвоєнню інформації та її аналізу.

Такі системи також відіграють ключову роль у підвищенні обізнаності та освіти громадськості щодо сталого розвитку. Вони можуть надавати доступ до освітніх матеріалів, дослідницьких звітів та інших ресурсів, що сприяють формуванню екологічної свідомості та відповідального ставлення до навколишнього середовища.

**Висновки до розділу 1**

У результаті дослідження було виявлено, що поняття імперативу сталого розвитку є важливим принципом для сучасного світу. Імператив сталого розвитку передбачає збалансоване поєднання економічних, екологічних та соціальних аспектів розвитку суспільства з метою забезпечення гармонійного функціонування нашої планети та підтримки благополуччя майбутніх поколінь. Важливо враховувати, що імператив сталого розвитку відображає не лише необхідність збереження природних ресурсів та охорони навколишнього середовища, але й сприяє створенню справедливого та економічно стабільного суспільства. Такий підхід вимагає системності та комплексного підходу до вирішення проблем сталого розвитку, але забезпечує перспективи для стійкого та гармонійного розвитку як нашого суспільства, так і планети в цілому.

## 2 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ АВТОМАТИЗОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ

### 2.1 Аналіз та вибір методів і технологій розробки

Розробка веб-сайту вимагає ретельного вибору методів і технологій, які будуть використовуватися для створення як фронтенд, так і бекенд частин проекту. Вибір цих технологій визначається низкою факторів, включаючи вимоги проекту, доступні ресурси, досвід команди розробників та очікування щодо продуктивності та масштабованості системи. У цьому розділі ми розглянемо можливі технології для розробки фронтенду та бекенду, а також обґрунтуємо вибір технологій для мого проекту.

Фронтенд: HTML, CSS та JavaScript. HTML є основною мовою розмітки для створення веб-сторінок. Вона використовується для структурування контенту на веб-сторінці. Використання HTML дозволяє розробникам визначати заголовки, абзаци, посилання, зображення та інші елементи веб-сторінки. HTML є стандартом, підтримуваним усіма браузерами, що робить його незамінним інструментом для будь-якого веб-розробника. CSS використовується для стилізації HTML-елементів, що дозволяє контролювати зовнішній вигляд веб-сторінки. За допомогою CSS можна задавати кольори, шрифти, відступи, розташування елементів та багато інших параметрів. Це дозволяє розділити структуру та стилізацію, роблячи код більш організованим та легким для підтримки. JavaScript — це мова програмування, яка дозволяє додавати інтерактивність до веб-сторінок. Вона використовується для створення динамічного контенту, такого як випадаючі меню, слайдери, валідація форм та інші інтерактивні елементи. JavaScript підтримується всіма сучасними браузерами і має велике число бібліотек та фреймворків, таких як jQuery, React та Vue.js, що значно полегшує розробку складних інтерфейсів.

HTML, CSS та JavaScript були обрані для розробки фронтенду нашого веб-сайту через їхню універсальність, широке розповсюдження та підтримку. Ці

технології дозволяють створити зручний та привабливий інтерфейс для користувачів, забезпечуючи при цьому високу сумісність з різними браузерами та пристроями.

PHP — це серверна мова програмування, яка широко використовується для розробки веб-додатків. Вона інтегрується з HTML, що робить її ідеальною для створення динамічних веб-сторінок. PHP є відкритим програмним забезпеченням, що означає, що воно безкоштовне для використання та має велику спільноту розробників, яка постійно підтримує та вдосконалює мову. PHP має багато вбудованих функцій для роботи з базами даних, обробки форм, управління сесіями та іншими важливими завданнями, що робить його потужним інструментом для створення серверної частини веб-додатків. Він також підтримує різні фреймворки, такі як Laravel, Symfony та CodeIgniter, які спрощують та прискорюють розробку складних проектів.

Альтернативні технології для бекенду Node.js. Node.js — це середовище виконання JavaScript на сервері, яке дозволяє використовувати JavaScript для розробки бекенд частини веб-додатків. Node.js має асинхронну архітектуру, що робить його високоефективним для обробки великої кількості одночасних запитів. Він також має великий вибір пакетів та модулів, доступних через npm (Node Package Manager).

Python (Django, Flask) Python є ще однією популярною мовою програмування для бекенду, відомою своєю простотою та читабельністю коду. Фреймворки Django та Flask забезпечують швидкий розвиток веб-додатків, пропонуючи багато вбудованих функцій та розширень. Django, зокрема, має високу продуктивність та безпеку, що робить його популярним вибором для багатьох розробників.

Ruby on Rails — це веб-фреймворк, написаний на мові програмування Ruby. Він відомий своєю філософією "конвенція над конфігурацією", що дозволяє розробникам швидко створювати веб-додатки з мінімальними налаштуваннями. Rails забезпечує високу продуктивність та багатий набір вбудованих інструментів для роботи з базами даних, маршрутизації, відображенням та іншими завданнями.

Проаналізувавши вище згадані мови програмування для розробки бекенд частини веб-сайту був обраний PHP з кількох причин:

1) простота інтеграції з HTML: PHP легко інтегрується з HTML, що дозволяє створювати динамічні веб-сторінки без необхідності використання додаткових мов програмування;

2) широке розповсюдження: PHP є однією з найпопулярніших серверних мов програмування, що забезпечує велику кількість ресурсів, документації та підтримки від спільноти розробників;

3) вбудовані функції: PHP має багато вбудованих функцій для роботи з базами даних, обробки форм, управління сесіями та іншими завданнями, що робить його потужним інструментом для розробки веб-додатків;

4) підтримка фреймворків: PHP підтримує різні фреймворки, такі як Laravel, Symfony та CodeIgniter, що спрощують та прискорюють розробку складних проєктів;

5) висока продуктивність: PHP забезпечує високу продуктивність та надійність, що важливо для створення масштабованих та ефективних веб-додатків.

Вибір HTML, CSS та JavaScript для фронтенду, а також PHP для бекенду для моєї задачі є обґрунтованим рішенням, яке враховує всі основні вимоги до проєкту.

## **2.2 Розробка сайту**

### **2.2.1 Розробка шапки сайту**

Шапка сайту (header) є одним з найважливіших елементів веб-сторінки, оскільки вона зазвичай містить логотип, навігаційне меню та інші важливі елементи, які допомагають користувачам орієнтуватися на сайті.

У цьому розділі буде розглянуто код, який відповідає за створення шапки сайту, а також пояснення функції кожного елемента та CSS-стилі, які забезпечують його візуальне оформлення.

Спочатку було створено основний контейнер шапки.

Ці теги визначають основний контейнер для шапки сайту. Атрибут `id="masthead"` задає унікальний ідентифікатор для цього елемента, а клас `site-header` використовується для стилізації. Внутрішній `div` з класом `main-header sticky-desktop` робить шапку "липкою", тобто вона залишатиметься на верхній частині сторінки під час прокручування (рис. 2.1).

```
<header id="masthead" class="site-header">  
<div id="cshero-header" class="header-1 main-header sticky-desktop">
```

Рисунок 2.1 – Код шапки сайту

Ці теги визначають основний контейнер для шапки сайту. Атрибут `id="masthead"` задає унікальний ідентифікатор для цього елемента, а клас `site-header` використовується для стилізації. Внутрішній `div` з класом `main-header sticky-desktop` робить шапку "липкою", тобто вона залишатиметься на верхній частині сторінки під час прокручування (рис. 2.2-2.3).



Рисунок 2.2 – Шапка сайту



Рисунок 2.3 – Шапка сайту під час прокручування

Логотип сайту (рис. 2.5) і блок "Сталий розвиток". Цей блок містить логотип сайту та текст "Сталий розвиток". `div` з класом `site-branding` містить логотип, який є посиланням на верхню частину сторінку сайту. Клас `header-caption` стилізує текст "Сталий розвиток" (рис. 2.4).



```

<div class="logo-and-caption-container">
<!-- Логотип -->
<div id="cshero-header-logo" class="site-branding col-xs-4 col-sm-6 col-md-3 col-lg-2">
<div class="main_logo hidden-xs"><a href="index.html"></a></div>
</div>
<!-- Блок "Сталий розвиток" -->
<div class="header-caption">Сталий розвиток</div>
</div>

```

Рисунок 2.4 – Код логотипу сайту



Рисунок 2.5 – Логотип сайту

Далі було створенно навігаційне меню (рис. 2.6-2.7).

```

<div id="cshero-header-navigation" class="col-xs-12 col-sm-12 col-md-6 col-lg-7 header-navigation">
<nav id="site-navigation" class="main-navigation">
<div class="menu-primary-menu-container">
<ul id="menu-primary-menu" class="nav-menu menu-main-menu">
<li id="menu-item-980" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-has-children menu-item-980">
<a href="center/index.htm">Мої роботи</a>
</li>
<li id="menu-item-1089" class="menu-item menu-item-type-custom menu-item-object-custom menu-item-has-children menu-item-1089">Чат</a>
</li>
<li id="menu-item-696" class="menu-item menu-item-type-custom menu-item-object-custom menu-item-has-children menu-item-696">
<a href="faq/index.htm">Інформація про мене</a></li>
<li id="menu-item-700" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-700">
<a href="contacts/index.htm">Особистий кабінет</a></li>
</ul>
</div>
</nav>
</div>

```

Рисунок 2.6 – Код навігаційного меню сайту

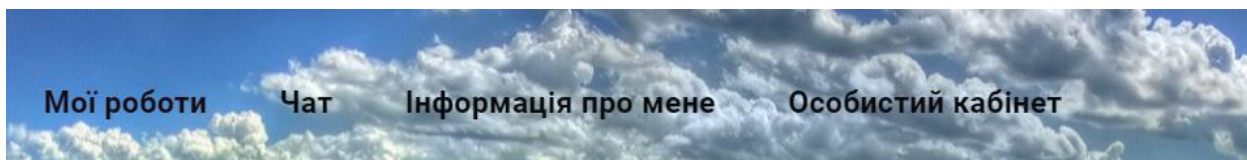


Рисунок 2.7 – Навігаційне меню

Також правіше від навігаційного меню була додана кнопка підтримки (рис. 2.8-2.9).

```
<a href="help/index.html" class="support-button">Підтримка</a>
```

Рисунок 2.8 – Код кнопки підтримки



Рисунок 2.9 – Кнопка підтримки

Шапка сайту відіграє ключову роль в забезпеченні зручності користування та навігації. Використання HTML, CSS та методів Flexbox дозволяє створити структурований та естетично привабливий інтерфейс. Включення логотипу, текстового блоку, навігаційного меню та кнопки підтримки забезпечує користувачам легкий доступ до основних розділів сайту. Такий підхід гарантує, що користувачі зможуть швидко знайти потрібну інформацію та з легкістю орієнтуватися на сайті.

### 2.2.2 Розробка основної частини

Основна частина сайту є важливим елементом веб-сторінки, оскільки вона містить основний контент, який призначений для інформування, залучення та взаємодії з користувачами. Спочатку ми зустрічаємося з поняттям імператива сталого розвитку. Аби дати користувачу базове розуміння про «Сталий розвиток» (рис. 2.11).

Рядок `<div class="content-container" style="margin-top: 450px;">` створює елемент з класом `content-container`. Крім того, для цього елемента встановлюється стиль `margin-top: 450px;`, що означає, що верхній відступ буде складати 450 пікселів. Це відсуває вміст контейнера вниз на 450 пікселів від верхньої межі свого батьківського елемента (рис. 2.10-2.11).

```
<div class="content-container" style="margin-top: 450px;">
<h3>Поняття імперативу сталого розвитку</h3>
<p>Імператив сталого розвитку представляє собою принцип або необхідність досягнення балансу між соціальними, економічними та екологічними аспектами людської діяльності з метою забезпечення як поточних, так і майбутніх поколінь. </p>
</div>
```

Рисунок 2.10 – Код основної частини сайту

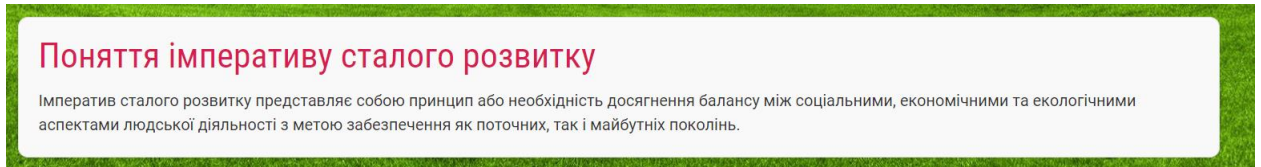


Рисунок 2.11 – Поняття імперативу сталого розвитку

Далі було створено контейнери в яких помістилися різні точки зору про імперативи сталого розвитку, а саме про екологічний, соціальний та економічний імперативи. Аналогічно коду нище (рис. 2.12) було створено ще 2 контейнери.

```
<div class="wpb_column vc_column_container vc_col-sm-4">
<div class="vc_column-inner">
<div class="wpb_wrapper">
<div class="fancy-box-3" id="cms-fancy-box-single-1">
<a>
<div class="fancy-image">

</div>
</a>
<div class="fancy-content">
<h4 style="color:#d81e50">
<a>Екологічний імператив</a>
</h4>
<p>Збережи і віднови природу для майбутніх поколінь</p>
</div>
</div></div></div>
```

Рисунок 2.12 – Код основної частини

Цей блок (рис. 2.13) надає коротку інформацію про імперативи. Він містить візуальний елемент, який привертає увагу і підсилює сприйняття інформації. Заголовок виділений червоним кольором, який вказує на важливість теми. Також внизу є короткий опис, який коротко і влучно пояснює суть кожного імперативу.

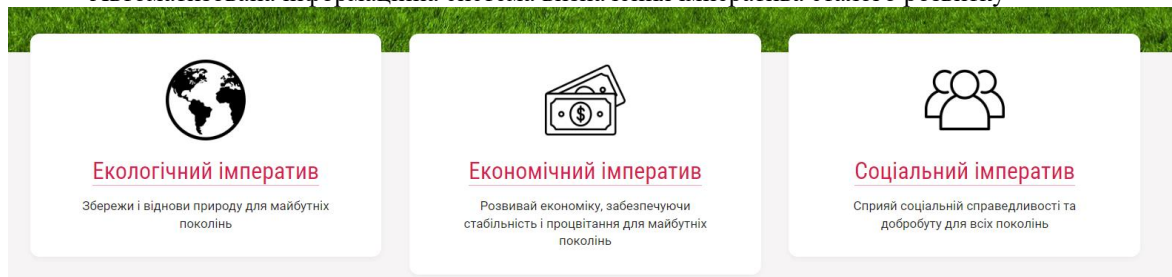


Рисунок 2.13 – Код основної частини

Далі йде контейнер який містить детальну інформацію про кожен імператив окремо (рис.2.14, 2.15, 2.16). Також внизу контейнера є кнопка «Редагувати», яка в майбутньому надасть можливість респондентам доповнювати або змінювати інформацію.

The image shows a detailed view of the 'Ecological Imperative' section. It features a green background on the left with the title 'Екологічний імператив' and a brief description. On the right, there is a white box containing a list of five bullet points. At the bottom left, there is a pink button labeled 'Редагувати' with a pencil icon.

## Екологічний імператив

Екологічний підхід до сталого розвитку базується на принципах збереження природних ресурсів, збалансованого використання екосистем та зменшення негативного впливу людської діяльності на навколишнє середовище.

- **Збереження біорізноманіття.** Це означає захист екосистем та різноманітності видів, що є важливим для стабільності екологічних систем та підтримки життєздатності екосистем.
- **Ефективне використання ресурсів.** Екологічний підхід спрямований на зменшення споживання ресурсів та мінімізацію відходів шляхом оптимізації виробничих процесів та використання відновлюваних джерел енергії.
- **Зменшення викидів та забруднення.** Цей принцип передбачає обмеження викидів та забруднення атмосфери, водних ресурсів та ґрунтів шляхом впровадження чистих технологій та зелених ініціатив.
- **Сприяння відновленню природних екосистем.** Екологічний підхід включає заходи з відновлення та охорони природних екосистем, що допомагають підтримувати екологічну стійкість та зберігати біорізноманіття.
- **Створення екологічно чистих та здорових середовищ.** Цей принцип передбачає розробку інфраструктури та технологій, спрямованих на забезпечення здорового та безпечного для життя середовища для всіх членів суспільства.

Редагувати

Рисунок 2.14 – Екологічний імператив

**Економічний імператив**

Економічний підхід до сталого розвитку спрямований на створення ефективної економіки, яка забезпечує підвищення благополуччя суспільства, не шкодячи природному середовищу та забезпечуючи його збереження для майбутніх поколінь.

Редагувати

- **Ефективне використання ресурсів.** Це передбачає оптимізацію виробничих процесів та раціональне використання ресурсів з максимальним економічним ефектом.
- **Стимулювання інновацій.** Економічний підхід до сталого розвитку сприяє розвитку інноваційних технологій та підтримує інноваційну активність у сферах, що сприяють сталому розвитку.
- **Забезпечення справедливості та рівності.** Це означає створення умов для рівного доступу до економічних можливостей та ресурсів для всіх членів суспільства.
- **Підтримка сталого споживання та виробництва.** Економічний підхід передбачає перехід до моделей споживання та виробництва, які не виснажують природні ресурси та не завдають шкоди навколишньому середовищу.
- **Розвиток зеленої економіки.** Це означає перехід до економіки, що базується на використанні відновлюваних джерел енергії, використанні чистих технологій та зелених ініціатив для зменшення викидів та забруднення.

Рисунок 2.15 – Економічний імператив

**Соціальний імператив**

Соціальний підхід до сталого розвитку акцентує увагу на гармонії між соціальними, культурними та економічними аспектами суспільства з метою забезпечення всебічного розвитку та підвищення якості життя всіх його членів.

Редагувати

- **Забезпечення соціальної справедливості.** Це передбачає створення умов для всебічного розвитку всіх верств суспільства, уникнення виникнення соціальних нерівностей та дискримінації.
- **Забезпечення освіти та культурного розвитку.** Освіта є найважливішим аспектом соціального підходу, оскільки вона виступає рушієм до переходу до сталого розвитку. Якісна освіта забезпечує інформованість населення про екологічні, економічні та соціальні аспекти сталого розвитку, сприяє формуванню свідомого та активного громадянського суспільства. Розвиток освіти, культури та мистецтва є ключовими складовими гармонійного розвитку суспільства.
- **Створення сприятливих умов для здоров'я та добробуту.** Це включає доступ до якісної медичної допомоги, житла, безпеки та інфраструктури, які сприяють здоровому та щасливому життю.
- **Забезпечення участі громадськості та демократії.** Соціальний підхід передбачає активну участь громадськості у процесах прийняття рішень та розвитку суспільства.
- **Підтримка гендерної рівності та прав людини.** Це означає визнання прав кожної людини незалежно від її статі, раси, віросповідання чи інших ознак.

Рисунок 2.16 – Соціальний імператив

Потім слідує розділ, що містить цитати відомих вчених та дослідників, чії праці стали основою для мого дослідження даної теми (рис. 2.17-2.18). В цьому розділі наведені ключові думки та висновки цих науковців, що підтверджують і доповнюють мої власні висновки. Матеріали цих авторитетних джерел

Автоматизована інформаційна система визначення імператива сталого розвитку забезпечують глибоке розуміння питання та слугують вагомим обґрунтуванням для розглядуваної проблеми сталого розвитку. Їхні роботи допомогли сформувати теоретичну базу, на якій базується моє дослідження, та надали цінні інсайти щодо різних аспектів даної теми.

```
<div class="scientist">
<p class="name">Фрідмен Мілтон</p>

<p class="quote">"Немає такої речі, як безкоштовний обід. Людям..</p></div>
```

Рисунок 2.17 – Код розділу з вченими

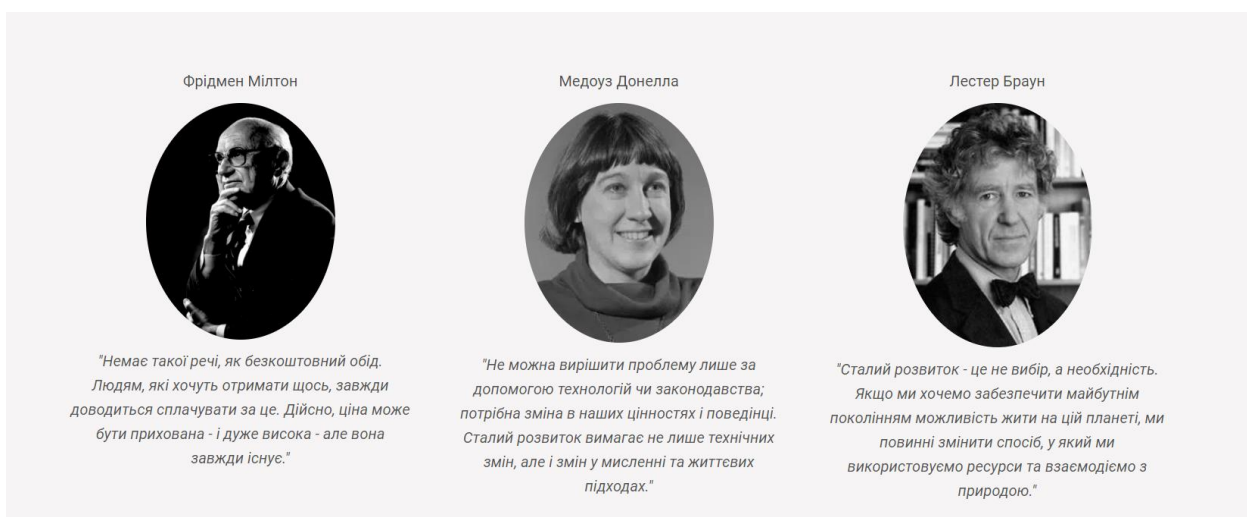


Рисунок 2.18 – Вчені та їх цитати

В найнижчій частині тега<body> ми можемо бачити висновки, які зробили після перегляду всього сайту (рис. 2.19). А саме, важливість імперативу сталого розвитку у сьогоденні, для теперішніх і майбутніх поколінь.

## Висновок

Сталий розвиток є критично важливим для забезпечення довгострокового благополуччя нашої планети і людства. У сучасному світі, де природні ресурси виснажуються, екосистеми деградують, а соціальна нерівність посилюється, необхідність прийняття принципів сталого розвитку стає все більш очевидною. Він забезпечує баланс між економічним зростанням, соціальною справедливістю та збереженням навколишнього середовища. Сталий розвиток сприяє ефективному використанню ресурсів, інноваціям і розвитку зелених технологій, що не лише підтримує економіку, але й знижує негативний вплив на довкілля. Він також акцентує увагу на соціальній справедливості, забезпечуючи рівні можливості для всіх верств населення, підтримуючи культурний і освітній розвиток, а також здоров'я та добробут громадян. Інтеграція принципів сталого розвитку у всі сфери життя дозволяє нам створити більш стійке і гармонійне суспільство, яке може задовольняти потреби нинішніх поколінь, не ставлячи під загрозу здатність майбутніх поколінь задовольняти свої потреби. Вибір сталого розвитку – це вибір на користь життя, процвітання та гармонії як для нинішніх, так і для майбутніх поколінь.

### Рисунок 2.19 – Висновки

#### 2.2.3 Розробка футеру

Футер - це нижня частина веб-сторінки, яка містить важливу інформацію і навігаційні елементи (рис. 2.20). Хоча він знаходиться внизу сторінки, його значення не можна недооцінити. Ось ключові складові футера:

- 1) контактна інформація. А саме: адреса, телефон, електронна пошта;
- 2) посилання на соціальні мережі, іконки та посилання на профілі в соціальних мережах;
- 3) посилання на важливі сторінки. Навігаційні посилання на сторінки "Про нас", "Контакти", "Правила використання", "Політика конфіденційності" тощо;
- 4) авторські права. Інформація про права на контент та дизайн сторінки.

+380 (96) 223-25-45  
[imperativ.stalogo.rozvitky@gmail.com](mailto:imperativ.stalogo.rozvitky@gmail.com)

Екологічний

Економічний

Соціальний

FAQ

► Telegram

► YouTube



Імператив сталого розвитку

[Політика конфіденційності](#)

Створення сайту – StreL1a

### Рисунок 2.20 – Футер сайту

### 2.3 Розробка чату та можливості редагувати інформацію

Для створення чату, необхідно розробити інтерфейс користувача та серверний обробник. Щоб користувачі могли обмінюватися повідомленнями, потрібно зберігати їхні імена, повідомлення та час відправлення. Для цього ми підключимо базу даних. У даному випадку будемо використовувати MySQL та phpMyAdmin.

**MySQL** — це одна з найпопулярніших систем керування реляційними базами даних (RDBMS - Relational Database Management System) з відкритим кодом. Вона розроблена і підтримується компанією Oracle Corporation. Мшироко використовується для зберігання, організації та керування даними у веб-застосунках та іншому програмному забезпеченні.

Основні особливості MySQL включають:

1) відкритий код. MySQL є безкоштовною для використання, і її вихідний код доступний для перегляду та модифікації. Це забезпечує гнучкість у налаштуванні та можливість адаптувати систему під конкретні потреби користувачів;

2) реляційна модель даних. MySQL підтримує реляційну модель даних, в якій дані організовані у вигляді таблиць. Таблиці складаються з рядків і стовпців, а також можуть бути пов'язані між собою за допомогою ключів. Це забезпечує структуроване та організоване зберігання даних, що полегшує їхню обробку та маніпулювання;

3) підтримка транзакцій. MySQL підтримує транзакції з ACID-властивостями (Atomicity, Consistency, Isolation, Durability), що важливо для забезпечення надійності та цілісності даних. Це особливо корисно для критично важливих застосунків, де будь-яка втрата даних є неприйнятною;

4) масштабованість і продуктивність. MySQL здатний обробляти великі обсяги даних та високу кількість одночасних запитів, що робить його придатним для великих і складних веб-застосунків;



5) **Безпека.** MySQL забезпечує потужні механізми безпеки для контролю доступу та шифрування даних, що дозволяє захистити інформацію від несанкціонованого доступу.

**phpMyAdmin** — це веб-додаток з відкритим кодом, який надає графічний інтерфейс для адміністрування MySQL баз даних. Він написаний на PHP і дозволяє користувачам легко управляти базами даних через веб-браузер. Основні можливості та переваги phpMyAdmin включають:

1) зручний графічний інтерфейс. phpMyAdmin надає інтуїтивно зрозумілий інтерфейс, який дозволяє користувачам взаємодіяти з базами даних без необхідності писати SQL-запити вручну. Це значно спрощує процес управління базами даних, особливо для користувачів, які не мають глибоких знань у SQL;

2) створення та управління базами даних. З phpMyAdmin користувачі можуть легко створювати, редагувати та видаляти бази даних, таблиці, поля, індекси та користувачів. Це дозволяє швидко налаштувати структуру бази даних та здійснювати необхідні зміни;

3) виконання SQL-запитів. Інструмент дозволяє виконувати довільні SQL-запити та переглядати їх результати. Це корисно для виконання складних операцій, які не можна виконати через графічний інтерфейс;

4) імпорт та експорт даних. phpMyAdmin підтримує імпорт та експорт баз даних у різних форматах, таких як SQL, CSV, XML тощо. Це полегшує резервне копіювання даних та їхнє перенесення між різними системами;

5) керування правами доступу. phpMyAdmin дозволяє налаштовувати права доступу користувачів до баз даних, забезпечуючи контроль над тим, хто і що може робити з даними;

6) моніторинг. Інструмент надає інформацію про статус сервера MySQL, продуктивність, використання ресурсів та інші важливі метрики, що допомагає в управлінні та оптимізації баз даних.

Першим кроком у створенні чату є розробка бази даних для збереження повідомлень. Використовуючи MySQL та phpMyAdmin, ми зможемо легко

Автоматизована інформаційна система визначення імператива сталого розвитку створити необхідну структуру бази даних і налаштувати всі потрібні таблиці та поля.

Створимо базу даних з назвою "chat " (рис. 2.21).

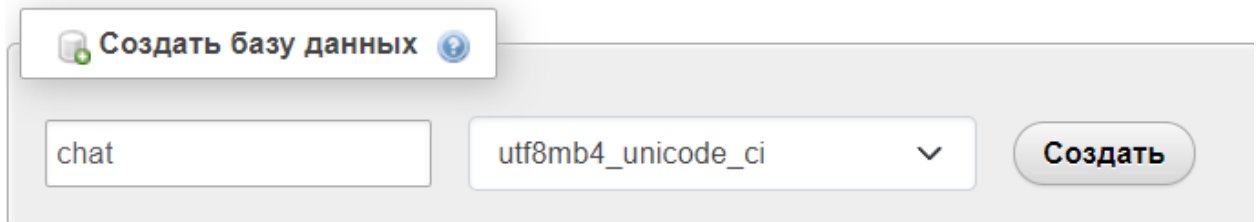


Рисунок 2.21 – Створення бази даних

Далі у базі даних "chat" створюємо таблицю "messages" (рис. 2.22) з наступними полями: "id", "username", "message", "timestamp". Така структура дозволить зберігати імена користувачів, їхні повідомлення та час відправки кожного повідомлення.

```
1 CREATE TABLE messages (  
2     id INT AUTO_INCREMENT PRIMARY KEY,  
3     username VARCHAR(50),  
4     message TEXT,  
5     timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
6 );  
7
```

Рисунок 2.22 – Створення таблиці в базі даних

Потім використовуючи HTML та PHP, ми створюємо форми для введення повідомлень, їх обробки та збереження у базі даних. Також ми зможемо відображати всі повідомлення на веб-сторінці, забезпечуючи таким чином повноцінний чат.

Далі ми маємо створити підключення до бази даних яке будемо використовувати в подальшому для можливості оперувати нею (рис 2.23).

```
$servername = "localhost";
```

```
$username = "root";
```

```
$password = "";
```

```
$databasename = "chat";
```

Рядки вище визначають параметри підключення до бази даних, а саме ім'я сервера, тут використовується локальний сервер. Потім задається ім'я користувача бази даних, далі пароль і останній рядок назва самої бази даних.

Далі ми маємо підключитися до бази даних. Для цього використовується функція `mysqli_connect()` за допомогою заданих параметрів (ім'я сервера, користувача, пароль і ім'я бази даних) ми зможемо підключитися до бази даних. В самому кінці у нас відбувається перевірка підключення до бази даних завдяки умові:

```
– if (!$conn) {die("Помилка підключення до бази даних: " .  
mysqli_connect_error());}
```

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$databasename = "chat";

$conn = mysqli_connect($servername, $username, $password, $databasename);

if (!$conn) {
    die("Помилка підключення до бази даних: " . mysqli_connect_error());
}
?>
```

Рисунок 2.23 – код підключення до бази даних

Далі створемо файл `load_messages.php` (рис. 2.24), який відображає повідомлення у браузері. Він підключається до бази даних MySQL і завантажує всі повідомлення з таблиці `messages`. Він формує HTML-розмітку для кожного повідомлення, відображаючи ім'я користувача, текст повідомлення та час його

Автоматизована інформаційна система визначення імператива сталого розвитку відправлення, і виводить цю розмітку, щоб вона могла бути інтегрована у веб-сторінку чату.

Спочатку ми як завжди підключаємося до бази даних завдяки `mysqli_connect()`, адже без цього в нас нічого не буде працювати. В кожному файлі постійно треба прописувати підключення, щоб був доступ до бази даних. Далі в нас формується SQL-запит:

- `$sql = "SELECT username, message, timestamp FROM messages ORDER BY timestamp ASC"`: змінна `$sql` містить SQL-запит для отримання полів `username`, `message` та `timestamp` з таблиці `messages`;
- `$result = mysqli_query($conn, $sql)`: виконання SQL-запиту зберігається в змінній `$result`;
- `mysqli_num_rows($result) > 0`: перевірка чи є результати, якщо вони є то кожен рядок обробляється в циклі `while`, використовуючи функцію `mysqli_fetch_assoc()`, яка повертає асоціативний масив для кожного рядка;
- `htmlspecialchars()`: функція використовується для захисту від XSS-атак, перетворюючи спеціальні символи в HTML-ескейп-послідовності;
- `mysqli_close($conn)`: закриття підключення.

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$databasename = "chat";

$conn = mysqli_connect($servername, $username, $password, $databasename);

if (!$conn) {
    die("Помилка підключення до бази даних: " . mysqli_connect_error());
}

$sql = "SELECT username, message, timestamp FROM messages ORDER BY timestamp ASC";
$result = mysqli_query($conn, $sql);

if (mysqli_num_rows($result) > 0) {
    while($row = mysqli_fetch_assoc($result)) {
        echo '<div class="message">';
        echo '<strong>' . htmlspecialchars($row['username']) . '</strong> ';
        echo htmlspecialchars($row['message']) . ' <small>(' . $row['timestamp'] . ')</small>';
        echo '</div>';
    }
} else {
    echo "Немає повідомлень.";
}

mysqli_close($conn);
?>
```

Рисунок 2.24 – вміст файлу "load\_messages.php"

Після цього створемо "send\_message.php" (рис. 2.25), який приймає дані про користувача та його повідомлення через POST-запит з веб-форми. Він підключається до бази даних MySQL і зберігає нове повідомлення разом з ім'ям користувача та поточним часом у таблиці messages. Після успішного збереження повідомлення у базі даних, файл повертає підтвердження або повідомляє про помилку:

- `if ($_SERVER['REQUEST_METHOD'] == 'POST')` { : перевіряється, чи метод HTTP-запиту є POST. Це означає, що оброблятиметься форма, відправлена методом POST;

- `$username = $_POST['user']; $message = $_POST['message'];` отримання даних з форм надісланих методом POST, які зберігаються і змінних `$username` і `$message`;

- `$sql = "INSERT INTO messages (username, message, timestamp) VALUES (?, ?, NOW());"`; `$stmt = $conn->prepare($sql); $stmt->bind_param("ss", $username, $message);` підготовка та виконання SQL-запиту для вставки даних, спочатку створюється SQL-запит для вставки нових даних у таблицю, далі місця підставок (?) замінюються відповідними значеннями змінних `$username` і `$message` за допомогою методу `bind_param()`;

- `if ($stmt->execute()) { echo "Повідомлення успішно надіслано";} else { echo "Помилка: " . $stmt->error; }` : перевіряється, чи вдалося виконати запит. Якщо так, виводиться повідомлення "Повідомлення успішно надіслано". Якщо ні, виводиться повідомлення про помилку з описом помилки (`$stmt->error`).

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$databasename = "chat";

$conn = mysqli_connect($servername, $username, $password, $databasename);
if (!$conn) {
    die("Помилка підключення до бази даних: " . mysqli_connect_error());
}

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $username = $_POST['user'];
    $message = $_POST['message'];

    $sql = "INSERT INTO messages (username, message, timestamp) VALUES (?, ?, NOW())";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("ss", $username, $message);

    if ($stmt->execute()) {
        echo "Повідомлення успішно надіслано";
    } else {
        echo "Помилка: " . $stmt->error;
    }

    $stmt->close();
}
mysqli_close($conn);
?>
```

Рисунок 2.25 – вміст файлу " send\_message.php "

Далі створимо візуальну частину чату. У файлі HTML ми задаємо структуру інтерфейсу чату з полями для введення імені користувача та повідомлення, а також кнопкою відправки. Стилзація забезпечує відображення чату у вигляді окремого блоку зі скролінгом, а JavaScript-код відповідає за завантаження та відправку повідомлень, взаємодіючи з сервером через AJAX-запити (рис. 2.26).

```

<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
  $(document).ready(function() {
    // Load chat messages
    function loadMessages() {
      $.get('load_messages.php', function(data) {
        $('#chat-box').html(data);
      });
    }
    loadMessages();
    setInterval(loadMessages, 2000);

    // Send message
    $('#send-btn').click(function() {
      var user = $('#username').val();
      var message = $('#message-input').val();
      if(user && message) {
        $.post('send_message.php', { user: user, message: message }, function() {
          $('#message-input').val('');
          loadMessages();
        });
      }
    });
  });
</script>

<div>
<div id="chat-box">
</div>
<input type="text" id="username" placeholder="Enter your name" /><br/>
<input type="text" id="message-input" placeholder="Enter your message" />
<button id="send-btn">Send</button>
</div>

```

Рисунок 2.26 – Вміст файлу index.html

Після успішного підключення всіх необхідних файлів і налаштування усіх компонентів, ми отримали повністю функціональний чат, який може використовуватися для надсилання та отримання повідомлень у реальному часі. Візуальна складова чата зображена на рис. 2.27.

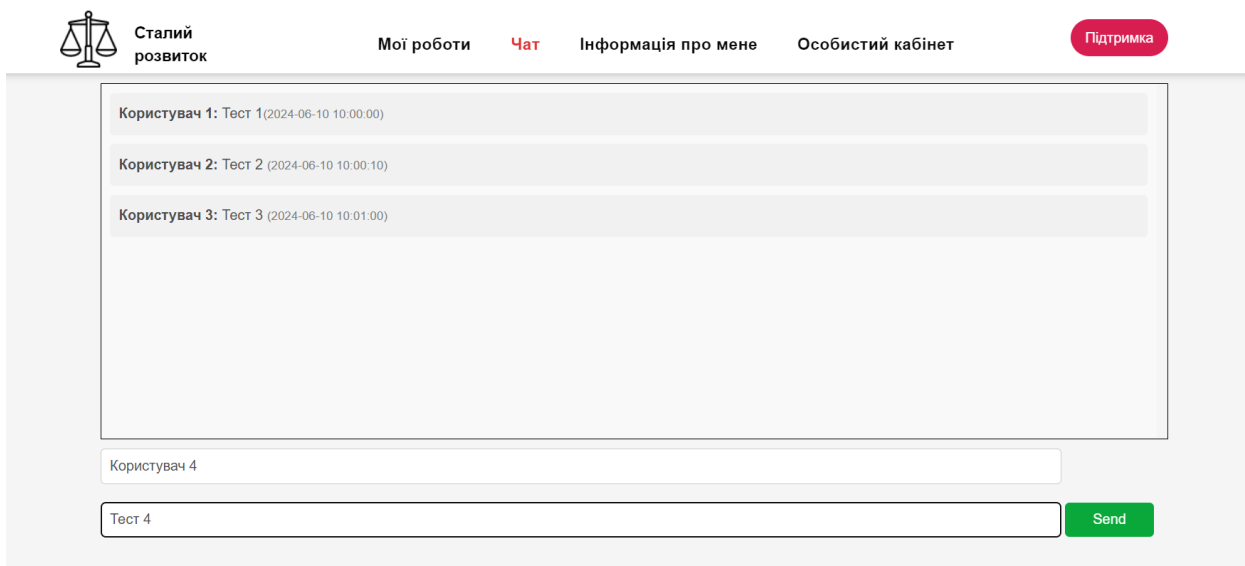
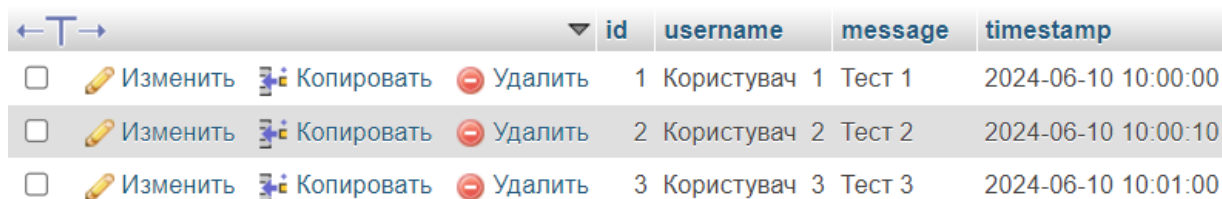


Рисунок 2.27 – Чат

Нижче наведена база даних (рис. 2.28), яка відображає інформацію про користувачів, їх повідомлення та дату відправки повідомлення.



	id	username	message	timestamp
<input type="checkbox"/> Изменить  Копировать  Удалить	1	Користувач 1	Тест 1	2024-06-10 10:00:00
<input type="checkbox"/> Изменить  Копировать  Удалить	2	Користувач 2	Тест 2	2024-06-10 10:00:10
<input type="checkbox"/> Изменить  Копировать  Удалить	3	Користувач 3	Тест 3	2024-06-10 10:01:00

Рисунок 2.28 – База даних в phpMyAdmin

Тепер перейдемо до можливості редагувати та доповнювати інформацію на сайті користувачами. Спочатку треба трішки змінити HTML код, адже зараз можливості редагувати інформації немає. Ми використаємо jQuery для додавання інтерактивності до веб-сторінки. За допомогою нього реалізуємо функції для відкриття та закриття форми редагування, а також обробка кліків по кнопці "Редагувати". Оновлений код виглядатиме ось так (рис. 2.29):

- `function openEditForm(titleId, contentId, listId) {`: функція відкриває форму редагування;
- `var titleText = document.getElementById(titleId).innerText;` `var contentTextLeft = document.getElementById(contentId).innerText;`: отримання тексту заголовка та основного контенту, який будемо редагувати;
- `var combinedContent = contentTextLeft + '\n\n---\n\n' + contentTextRight;`: об'єднання лівого контенту та списку у один текстовий блок з роздільником;
- `document.getElementById('content-id').value = contentId;`  
`document.getElementById('edit-title').value = titleText;`: встановлюємо значення форм редагування;
- `document.getElementById('edit-form').style.display = 'block';`: показ форми редагування;
- `function closeEditForm() {`: функція для закриття форми редагування;
- `document.getElementById('edit-content-form').onsubmit = function(event) {`: обробка події відправки форми редагування;



- `var titleText = document.getElementById('edit-title').value`: отримання заголовка з форми;
- `var contentText = document.getElementById('edit-textarea').value`: отримання контенту з форми;
- `var parts = contentText.split('\n\n---\n\n')`: розділення комбінованого контенту на частини;
- `var newLeftContent = parts[0].trim()`: отримання нових значень для лівого контенту та списку;
- `document.getElementById('eco-title').innerText = titleText`: оновлення значень в документі.

```

<script>
function openEditForm(titleId, contentId, listId) {
  var titleText = document.getElementById(titleId).innerText;
  var contentTextLeft = document.getElementById(contentId).innerText;
  var contentTextRight = Array.from(document.getElementById(listId).querySelectorAll('li')).map(li => li.textContent.trim()).join('');
  var combinedContent = contentTextLeft + '\n\n---\n\n' + contentTextRight;

  document.getElementById('content-id').value = contentId;
  document.getElementById('edit-title').value = titleText;
  document.getElementById('edit-textarea').value = combinedContent;
  document.getElementById('edit-form').style.display = 'block';
}

function closeEditForm() {
  document.getElementById('edit-form').style.display = 'none';
}

document.getElementById('edit-content-form').onsubmit = function(event) {
  event.preventDefault();

  var titleText = document.getElementById('edit-title').value;
  var contentText = document.getElementById('edit-textarea').value;
  var parts = contentText.split('\n\n---\n\n');

  if (parts.length === 2) {
    var newLeftContent = parts[0].trim();
    var newRightContent = parts[1].trim().split('\n').map(line => '<li>' + line + '</li>').join('');

    document.getElementById('eco-title').innerText = titleText;
    document.getElementById('eco-paragraph').innerText = newLeftContent;
    document.getElementById('eco-list').innerHTML = newRightContent;
  }

  closeEditForm();
};
</script>

```

Рисунок 2.29 – Оновлений вміст файлу index.html

Тепер треба створити базу даних і підключити її, щоб зберігати в ній контент (рис. 2.30).

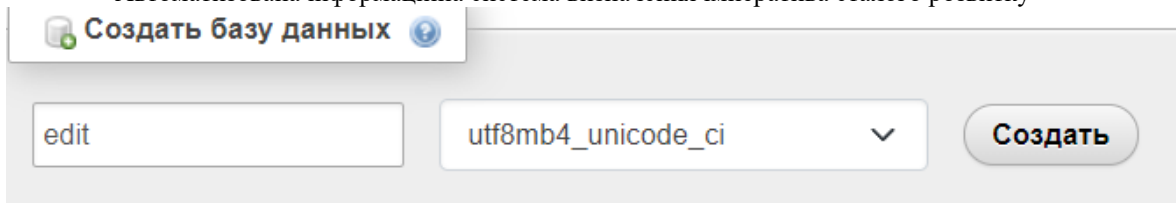


Рисунок 2.30 – Створення бази даних "edit"

За допомогою MySQL та phpMyAdmin створимо таблицю "content\_table" (рис. 2.31), вона нам потрібна для зберігання інформації яку користувач захоче внести на сайт.

```
CREATE TABLE content_table (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    title VARCHAR(255) NOT NULL,  
    content TEXT NOT NULL  
);|
```

Рисунок 2.31 – Створення таблиці "content\_table"

Тепер ми маємо знову підключитися до бази даних у PHP. Зробити обробку POST-запиту для збереження змін та закрити підключення. Для цього створимо PHP-скрипт (рис 2.32), який виконуватиме операції з редагуванням даних:

```

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    if (isset($_POST['content_id']) && isset($_POST['content_title']) && isset($_POST['content_text'])) {
        $contentId = $_POST['content_id'];
        $title = $_POST['content_title'];
        $content = $_POST['content_text'];
        $stmt = $conn->prepare("UPDATE content_table SET title=?, content=? WHERE id=?");
        $stmt->bind_param("ssi", $title, $content, $contentId);
        if ($stmt->execute()) {
            echo "Зміни успішно збережені.";
        } else {
            echo "Помилка під час збереження змін: " . $conn->error;
        }
        $stmt->close();
    } else {
        echo "Недостатньо даних для оновлення.";
    }
}

function getContentData($conn, $contentId) {
    $sql = "SELECT title, content FROM content_table WHERE id = ?";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("i", $contentId);
    $stmt->execute();
    $result = $stmt->get_result();
    $data = $result->fetch_assoc();
    $stmt->close();
    return $data;
}

```

Рисунок 2.32 – Вміст файлу "edit\_content.php"

Далі трішки редагуємо файл index.html, підключаємо до нього "edit\_content.php" і маємо ось такий результат (рис 2.33). Спробуємо щось змінити і подивимося чи можна редагувати дані на сайті (рис. 2.34).

# Редагувати





Рисунок 2.33 – Форма редагування та доповнення інформації

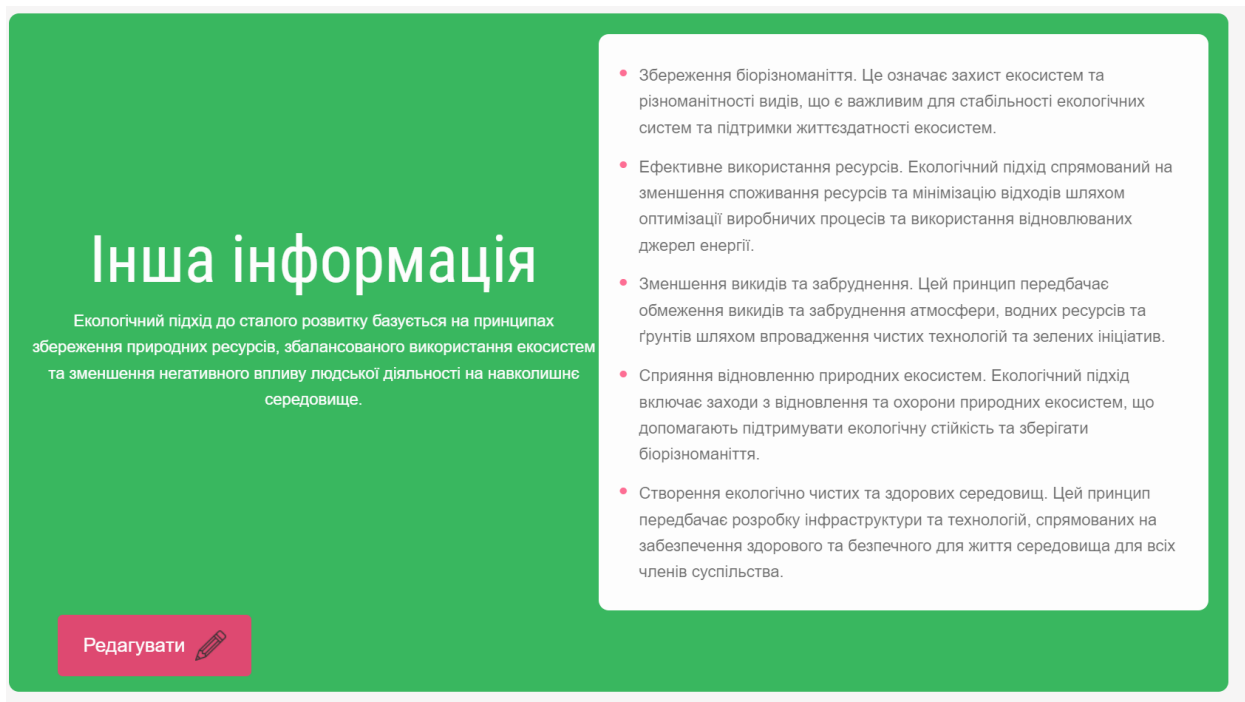


Рисунок 2.34 – результат редагування тексту через форму

## Висновки до розділу 2

У другому розділі кваліфікаційної роботи було здійснено проектування та реалізацію автоматизованої інформаційної системи, яка включає розробку веб-сайту та інтеграцію чату і можливість редагування інформації кореспондентами. Проведений аналіз дозволив обрати оптимальні методи та технології для реалізації автоматизованої інформаційної системи. Вибір технологій базувався на сучасних вимогах до функціональності, безпеки та продуктивності веб-додатків. Використання таких інструментів, як HTML, CSS, JavaScript, PHP та MySQL, забезпечило створення надійної та ефективної платформи. Розробка веб-сайту включала кілька етапів. Спочатку була створена шапка сайту, яка містить логотип, навігаційне меню та інші елементи інтерфейсу. Це дозволило створити зручний та інтуїтивно зрозумілий користувацький інтерфейс. Основна частина сайту була спроектована таким чином, щоб забезпечити зручність у використанні та легкість доступу до основної інформації. Структурований та зрозумілий контент забезпечує

користувачам легкий доступ до необхідних даних. Футер сайту включає контактну інформацію, посилання на соціальні мережі та інші важливі елементи. Це підвищує зручність користування сайтом та забезпечує додаткові можливості для взаємодії з користувачами. Інтеграція чату з можливістю редагування інформації стала ключовим елементом автоматизованої інформаційної системи. Загалом, проектування та реалізація автоматизованої інформаційної системи дозволили створити функціональний, зручний та безпечний веб-сайт. Виконані роботи забезпечили досягнення поставлених цілей та задач, а також створили надійну основу для подальшого розвитку та вдосконалення системи.

### 3 РЕКОМЕНДАЦІЇ ТА ПЕРСПЕКТИВИ ВИКОРИСТАННЯ АВТОМАТИЗОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ

#### 3.1 Оновлення та модернізація вебсайту

Після успішної реалізації автоматизованої інформаційної системи, а саме створення вебсайту, надзвичайно важливо забезпечити його регулярне оновлення та модернізацію. Оновлення вебсайту є критично важливим для підтримки актуальності інформації, підвищення рівня безпеки та покращення загального користувацького досвіду.

Регулярні оновлення контенту відіграють ключову роль у забезпеченні користувачів найсвіжішою інформацією. Це не лише підвищує рівень залученості користувачів, але й сприяє зростанню відвідуваності сайту. Постійне оновлення контенту повинно включати публікацію нових матеріалів, таких як блоги, статті, новини та інші ресурси, які можуть бути корисними для аудиторії. Крім того, слід приділяти особливу увагу підтримці різноманітних форматів контенту, таких як текстові матеріали, відеоролики, інфографіка та інтерактивні елементи, що допоможуть зробити інформацію більш доступною та цікавою для користувачів.

Одним з ключових аспектів створення контенту є розуміння того, як різні формати контенту можуть впливати на залучення та утримання уваги користувачів. Для цього доцільно порівняти ефективність текстових матеріалів та відеороликів за різними показниками.

Таблиця 3.1 – Порівняння ефективності тексту та відеоматеріалів

Показник	Текстові матеріали	Відеоматеріали
Залучення уваги	Середнє	Високе
Утримання уваги	Низьке	Високе
Середній час взаємодії	Короткий	Довгий
Рівень розуміння інформації	Високий	Дуже високий

## Закінчення таблиці 3.1

Пам'ятність	Середня	Висока
Залученість	Низька	Висока

З таблиці видно, що відеоролики значно перевершують текстові матеріали за багатьма ключовими показниками. Відеоролики мають високий рівень залучення та утримання уваги, що сприяє довшому часу взаємодії користувачів з контентом. Завдяки візуальним та аудіальним елементам відео, інформація передається більш ефективно та запам'ятовується краще, ніж текст. Це робить відеоролики дуже потужним інструментом для донесення важливих повідомлень та інформації.

Переваги відеоконтенту:

1) високе залучення. Відеоролики привертають увагу користувачів швидше, ніж текст, завдяки динамічному та візуально привабливому формату. Це дозволяє ефективніше донести інформацію до аудиторії та утримати її увагу на триваліший час;

2) легкість сприйняття. Відео поєднує в собі візуальні та аудіальні елементи, що спрощує процес сприйняття та розуміння інформації. Користувачі можуть швидше зрозуміти сутність повідомлення та запам'ятати його основні моменти;

3) збільшення взаємодії. Відеоконтент зазвичай отримує більше лайків, коментарів та поширень у соціальних мережах, ніж текстові матеріали. Це сприяє підвищенню залученості користувачів та розширенню аудиторії;

4) ефективність навчання. Відеоуроки та навчальні ролики дозволяють більш ефективно передавати знання та навички, оскільки користувачі можуть бачити процес у дії та слухати пояснення одночасно;

5) мобільна адаптивність. Відеоконтент легко адаптується до різних пристроїв, зокрема мобільних телефонів та планшетів, що робить його доступним для більшої кількості користувачів незалежно від місця та часу.

Бачачи наскільки відеоматеріали ефективніші за текстові, в майбутньому необхідно збільшити кількість відеоконтенту на вебсайті. Додавання більшої

Автоматизована інформаційна система визначення імператива сталого розвитку кількості відеороликів сприятиме кращому розумінню та усвідомленню інформації користувачами. Відео дозволять не лише передати складні ідеї та концепції у доступний спосіб, але й створити більш емоційно насичений та цікавий досвід для відвідувачів сайту.

Також модернізація вебсайту включає цілий ряд технічних оновлень, таких як оптимізація коду, покращення швидкості завантаження сторінок та забезпечення сумісності з новими версіями веббраузерів. Це допоможе підвищити продуктивність сайту та забезпечити його стабільну роботу на різних пристроях. Технічні оновлення повинні включати впровадження нових технологій та інструментів, таких як системи кешування, використання CDN (Content Delivery Network) для розповсюдження контенту та оптимізація зображень для зменшення часу завантаження сторінок. Для оптимізації коду можна використовувати такі технології, як мінімізація JavaScript та CSS, об'єднання файлів, а також видалення зайвих коментарів і пробілів. Ці заходи зменшать розмір файлів та знизять час завантаження сторінок. Використання асинхронного завантаження ресурсів дозволить уникнути блокування відображення сторінки під час завантаження великих файлів.

В майбутньому можна покращити швидкість завантаження сторінок. Це досягається за допомогою впровадження систем кешування, таких як Varnish Cache або Redis. Ці системи дозволяють зберігати копії часто запитуваних сторінок та даних, що зменшує навантаження на сервер і прискорює доставку контенту користувачам. Використання CDN (Content Delivery Network) дозволить розповсюджувати контент через мережу серверів, розташованих у різних частинах світу, що зменшить затримки та підвищить швидкість завантаження. Можна оптимізувати зображення, а саме зменшити їх розмір без втрати якості за допомогою таких інструментів, як TinyPNG або ImageOptim. Використання форматів зображень нового покоління, таких як WebP, дозволить зменшити обсяг даних, що передаються, та прискорити завантаження сторінок. Додатково, можна



впровадити методи *lazy loading*, що дозволяють завантажувати зображення лише тоді, коли вони стають видимими на екрані користувача.

Одним з ключових аспектів модернізації є впровадження нових функціональних можливостей, таких як інтеграція з соціальними мережами, додавання нових форм зворотного зв'язку та розширення функціональності чату. Інтеграція з соціальними мережами дозволяє користувачам легко ділитися контентом та взаємодіяти з організацією через свої улюблені платформи. Це може бути реалізовано за допомогою використання API соціальних мереж, таких як Facebook Graph API або Twitter API, що дозволяють автоматично публікувати новини та оновлення на сторінках соціальних мереж.

Додавання нових форм зворотного зв'язку, таких як опитування, анкети та коментарі, сприяє залученню користувачів та покращенню комунікації з ними. Це може бути реалізовано за допомогою таких інструментів, як Google Forms або SurveyMonkey, що дозволяють створювати інтерактивні опитування та анкети. Розширення функціональності чату, включаючи впровадження чат-ботів та системи онлайн-підтримки, дозволяє забезпечити оперативну допомогу користувачам у разі виникнення проблем. Чат-боти можуть бути створені за допомогою таких платформ, як Dialogflow або Microsoft Bot Framework, що дозволяють автоматизувати відповіді на поширені запитання та забезпечити цілодобову підтримку користувачів.

Забезпечення мобільної адаптивності сайту є особливо важливим, оскільки все більше користувачів використовують мобільні пристрої для доступу до Інтернету. Адаптивний дизайн дозволяє сайту коректно відображатися на різних пристроях, забезпечуючи зручний доступ до інформації незалежно від розміру екрану. Це включає використання медіа-запитів CSS для адаптації макету, оптимізацію зображень для мобільних пристроїв та забезпечення швидкого завантаження сторінок на повільних мережах. Впровадження технологій AMP (Accelerated Mobile Pages) допоможе створити швидкі та легкі мобільні версії сторінок, що значно підвищить продуктивність та зручність використання сайту на

Автоматизована інформаційна система визначення імператива сталого розвитку мобільних пристроях. Використання фреймворків для адаптивного дизайну, таких як Bootstrap або Foundation, дозволить швидко та ефективно створювати адаптивні макети, які будуть коректно відображатися на будь-яких пристроях.

Забезпечення безпеки вебсайту є ще одним важливим аспектом оновлення та модернізації. Регулярне оновлення систем безпеки, впровадження нових методів захисту від атак та забезпечення захисту даних користувачів допоможуть запобігти потенційним загрозам та забезпечити надійність інформаційної системи. Це може включати встановлення сертифікатів SSL для шифрування даних, використання сучасних методів аутентифікації та авторизації, регулярні перевірки на вразливості та впровадження механізмів резервного копіювання і відновлення даних. Використання сучасних засобів захисту, таких як WAF (Web Application Firewall), допоможе захистити вебсайт від поширених атак, таких як SQL-ін'єкції та XSS-атаки. Регулярні перевірки на вразливості за допомогою таких інструментів, як OWASP ZAP або Nessus, дозволять виявити та виправити потенційні проблеми безпеки до того, як вони будуть використані зловмисниками.

Не менш важливим є забезпечення підтримки користувачів. Це може включати створення розділу з часто задаваними питаннями (FAQ), інтеграцію системи онлайн-підтримки та забезпечення можливості швидкого зв'язку з технічною підтримкою у разі виникнення проблем. Регулярне збирання зворотного зв'язку від користувачів допоможе виявити проблемні місця та покращити функціональність і зручність вебсайту. Використання систем онлайн-підтримки, таких як Zendesk або Freshdesk, дозволить забезпечити оперативну та ефективну підтримку користувачів, швидко реагуючи на їхні запитання та проблеми. Інтеграція системи зворотного зв'язку на основі опитувань та аналізу даних допоможе краще розуміти потреби користувачів та вчасно вносити необхідні покращення.

Отже, регулярне оновлення та модернізація вебсайту є ключовими аспектами забезпечення його ефективного функціонування. Вони включають оновлення контенту, впровадження нових технологій та функціональних можливостей,

Автоматизована інформаційна система визначення імператива сталого розвитку забезпечення мобільної адаптивності та безпеки, а також підтримку користувачів. Завдяки цим заходам можна забезпечити стабільну роботу вебсайту, підвищити його привабливість для користувачів та забезпечити ефективну взаємодію з аудиторією і збільшити рівень її освіченості.

### **3.2 Використання нових технологій**

Для забезпечення конкурентоспроможності та підвищення ефективності роботи автоматизованої інформаційної системи важливо активно впроваджувати нові технології. Використання сучасних технологій дозволить покращити функціональність вебсайту, забезпечити його інтерактивність та підвищити зручність користування.

Однією з перспективних технологій є використання штучного інтелекту та машинного навчання для аналізу даних та автоматизації процесів. Наприклад, можна впровадити систему рекомендацій, яка буде надавати користувачам персоналізовані пропозиції на основі їхніх інтересів та поведінки на сайті. Також можна використовувати чат-ботів для автоматизації відповіді на запити користувачів та надання швидкої допомоги у режимі реального часу. Штучний інтелект може також бути використаний для автоматичного аналізу та класифікації великих обсягів даних, що допоможе у прийнятті більш обґрунтованих рішень. Наприклад, на сайті можна впровадити розумного помічника, який буде надавати персоналізовану допомогу користувачам, відповідаючи на їхні запитання та надаючи рекомендації на основі аналізу їхньої поведінки. Такий помічник може використовувати технології обробки природної мови (NLP) для розуміння запитів користувачів та надання відповідей у режимі реального часу. Це значно покращить користувацький досвід та підвищить задоволеність відвідувачів сайту.

Штучний інтелект також може використовуватися для аналізу великих обсягів даних, що генеруються користувачами, та надання управлінських інсайтів. Наприклад, аналізуючи поведінку користувачів на сайті, можна виявити найпопулярніші сторінки та контент, визначити слабкі місця у навігації або

Автоматизована інформаційна система визначення імператива сталого розвитку виявити потенційні проблеми з користувацьким інтерфейсом. Це дозволить вчасно реагувати на проблеми та постійно вдосконалювати сайт.

Інтеграція технологій великих даних (Big Data) дозволить здійснювати аналіз великого обсягу інформації та отримувати цінні інсайти для прийняття управлінських рішень. Це може включати аналіз поведінки користувачів, моніторинг ефективності рекламних кампаній та оптимізацію контенту сайту на основі отриманих даних. Використання Big Data також дозволить виявляти тренди та прогнозувати зміни на ринку, що допоможе організації залишатися конкурентоспроможною.

Використання хмарних технологій (Cloud Computing) забезпечить масштабованість та гнучкість інформаційної системи. Хмарні рішення дозволяють зберігати великі обсяги даних, забезпечувати доступ до них з будь-якого пристрою та здійснювати автоматичні резервні копії для запобігання втрати інформації. Крім того, хмарні технології дозволяють знизити витрати на інфраструктуру та забезпечити високу доступність і надійність сервісів.

Інтеграція технологій доповненої та віртуальної реальності (AR/VR) може надати користувачам нові можливості для взаємодії з контентом сайту. Це може бути корисно для демонстрації продуктів, проведення віртуальних турів та організації онлайн-заходів. Використання AR/VR технологій дозволить створити більш захоплюючий та інтерактивний досвід для користувачів, що сприятиме підвищенню їх залученості та задоволеності.

Одним з важливих напрямків є впровадження VR-технологій на вебсайті. Використання окулярів віртуальної реальності дозволяє створити новий рівень взаємодії з користувачами, надаючи їм можливість зануритися у віртуальне середовище та взаємодіяти з контентом у тривимірному просторі. Це може бути особливо корисно для демонстрації, які важко повністю оцінити через традиційні 2D-зображення. Наприклад ми можемо візуалізувати наслідки небалого ставлення до нашої планети через сотні років, застосовуючи технології віртуальної реальності (VR), що допоможе краще зрозуміти наслідки несталого розвитку.

Використовуючи VR, ми можемо створити реалістичні віртуальні середовища, де користувачі зможуть інтерактивно взаємодіяти з моделями змінених екосистем та об'єктів, щоб посилити їхнє розуміння і співчуття до потенційної катастрофи, яка може виникнути внаслідок недбалого ставлення до природних ресурсів та навколишнього середовища.

Для налаштування сайту під VR можна використовувати технології, такі як WebVR і WebXR, які дозволяють розробляти віртуальні та доповнені реальності прямо у веббраузері. Це дозволить користувачам відчувати повноцінний VR-досвід без необхідності завантаження додаткового програмного забезпечення. Додатково, інтеграція з платформами, такими як Google Cardboard або Oculus Rift, може забезпечити сумісність з різними VR-пристроями.

Загалом, використання нових технологій дозволить покращити функціональність та ефективність автоматизованої інформаційної системи, забезпечити її актуальність та конкурентоспроможність у сучасному динамічному середовищі. Це вимагатиме постійного моніторингу новітніх розробок, навчання персоналу та готовності до змін, але в підсумку сприятиме створенню більш надійної, гнучкої та ефективної інформаційної системи.

### **3.3 Впровадження соціальних логінів**

Впровадження соціальних логінів на вебсайті є важливим кроком для підвищення зручності користування та покращення загальної безпеки системи. Соціальні логіни дозволяють користувачам реєструватися та входити на вебсайт, використовуючи свої облікові записи в популярних соціальних мережах, таких як Facebook, Google та Twitter. Це значно спрощує процес реєстрації та знижує бар'єри для нових користувачів, одночасно знижуючи ризик анонімності та активності спам-ботів. Інтеграція соціальних логінів забезпечує кілька ключових переваг:

1) зручність для користувачів. Користувачі можуть швидко та легко реєструватися і входити на сайт без необхідності заповнювати довгі форми реєстрації або запам'ятовувати нові паролі;

2) зниження бар'єрів для нових користувачів. Спрощений процес реєстрації знижує психологічні бар'єри для нових користувачів, що може сприяти збільшенню кількості зареєстрованих користувачів та активних відвідувачів сайту;

3) боротьба зі спам-ботами та анонімністю. Використання облікових записів із соціальних мереж знижує ризик анонімної активності та активності спам-ботів, оскільки соціальні мережі зазвичай вимагають підтвердження особистих даних під час створення облікових записів.

Інтеграція соціальних логінів також дозволяє автоматично заповнювати профілі користувачів даними з їхніх облікових записів у соціальних мережах. Інформація, така як ім'я, прізвище, електронна адреса, аватар, та інші деталі, автоматично переноситься з профілю соціальної мережі користувача до його профілю на вебсайті. Це значно зменшує кількість кроків, необхідних для заповнення профілю, і робить процес більш зручним для користувачів. Бонусом є зменшення ризику помилок, які можуть виникнути під час ручного введення даних, це забезпечує більш точну і повну інформацію про користувачів.

Першим кроком у впровадженні соціальних логінів є вибір відповідних провайдерів аутентифікації. Найпопулярнішими варіантами є Facebook Login, Google Sign-In та Twitter Login. Кожен з цих провайдерів має свої переваги та особливості, тому доцільно інтегрувати декілька варіантів, щоб надати користувачам вибір.

Під час інтеграції соціальних логінів важливо забезпечити захист персональних даних користувачів. Це включає:

1) шифрування даних. Використання протоколів HTTPS для забезпечення безпечного передавання даних між вебсайтом і серверами соціальних мереж;

2) обмеження доступу до даних. Збирання лише необхідної інформації для реєстрації та аутентифікації, дотримуючись принципу мінімізації даних;

3) згода користувачів. Отримання чіткої та явної згоди користувачів на доступ до їх даних у соціальних мережах.

Переваги соціальних логінів для вебсайту:

1) збільшення кількості користувачів. Спрощений процес реєстрації сприяє залученню нових користувачів та збільшенню активності на сайті;

2) покращення якості даних. Автоматичне заповнення профілів забезпечує точніші та повніші дані про користувачів;

3) зменшення спаму та анонімної активності. Використання соціальних логінів знижує ризик активності спам-ботів та анонімних користувачів, оскільки соціальні мережі вимагають підтвердження особистих даних.

Впровадження соціальних логінів відкриває нові можливості для покращення функціональності вебсайту:

1) персоналізація контенту. Використання даних з соціальних мереж для надання персоналізованих рекомендацій та контенту, що підвищує залученість користувачів;

2) аналітика та моніторинг. Аналіз поведінки користувачів, які увійшли через соціальні логіни, дозволяє отримувати цінні інсайти для покращення функціональності та контенту сайту;

3) інтеграція з іншими сервісами. Використання соціальних логінів для інтеграції з іншими сервісами та додатками, що дозволяє розширити функціональність вебсайту та забезпечити користувачам більш зручний та цілісний досвід.

Отже, впровадження соціальних логінів є важливим кроком для забезпечення зручності та безпеки користувачів вебсайту. Це дозволяє спростити процес реєстрації, знизити бар'єри для нових користувачів та забезпечити високий рівень безпеки даних. Завдяки цьому вебсайт зможе залучити більше користувачів, покращити якість даних та забезпечити ефективну боротьбу зі спам-ботами та анонімною активністю.

### Висновки до розділу 3

У третьому розділі були розглянуті рекомендації та перспективи використання автоматизованої інформаційної системи. Одним з ключових аспектів є регулярне оновлення та модернізація вебсайту, що включає оновлення контенту, технічні покращення, забезпечення безпеки та підтримку користувачів. Ці заходи спрямовані на підтримку актуальності інформації, підвищення безпеки та покращення користувацького досвіду. Регулярні оновлення контенту, оптимізація коду, впровадження нових функціональних можливостей та забезпечення мобільної адаптивності допоможуть зберегти вебсайт конкурентоспроможним та привабливим для користувачів. Також були розглянуті можливості використання нових технологій, таких як штучний інтелект, машинне навчання, великі дані, хмарні технології та доповнена і віртуальна реальність. Впровадження цих технологій дозволить покращити функціональність вебсайту, забезпечити його інтерактивність, підвищити зручність користування та ефективність роботи. Використання новітніх технологій сприятиме створенню більш надійної, гнучкої та ефективної інформаційної системи, яка зможе швидко адаптуватися до змінних умов та забезпечити високий рівень конкурентоспроможності. Також впровадження соціальних логінів є важливим кроком для підвищення зручності користування та покращення загальної безпеки системи. Соціальні логіни дозволяють користувачам реєструватися та входити на вебсайт, використовуючи свої облікові записи в популярних соціальних мережах, таких як Facebook, Google та Twitter. Це значно спрощує процес реєстрації та знижує бар'єри для нових користувачів, одночасно знижуючи ризик анонімності та активності спам-ботів. Автоматичне заповнення профілів користувачів даними з їхніх облікових записів у соціальних мережах зменшує кількість кроків, необхідних для заповнення профілю, і робить процес більш зручним для користувачів. Загалом, забезпечення регулярного оновлення та впровадження нових технологій є критично важливими для підтримки ефективності та конкурентоспроможності автоматизованої інформаційної системи. Інтеграція соціальних логінів сприятиме залученню



Автоматизована інформаційна система визначення імператива сталого розвитку більшої кількості користувачів, покращенню якості даних та забезпеченню ефективної боротьби зі спам-ботами та анонімною активністю. Це вимагатиме постійного моніторингу новітніх розробок, навчання персоналу та готовності до змін, що в кінцевому підсумку сприятиме створенню безпечного, зручного та ефективного інформаційного середовища для користувачів.

## **ВИСНОВКИ**

На основі розглянутих аспектів екологічного, економічного та соціального підходів до сталого розвитку, можемо зробити висновок, що лише взаємодія цих трьох сфер може забезпечити сталість та гармонію в розвитку суспільства. Екологічний підхід допомагає зберегти природні ресурси та зменшити негативний вплив на довкілля, що є основою для економічного процвітання та соціального добробуту. Економічна стійкість не може бути досягнута без урахування екологічних та соціальних аспектів, а соціальна справедливість і взаєморозуміння в суспільстві є запорукою його стабільності та розвитку. Кожен з принципів екологічного, економічного та соціального підходів важливий для сталого розвитку, оскільки вони взаємопов'язані та доповнюють один одного. Збереження біорізноманіття та природних ресурсів, ефективне використання ресурсів та стимулювання інновацій у економіці, а також забезпечення соціальної справедливості та розвитку людського потенціалу - усе це разом формує основу для сталого та гармонійного розвитку суспільства. Таким чином, розвиток, який ґрунтується на принципах сталого розвитку, є ключем до стабільності та процвітання сучасного світу. Шлях до досягнення цієї мети лежить у поєднанні екологічних, економічних та соціальних зусиль, спрямованих на збереження природи, підтримку економічного розвитку та забезпечення благополуччя всіх членів суспільства. Розробка та впровадження вебсайту, яка є важливою частиною цієї роботи, забезпечує ефективну комунікацію та доступ до інформації для користувачів. Завдяки сучасному дизайну, інтерактивним елементам та можливостям для зворотного зв'язку, вебсайт сприятиме підвищенню рівня залученості користувачів та забезпеченню їхнього задоволення від користування.

**СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. "Plan B: Mobilizing to Save Civilization" by Lester R. Brown.
2. "Limits to Growth" by Donella H. Meadows, Dennis L. Meadows, Jørgen Randers, and William W. Behrens III.
3. "Development as Freedom" by Amartya Sen.
4. "The Limits to Growth: A Report for the Club of Rome's Project on the Predicament of Mankind" by Donella H. Meadows, Dennis L. Meadows, Jørgen Randers, and William W. Behrens III.
5. "The Age of Sustainable Development" by Jeffrey D. Sachs.
6. "Blueprint for Survival" by Hermann Daly and E. F. Schumacher.
7. "Sustainable Development: Exploring the Contradictions" by Frances Fahy.
8. "Economics of Sustainable Development" by James E. Mitchell.
9. "Beyond Economic Growth: An Introduction to Sustainable Development" by Nel W. J. Nelson.
10. "State of the World" by Worldwatch Institute.
11. "The Economics of Ecosystems and Biodiversity: Ecological and Economic Foundations" edited by Pushpam Kumar.
12. "Ecological Economics: Principles and Applications" by Herman E. Daly and Joshua Farley.
13. "Sustainable Development Handbook" edited by Stephen A. Roosa and Stuart S. Nagel.
14. "Global Environmental Governance: Foundations of Contemporary Environmental Studies" by Jean-Frédéric Morin and Amandine Orsini.
15. "The Green Economy: Environment, Sustainable Development and the Politics of the Future" by Adrian Parr.
16. "Sustainable Development Goals: Their Impacts on Forests and People" by John A. Parrotta and Stephanie Mansourian.

17. "HTML and CSS: Design and Build Websites" by Jon Duckett - A visually engaging guide to HTML and CSS. <https://www.amazon.com/HTML-CSS-Design-Build-Websites/dp/1118008189>
18. "JavaScript and JQuery: Interactive Front-End Web Development" by Jon Duckett - This book complements the HTML and CSS book by diving into JavaScript and jQuery. <https://www.amazon.com/JavaScript-JQuery-Interactive-Front-End-Development/dp/1118531647>
19. Mozilla Developer Network (MDN) Web Docs - Comprehensive resource for HTML, CSS, JavaScript, and more. <https://developer.mozilla.org/>
20. "Eloquent JavaScript" by Marijn Haverbeke - A thorough introduction to JavaScript. <https://eloquentjavascript.net/>
21. "You Don't Know JS" by Kyle Simpson - A deep dive into JavaScript. <https://github.com/getify/You-Dont-Know-JS>
22. W3Schools - Tutorials and references for web development languages. <https://www.w3schools.com/>
23. "Learning PHP, MySQL & JavaScript" by Robin Nixon - Covers PHP, MySQL, and JavaScript. <https://www.amazon.com/Learning-PHP-MySQL-JavaScript-JavaScript/dp/1491978910>
24. Codecademy - Interactive lessons for web development. <https://www.codecademy.com/>
25. freeCodeCamp - Free courses and community for web development. <https://www.freecodecamp.org/>
26. "PHP & MySQL: Novice to Ninja" by Kevin Yank - A beginner-friendly guide to PHP and MySQL. <https://www.amazon.com/PHP-MySQL-Novice-Kevin-Yank/dp/0987153087>
27. Stack Overflow - A forum for developers to ask questions and share knowledge. <https://stackoverflow.com/>

28. "CSS: The Definitive Guide" by Eric A. Meyer and Estelle Weyl - An in-depth reference on CSS. <https://www.amazon.com/CSS-Definitive-Guide-Visual-Presentation/dp/1449393195>
29. "Pro Git" by Scott Chacon and Ben Straub - A comprehensive guide to Git, available for free online. <https://git-scm.com/book/en/v2>
30. The Odin Project - A project-based curriculum for learning web development. <https://www.theodinproject.com/>