

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавр

(ступінь вищої освіти)

на тему

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

ВЕБСАЙТУ ТУРИСТИЧНИХ МІСЦЬ НАСЕЛЕНОГО ПУНКТУ

SOFTWARE FOR SETTLEMENT'S

TOURIST PLACES WEBSITE

Виконав(ла): студент(ка) 4 курсу, групи КНТ-112сп

Спеціальності 121 Інженерія програмного

(код і найменування спеціальності)

забезпечення

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

ДОМАШЕНКО Б.Р.

(ПРИЗВИЩЕ та ініціали)

Керівник ІЛЬЯШЕНКО М.Б.

(ПРИЗВИЩЕ та ініціали)

Рецензент ШИТІКОВА О.В.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет КНТ
Кафедра програмних засобів
Ступінь вищої освіти бакалавр
Спеціальність 121 Інженерія програмного забезпечення
(код і найменування)

Освітня програма (спеціалізація) Інженерія програмного забезпечення
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
Сергій СУББОТІН
“ ” 2025 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

ДОМАШЕНКА Богдана Романовича

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Програмне забезпечення вебсайту туристичних місць населеного пункту. Software for Settlement's Tourist Places Website

керівник проєкту (роботи) к.т.н., доцент, ІЛЬЯШЕНКО Матвій Борисович,
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від “ 28 ” квітня 2025 року № 209

2. Строк подання студентом проєкту (роботи) 03 червня 2025 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Матеріали і методи. 3. Опис програми. 4. Експлуатація, тестування та експериментальне дослідження програми.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів) Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	ІЛІЯШЕНКО М.Б., доцент		
Нормоконтроль	КАМІНСЬКА Ж.К., асистент		

7. Дата видачі завдання “28” квітня 2025 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	1 тиждень	Розділ 1
3	Вибір мови програмування та інших технологій розробки.	2 тиждень	Розділ 2
4	Розробка структури програми.	2 тиждень	Розділ 3
5	Розробка програми.	3-4 тижні	Розділи 3, 4
6	Тестування та експериментальне дослідження програмного забезпечення.	5 тиждень	Розділ 4
7	Оформлення пояснювальної записки та документів до неї.	6 тиждень	Додатки
8	Нормоконтроль та рецензування.	7 тиждень	
9	Захист роботи.	8 тиждень	

Студент(ка)

(підпис) Богдан ДОМАШЕНКО
(Імя ПРИЗВИЩЕ)

Керівник проєкту (роботи)

(підпис) Матвій ІЛІЯШЕНКО
(Імя ПРИЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра:
83 с., 4 табл., 30 рис., 3 дод., 15 джерел.

JAVASCRIPT, REACT, НАСЕЛЕНИЙ ПУНКТ, МОВА ПРОГРАМУВАННЯ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ТУРИСТИЧНІ МІСЦЯ.

Об'єкт дослідження – процес розробки програмного забезпечення для підтримки муніципальних вебсайтів.

Предмет дослідження – програмні засоби для підтримки вебсайтів населених пунктів.

Мета роботи – розробка програмного забезпечення вебсайту туристичних місць населеного пункту.

Матеріали, методи та технічні засоби: мова програмування JavaScript, фреймворки React та Redux.

Результати. Розроблено проєктні рішення для створення програмного забезпечення вебсайту туристичних місць населеного пункту. Створено програмне забезпечення вебсайту туристичних місць населеного пункту. Проведено тестування розробленої програмної системи для підтримки вебсайту туристичних місць населеного пункту.

Висновки. Мету роботи досягнуто. Розроблено програмне забезпечення для підтримки вебсайту туристичних місць населеного пункту за допомогою мови програмування JavaScript та середовища розробки фреймворки React та Redux.

Галузь використання – туризм.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 83 pages, 4 tables, 30 figures, 3 appendixes, 15 sources.

JAVASCRIPT, REACT, SETTLEMENT, PROGRAMMING LANGUAGE, SOFTWARE, TOURIST PLACES.

The object of research is the process of developing software to support municipal websites.

The subject of the research is software to support settlement websites.

The purpose of this work is to develop the software for settlement's tourist places website.

Materials, methods and technical tools: JavaScript programming language, React and Redux frameworks.

Results. Project solutions for the creation of software for settlement's tourist places website has been designed. The software for settlement's tourist places website has been developed. The developed software system for settlement's tourist places website has been tested.

Conclusions. The goal of the work has been achieved. The software tools for settlement's tourist places website using the JavaScript programming language and React and Redux frameworks have been developed.

Scope of use – tourism.

ЗМІСТ

	С.
Перелік скорочень та умовних позначок	8
Вступ	9
1 Аналіз предметної області	11
1.1 Програмне забезпечення для підтримки вебсайтів населених пунктів ..	11
1.2 Аналіз програмного забезпечення для підтримки вебсайтів населених пунктів	23
1.3 Висновки за розділом 1	34
2 Матеріали і методи	36
2.1 Вибір мови програмування	36
2.2 Вибір середовища розробки для створення програмного забезпечення вебсайту туристичних місць населеного пункту	38
2.3 Висновки за розділом 2	40
3 Опис програми	41
3.1 Структура програмного забезпечення вебсайту туристичних місць населеного пункту	41
3.2 Функціонування програмного забезпечення вебсайту туристичних місць населеного пункту	43
3.3 Розробка бази даних програмного забезпечення вебсайту туристичних місць населеного пункту	46
3.4 Проєктування інтерфейсу програмного забезпечення вебсайту туристичних місць населеного пункту	47
3.5 Висновки за розділом 3	50
4 Експлуатація, тестування та експериментальне дослідження програми	52
4.1 Призначення й умови застосування програми	52
4.2 Характеристики вебсайту туристичних місць населеного пункту	53
4.3 Інструкція по експлуатації програми	54
4.3.1 Звернення до програми	54
4.3.2 Вхідні й вихідні дані	54

4.3.3 Повідомлення.....	54
4.4 Виконання програмного забезпечення вебсайту туристичних місць населеного пункту.....	55
4.5 Тестування програмного забезпечення вебсайту туристичних місць населеного пункту.....	58
4.6 Висновки за розділом 4.....	58
Висновки.....	59
Перелік джерел посилань.....	62
Додаток А Технічне завдання.....	64
Додаток Б Фрагмент тексту програми	68
Додаток В Слайди презентації	76

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

API – Application Programming Interface;

JSX – JavaScript Syntax eXtension;

MVC – Model-View-Controller;

БД – база даних;

НП – населений пункт;

ПЗ – програмне забезпечення.

ВСТУП

Актуальність програмного забезпечення вебсайту туристичних місць населеного пункту зумовлена сучасними тенденціями розвитку цифрових технологій і зростаючою роллю онлайн ресурсів у повсякденному житті. Такий вебсайт стає важливим інструментом популяризації місцевого туризму, дозволяючи привернути увагу як внутрішніх, так і іноземних туристів. У світі, де більшість подорожей починається з пошуку інформації в Інтернеті, наявність якісного вебресурсу є ключем до ефективного залучення нових відвідувачів [1], [2].

Для туристів вебсайт забезпечує доступ до актуальної інформації про визначні місця, культурні події, заклади харчування та проживання. Інтерактивні можливості, такі як інтегровані карти чи онлайн бронювання, роблять планування подорожі зручним і швидким. Водночас вебсайт сприяє підтримці місцевого бізнесу, адже надає платформу для просування готелів, ресторанів та інших послуг [3], [4].

З точки зору місцевих органів влади чи власників ресурсу, програмне забезпечення дозволяє ефективно управляти туристичними потоками. Інтеграція аналітичних інструментів дає можливість оцінювати популярність окремих місць чи заходів, що допомагає приймати стратегічні рішення щодо розвитку інфраструктури та туристичних продуктів [5], [6].

Таким чином, вебсайт туристичних місць є важливим інструментом для розвитку туризму, стимулювання економічної активності та підвищення якості обслуговування відвідувачів. У сучасних умовах його створення є не лише актуальним, але й необхідним для забезпечення конкурентоспроможності регіону на туристичному ринку [6], [7].

Проте деякі програмні засоби та вебсайти туристичних місць населених пунктів характеризуються недостатньо широким функціоналом. Також поширеною проблемою є відсутність актуальної інформації або складність її оновлення через застарілий або незручний інтерфейс, внаслідок чого туристи

можуть отримувати застарілі дані про події, об'єкти чи послуги. Відсутність адаптації вебсайтів туристичних місць населених пунктів до мобільних пристроїв унеможлиблює комфортне використання ресурсу з телефонів чи планшетів, що є важливим у сучасному світі, де мобільний Інтернет займає значну частку обсягу передачі даних. Ще одним недоліком може стати відсутність багатомовної підтримки, що обмежує доступ до сайту для іноземних туристів. Крім того, ускладнений інтерфейс або незрозуміла навігація знижують задоволеність користувачів і роблять пошук потрібної інформації тривалим і виснажливим [1]-[7].

Тому актуальною є розробка програмного забезпечення вебсайту туристичних місць населеного пункту. У дипломній кваліфікаційній роботі бакалавра розв'язується актуальне завдання розробки програмного забезпечення вебсайту туристичних місць населеного пункту.

Для досягнення поставленої мети у кваліфікаційній роботі бакалавра необхідно розв'язати такі задачі:

- виконати аналіз предметної області та програмних засобів для підтримки вебсайтів населених пунктів;
- здійснити проєктування програмного забезпечення вебсайту туристичних місць населеного пункту;
- створити програмне забезпечення вебсайту туристичних місць населеного пункту;
- виконати тестування розробленого програмного забезпечення вебсайту туристичних місць населеного пункту.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Програмне забезпечення для підтримки вебсайтів населених пунктів

Програмне забезпечення для вебсайтів населених пунктів створюється для полегшення організації муніципальних процесів і вирішення адміністративних завдань. Воно спрямоване на автоматизацію таких дій, як управління дозволами, ліцензування, дотримання норм і забезпечення різноманітних послуг для громадян [1].

Ці рішення розробляються для підвищення ефективності роботи місцевих адміністрацій, зниження витрат і покращення управлінських процесів. Завдяки модульній структурі програмне забезпечення легко адаптується до потреб різних громад і органів самоврядування, дозволяючи налаштовувати функціонал залежно від завдань кожного відділу [1].

Цінова політика таких програмних рішень залежить від кількості модулів, які обираються для використання, а також від масштабів впровадження. Програмне забезпечення для підтримки діяльності органів влади населених пунктів може бути інтегроване як для окремого відділу, так і для всієї структури муніципалітету [1], [2].

Користувачам програмного забезпечення для вебсайтів населених пунктів (НП) надається доступ до готових шаблонів і багатофункціональних інструментів, таких як інтерактивні мапи та багатокористувацькі облікові записи. Це забезпечує зручність у роботі та ефективне виконання завдань, пов'язаних з управлінням населеним пунктом [1], [2].

Програмне забезпечення для вебсайтів населених пунктів забезпечує оптимізацію роботи місцевих органів влади, дозволяючи підвищити їх ефективність та покращити якість наданих послуг. Завдяки впровадженню таких цифрових рішень, муніципальні структури можуть більш раціонально використовувати ресурси та скорочувати витрати часу на виконання повторюваних завдань [1], [2].

Технології автоматизації допомагають зменшити залежність від ручної праці, дозволяючи працівникам зосередитися на більш важливих задачах. Оптимізовані процеси покращують доступність послуг для мешканців НП, забезпечуючи швидке та зручне їх отримання [1], [2].

Додатковим аспектом використання програмного забезпечення для підтримки вебсайтів населених пунктів є модернізація фінансових операцій, яка знижує ймовірність помилок і спрощує процес розрахунків. Це сприяє не тільки зручності для мешканців, але й ефективнішому управлінню бюджетними коштами [1], [2].

Хмарні технології, інтегровані в такі програмні рішення, дозволяють зберігати та обробляти дані з будь-якого місця. Завдяки цьому співробітники адміністрації населених пунктів можуть виконувати свої обов'язки дистанційно, а мешканці отримують доступ до необхідної інформації та послуг без відвідування адміністративних установ [1], [2].

Програмне забезпечення для вебсайтів населених пунктів сприяє ефективній організації даних, забезпечуючи їх зручне використання та інтеграцію. Завдяки такому підходу налагоджується взаємодія між структурами, покращується координація надання послуг, а також забезпечується точність та узгодженість інформації [1], [2].

Для захисту даних використовуються сучасні заходи безпеки даних, які гарантують надійність програмного забезпечення для підтримки вебсайтів населених пунктів та її відповідність вимогам. Застосовуються регулярні перевірки, резервне копіювання і безпечне середовище зберігання даних для мінімізації ризиків [1], [2].

Географічні інформаційні карти, інтегровані в такі програмні рішення, надають органам влади зручний інструмент для планування інфраструктури та управління критичними послугами, а також подання інформації про туристичні місця населеного пункту. Завдяки цьому рішення ухвалюються на основі актуальної та точної інформації [1], [2].

Цифровізація процесів за допомогою програмного забезпечення для підтримки діяльності населених пунктів дозволяє спростити подання заявок на отримання дозволів чи ліцензій, роблячи їх доступними в онлайн форматі. Автоматизований процес обробки заявок скорочує час очікування та підвищує прозорість [1], [2].

Аналітичні інструменти, вбудовані у програмне забезпечення для підтримки діяльності населених пунктів, допомагають оцінювати ефективність роботи, прогнозувати майбутні потреби та виявляти аспекти, що потребують вдосконалення. Це дозволяє швидко реагувати на запити громади та покращувати якість послуг [1], [2].

Програмне забезпечення для вебсайтів населених пунктів змінює підхід до організації роботи місцевих структур та взаємодії з громадою. Завдяки таким рішенням керівники отримують можливість приймати обґрунтовані рішення, засновані на аналізі даних, що сприяє ефективному управлінню та координації навіть за умов віддаленої роботи [1], [2].

У сфері будівництва такі програмні платформи спрощують обробку документації та забезпечують зручність у відстеженні проектів. Подання електронних заявок дозволяє прискорити процеси затвердження та зробити їх доступними для громадян у режимі онлайн [1], [2].

Також програмне забезпечення для вебсайтів НП відіграє важливу роль у сфері охорони здоров'я, сприяючи моніторингу даних про епідемії, підтримці санітарних норм та впровадженню програм спостереження. Крім цього, автоматизація процесів у цій галузі допомагає у видачі ліцензій, дозволів і контролі відповідних об'єктів, що покращує управління загальним станом здоров'я населення [1], [2].

Програмне забезпечення для вебсайтів населених пунктів сприяє оптимізації планування та управління громадським транспортом, моніторингу інфраструктури, а також допомагає впорядковувати процеси, пов'язані з доступом до послуг і розв'язанням питань щодо обмежень руху чи ремонту доріг. Використання автоматизованих систем дозволяє уникнути помилок,

мінімізувати затримки, централізувати дані й підвищити якість обслуговування жителів [1], [2].

Програмне забезпечення для підтримки діяльності населених пунктів спрощують обробку документів, управління правовою інформацією та впровадження нових заходів з безпеки. Завдяки можливостям цифрового зберігання інформації працівники та мешканці можуть швидко отримувати доступ до необхідних даних і використовувати їх для вирішення різних завдань [1], [2].

Хоча такі програмні продукти найчастіше застосовуються в окремих сферах, їх можна адаптувати до потреб усіх структур місцевого управління, незалежно від їхньої спеціалізації. Модернізація діяльності стає ключовим елементом підвищення ефективності як у сфері охорони довкілля чи житлового господарства, так і в управлінні рекреаційними зонами чи розвитком локальної торгівлі [1], [2].

Перехід від використання паперової документації до сучасних хмарних програмних рішень значно спрощує процеси управління для місцевих органів влади. Такий перехід може бути викликом, але його впровадження дозволяє ефективно оптимізувати роботу та підвищити доступність інформації. Початковим етапом стає аналіз поточних процедур, що допомагає визначити, які з них доцільно перенести в цифрове середовище. При виборі хмарного рішення важливо враховувати надійність постачальника, його технічну підтримку та адаптованість до потреб користувачів. Успішна інтеграція передбачає навчання персоналу, що дозволяє забезпечити їхню готовність до роботи з новими технологіями [1], [2].

Проаналізуємо процес розробки програмного забезпечення для підтримки діяльності населених пунктів.

При розробці програмного забезпечення для підтримки діяльності населених пунктів, яке задовольняє сучасні вимоги управління містами, потрібно врахувати, що таке програмне забезпечення повинно поєднувати зручність використання та високі стандарти безпеки. Інтуїтивний інтерфейс

дозволяє легко освоїти систему навіть тим користувачам, які мають обмежений досвід роботи з технологіями. Простота дизайну та функціональність забезпечують економію часу, зосереджуючи увагу на основних завданнях [1], [2].

Крім зручності, пріоритетом стає захист даних. Ефективні заходи з безпеки даних дозволяють забезпечити безпеку інформації на всіх етапах роботи. Надійне шифрування, регулярний моніторинг та відповідність сучасним стандартам є ключовими елементами захисту. Такі програмні рішення дають змогу довіряти системі та забезпечують стабільність роботи органів місцевого самоврядування [1], [2].

Організація навчання та підтримки користувачів у рамках впровадження програмного забезпечення для вебсайтів населених пунктів забезпечує легкість інтеграції нових рішень у щоденну діяльність. Замість тривалих місяців навчання, співробітники швидко адаптуються до нових інструментів, отримуючи доступ до необхідної підтримки в процесі роботи. Такий підхід гарантує, що всі учасники процесу можуть скористатися технічною допомогою, що підвищує загальну ефективність системи [1], [2].

Впровадження інтегрованої платформи для управління НП дозволяє значно спростити адміністративні процеси. Перехід від застарілих паперових систем до сучасних цифрових рішень забезпечує оптимізацію роботи органів влади, полегшує управління ресурсами та покращує якість обслуговування громадян. Завдяки інтуїтивно зрозумілому інтерфейсу, надійному захисту даних та безперервній підтримці користувачів такі платформи сприяють прозорості та злагодженості дій усіх відділів [1], [2].

Програмні рішення для вебсайтів місцевих громад створюються для того, щоб забезпечити ефективне управління і підвищити рівень комунікації. Вони сприяють автоматизації рутинних завдань, зменшують адміністративне навантаження, покращують збір, аналіз та безпеку даних. Використання таких інструментів дозволяє досягти нових стандартів у роботі місцевих органів влади, роблячи їхню діяльність більш доступною та ефективною [1], [2].

Програмне забезпечення для вебсайтів населених пунктів створює можливості для автоматизації адміністративних процесів, скорочуючи необхідність у рутинній ручній роботі та дозволяючи персоналу зосереджуватися на пріоритетних завданнях. Такі рішення підвищують продуктивність і забезпечують кращу взаємодію між різними департаментами, що сприяє ефективнішій організації роботи. Завдяки інтеграції інструментів для управління документами та аналізу ключових показників діяльності, програмне забезпечення надає чітке уявлення про поточний стан справ у громаді. Особливу увагу приділено безпеці, щоб гарантувати надійний захист даних, особливо в умовах постійного розширення цифрових технологій та злочинів у галузі інформаційних технологій [1], [2].

Програмні рішення для управління міськими процесами можуть називатися по-різному, зокрема системами для місцевого самоврядування, технологіями для муніципальних послуг чи хмарними урядовими платформами. Незалежно від назви, мета таких рішень залишається однаковою та полягає у забезпеченні ефективності, прозорості та безпеки у функціонуванні місцевих органів влади [1], [2].

Сучасні вебсайти населених пунктів стали віртуальними міськими просторами, які забезпечують доступ до важливої інформації, підтримують спілкування, сприяють веденню бізнесу та зміцнюють довіру громади. Ці програмні платформи слугують своєрідними цифровими центрами, які відповідають на потреби мешканців і організовують взаємодію між ними та місцевою владою [3], [4].

Розробка програмного забезпечення для вебсайтів населених пунктів вимагає врахування численних аспектів, оскільки вони повинні охоплювати широкий спектр ініціатив і послуг. Зокрема, створення зручного і зрозумілого інтерфейсу стає справжнім викликом, коли необхідно інтегрувати численні функції, інформаційні блоки та повідомлення. Важливо не лише наповнити сайт інформацією, а й забезпечити його реальну функціональність, яка відповідає потребам кожного мешканця [3], [4].

Ефективність програмного забезпечення для вебсайтів населених пунктів залежить від того, як вони поєднують доступність для всіх груп населення з адаптацією до локальних особливостей. Розробники, які співпрацюють із муніципалітетами, створюють комплексні програмні рішення, що забезпечують не лише естетичний вигляд сайтів, але й їхню ефективну роботу для громади. Розробка вебсайтів для населених пунктів потребує зосередженості на спільній роботі з громадою. Від початкових етапів планування важливо залучати зацікавлених сторін для формування бачення та визначення ключових напрямків удосконалення. Проведення заходів, спрямованих на дослідження потреб, допомагає створювати ресурси, які відповідають очікуванням місцевих мешканців [3], [4]. Активне залучення громадян дозволяє глибше зрозуміти їхні запити. Вивчення думок через опитування, зустрічі або фокус-групи допомагає отримати корисну інформацію для оптимізації роботи сайту. У процесі тестування вмісту користувачі взаємодіють із прототипом, що дає змогу виявляти та вирішувати основні проблеми, забезпечуючи інтуїтивний і зручний інтерфейс [3], [4].

Додатково проводиться оцінка доступності програмного ресурсу, яка спрямована на виявлення аспектів, що потребують доопрацювання для задоволення потреб різних груп населення, включно з людьми з обмеженими можливостями. Постійний збір зворотного зв'язку гарантує, що вебсайт населеного пункту зможе еволюціонувати разом із потребами громади [3], [4].

Важливо враховувати унікальність кожної спільноти, її особливу культуру та очікування. Підхід до розробки таких сайтів має бути гнучким, акцентуючи увагу на характерних рисах та першочергових пріоритетах [3], [4].

Вебсайт для населених пунктів повинен бути спроектований таким чином, щоб забезпечити легкий доступ до актуальної інформації та можливість оперативно оновлювати її. Це дозволяє користувачам і організаторам подій бути в курсі змін і отримувати важливі оновлення в реальному часі. Наявність точних і своєчасних даних на сайті підвищує його авторитет і довіру серед мешканців [3], [4].

Для цього важливо налаштувати сайт населеного пункту таким чином, щоб персонал, який не має технічних знань, міг самостійно оновлювати контент і вносити зміни, не звертаючись до технічної підтримки. Використання простих в управлінні систем керування контентом дозволяє співробітникам ефективно працювати з вмістом і забезпечувати його актуальність без необхідності втручання розробників [3], [4]. Управління доступом на основі ролей дозволяє чітко визначити, хто має право на внесення змін, що забезпечує належний рівень контролю без потреби втручання фахівців з інженерії програмного забезпечення [3], [4].

Завдяки можливості надсилати оновлення в реальному часі, такі програмні системи дозволяють оперативно інформувати громадськість про важливі події, такі як зміни погоди або надзвичайні ситуації. Це сприяє швидкому реагуванню у випадках, коли інформація критична для безпеки населення [3], [4].

Програмне забезпечення має також надавати навчальні матеріали для співробітників, які працюють з сайтом, допомагаючи їм швидко освоїти інструменти для управління контентом і використовувати систему за призначенням [3], [4].

Програмне забезпечення вебсайтів населених пунктів має надавати актуальну та точну інформацію без затримок, зокрема, щодо змін у політиці, публічних оголошень, майбутніх подій або правових питань, що стосуються місцевої громади. Одним із важливих аспектів є забезпечення доступу до вебсайтів через мобільні пристрої. Зручність використання мобільної версії вебсайту населеного пункту та швидкість його роботи є критичними для користувачів, оскільки вони мають можливість отримувати інформацію в будь-який час та з будь-якого місця. Врахування вимог доступності також має велике значення, оскільки відповідність нормам може убезпечити від юридичних наслідків та штрафів [3], [4].

У процесі створення сайту населеного пункту необхідно враховувати різні потреби користувачів, зокрема тих, хто має обмеження у можливостях.

Важливо передбачити варіанти доступу до контенту вебсайту населеного пункту, наприклад, через стенограми або описові тексти для відео. Такі підходи можуть позитивно вплинути на пошукову оптимізацію, оскільки сайти, які забезпечують широкий доступ, отримують вищі позиції в результатах пошукових програмних систем, що допомагає досягти більшої аудиторії [3], [4].

Мобільна адаптивність також є важливим чинником у створенні ефективного вебсайту населеного пункту. Оскільки більшість користувачів отримує доступ до Інтернету через мобільні пристрої, потрібно забезпечити оптимальну роботу сайту на різних екранах. Для цього важливо працювати з досвідченими розробниками, щоб сайт був функціональним та зручним на будь-яких пристроях [3], [4].

Крім того, варто орієнтуватися на зручність користувацького досвіду, налаштовуючи інтерфейс та навігацію таким чином, щоб користувачам було легко знаходити потрібну інформацію. Інструменти для моніторингу та аналізу проблем у дизайні допоможуть своєчасно виявляти та усувати недоліки [3], [4].

При розробці вебсайтів для громад необхідно дотримуватися законодавчих вимог щодо доступності та конфіденційності. Це передбачає створення програмних платформ, доступних для всіх, включаючи людей з обмеженими можливостями, відповідно до стандартів, встановлених законодавством. Невиконання цих вимог може призвести до юридичних наслідків, включаючи позови та штрафи з боку органів, що контролюють доступність послуг та рівень інформування громадськості [3], [4].

Це питання є пріоритетним для державних установ, які займаються розробкою та обслуговуванням таких вебсайтів. Важливо, щоб всі громадяни мали рівний доступ до інформації, що надається через ці програмні платформи, а також до сучасних технологій, що допомагають покращити взаємодію з адміністрацією [3], [4].

Інтеграція штучного інтелекту в пошукові системи сайту населеного пункту дозволяє користувачам отримувати точну і релевантну інформацію, покращуючи ефективність пошуку та загальну якість взаємодії. Вебсайти населених пунктів можуть пропонувати різні функції самообслуговування, такі як можливість призначення зустрічей, завантаження важливих документів або подання заявок [3], [4].

Для ефективного функціонування вебсайтів міст та інших населених пунктів важливо забезпечити швидке реагування на запити користувачів і надавати чіткі інструкції для самостійного виконання завдань. Це включає надання доступу до покрокових керівництв, навчальних матеріалів та додаткових ресурсів, які сприяють кращому розумінню процесів і дозволяють громадянам виконувати необхідні дії без зовнішньої допомоги [3], [4].

Одним з основних аспектів функціонування вебсайтів населених є забезпечення прозорості і підзвітності в управлінні. Мешканці міста або іншого населеного пункту повинні мати доступ до інформації, що стосується діяльності органів влади, включаючи бюджети, політику та звіти про виконання завдань. Чітке представлення інформації про державні послуги та процеси прийняття рішень допомагає зменшити бюрократію і зміцнює довіру між громадянами та органами влади [3], [4].

Інтерактивні функції, такі як платформи для відкритих даних або розділи для громадських коментарів, сприяють залученню громадян до участі в управлінні і демократичних процесах. Важливо також уникати залежності від одного постачальника послуг, що дозволяє забезпечити гнучкість і стабільність у роботі вебсайту населеного пункту, підвищити їхню безпеку і знижувати витрати на підтримку [3], [4].

Місцеві органи влади все частіше звертаються до використання систем управління вмістом для підтримки своїх вебсайтів, де розробка та управління контентом здійснюються за допомогою фахівців з інформаційних технологій та програмістів. Вибір таких платформ набуває популярності, оскільки вони дозволяють організаціям здійснювати міграцію своїх сайтів на більш гнучкі та

масштабовані рішення. Різниця між комерційним програмним забезпеченням та рішеннями з відкритим кодом, розробленими на основі систем управління вмістом, є досить суттєвою. У той час як відкритий код забезпечує більшу автономність і гнучкість, він також не позбавлений ризиків, оскільки з часом можливе припинення підтримки або припинення діяльності розробників, що може вплинути на стабільність і функціональність ресурсу [3], [4].

При використанні відкритих рішень користувачі мають доступ до коду і можуть співпрацювати з іншими розробниками для покращення функціоналу або адаптації сайту до нових умов. Однак важливо розуміти, що робота з постачальниками має свої переваги та недоліки порівняно з розробкою спеціалізованих систем для конкретних потреб. Належна підтримка та обізнаність про можливості місцевих фахівців може стати вирішальним фактором у виборі оптимального шляху для розробки та підтримки вебсайтів населених пунктів [3], [4].

Під час розгляду можливих варіантів при виборі технологій розробки програмного забезпечення вебсайтів населених пунктів важливо звертати увагу на методи та рішення, які вони пропонують, і на те, як ці рішення можуть відповідати специфічним вимогам вебсайту конкретного населеного пункту. Кожен проект потребує ретельного планування, і для цього варто залучати організацію на самому початку. Усі важливі аспекти, такі як цілі, плани та обсяг проекту, повинні бути чітко сформульовані в запиті на пропозицію, що дозволить забезпечити якісне виконання робіт [3], [4].

Швидкість виконання адміністративних завдань, заповнення форм і отримання необхідної інформації є важливими для підвищення ефективності роботи користувачів вебсайтів населених пунктів. У разі можливості, форми повинні бути заповнені та надіслані безпосередньо через вебсайт, оскільки більшість людей не мають доступу до принтерів або не працюють у традиційних офісах. Ефективний аналіз та доступність даних є важливою частиною кожного проекту. Забезпечення того, щоб дані були доступними для всіх учасників процесу, значно сприяє успішному розвитку вебсайтів та їх

подальшій підтримці. Для розуміння користувацької поведінки та вподобань застосовуються різні аналітичні моделі. Вони допомагають оцінити ефективність форм, зокрема відстежувати кількість поданих форм та файлів, які потребують перегляду. Важливо також аналізувати відвідуваність цільових сторінок, що дозволяє зрозуміти, які сторінки користувачі вебсайтів населених пунктів відвідують найчастіше, а також вивчити, яким чином здійснюється доступ до сайту – через прямі переходи за посиланнями або пошукові системи та сервіси [3], [4].

Розуміння шляху користувача вебсайту населених пунктів та його поведінки дозволяє аналізувати, як саме відвідувачі рухаються по сайту та досягають кінцевої мети, що також є важливим для покращення користувацького досвіду. Крім того, дослідження локального пошуку допомагає визначити, які ключові слова найчастіше використовуються для пошуку на сайті [3], [4].

Час завантаження сторінок вебсайтів населених пунктів і показник відмов є важливими показниками, які вказують на ефективність сайту. Затримка у завантаженні сторінок може призводити до високих показників відмов, що негативно позначається на залученості користувачів і їхній взаємодії з сайтом [3], [4].

Вебсайти органів місцевої влади мають важливу роль у наданні якісної інформації та ресурсів для жителів. Вони повинні бути не лише джерелом корисної інформації, але й пропонувати реальні рішення для повсякденних потреб громадян. Оскільки люди очікують, що їхні міста будуть організованими, гостинними та добре керованими, так само вони чекають на сучасні, зручні та привабливі вебсайти, які відображають культурні цінності громади. Незалежно від технічного рівня користувачів, сучасні сайти мають бути інтуїтивно зрозумілими та приємними для використання. Це не тільки покращує взаємодію з громадянами, але й зміцнює соціальні зв'язки та може впливати на підвищення доходів місцевих органів влади [3], [4].

Відзначено, що програмне забезпечення для вебсайтів населених пунктів створюється для полегшення організації муніципальних процесів і вирішення адміністративних завдань. Показано, що програмне забезпечення для підтримки діяльності населених пунктів забезпечує оптимізацію роботи місцевих органів влади, дозволяючи підвищити їх ефективність та покращити якість наданих послуг. Завдяки впровадженню таких цифрових рішень, муніципальні структури можуть більш раціонально використовувати ресурси та скорочувати витрати часу на виконання повторюваних завдань. Відзначено, що програмні засоби для підтримки діяльності населених пунктів надають органам влади зручні інструменти для планування інфраструктури та управління критичними послугами, а також подання інформації про туристичні місця населеного пункту. Проаналізовано процес розробки програмного забезпечення для підтримки діяльності населених пунктів.

1.2 Аналіз програмного забезпечення для підтримки вебсайтів населених пунктів

Проаналізуємо програмне забезпечення для підтримки вебсайтів населених пунктів [5]-[11].

Програмне забезпечення CityGrows (рис. 1.1) призначене для автоматизації, оптимізації та цифровізації управлінських процесів в органах місцевого самоврядування. Воно створене, щоб допомогти населеним пунктам забезпечити ефективне управління, підвищити прозорість роботи та покращити взаємодію з громадянами. Основною метою цього програмного інструмента є спрощення та прискорення виконання адміністративних завдань шляхом переведення процесів у цифровий формат. Програмне забезпечення CityGrows підходить для населених пунктів, які прагнуть зробити свою діяльність більш ефективною, забезпечити якісніші послуги для громадян і сприяти їхній активній участі у житті громади [5], [9].

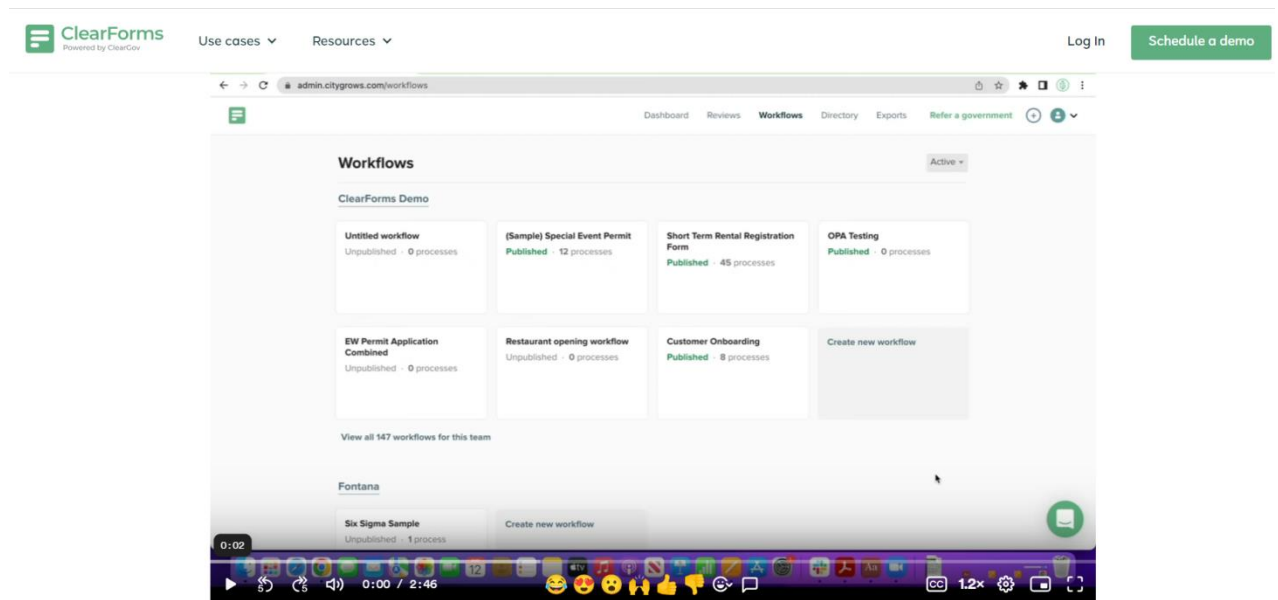


Рисунок 1.1 – Програмне забезпечення CityGrows [9]

Програмне забезпечення CityGrows дозволяє органам влади швидко адаптуватися до сучасних викликів та покращувати управління ресурсами. Програмне забезпечення CityGrows розроблено так, щоб відповідати потребам як невеликих громад, так і великих міст. Програмне забезпечення CityGrows може масштабуватися разом із розвитком громади, додаючи нові модулі чи функції за потреби [5], [9].

Програмне забезпечення не лише полегшує управління, але й враховує специфіку населених пунктів, таких як їхній розмір, складність інфраструктури чи культурні особливості. Це дозволяє створювати індивідуальні рішення для кожного населеного пункту, забезпечуючи ефективне використання ресурсів та підвищення задоволеності мешканців громади [5], [9].

Основними характеристиками програмного забезпечення CityGrows є такі [5], [9]:

- автоматизація процесів: програмне забезпечення CityGrows дозволяє переводити складні бюрократичні процедури, такі як видача дозволів, обробка заявок чи затвердження документів, у повністю автоматизований режим. Це зменшує час на виконання завдань, знижує кількість помилок і дозволяє

співробітникам адміністративних департаментів населених пунктів зосереджуватись на важливіших питаннях [5], [9];

– онлайн взаємодія з громадянами: програмне забезпечення CityGrows надає громадянам можливість подавати заявки, сплачувати збори чи податки, а також отримувати зворотний зв'язок онлайн. Завдяки цьому мешканці можуть легко взаємодіяти з органами влади, не виходячи з дому, що особливо важливо для віддалених районів [5], [9];

– забезпечення прозорості: CityGrows сприяє відкритості в управлінні. Громадяни можуть стежити за статусом своїх заявок у реальному часі, а також отримувати доступ до важливих даних, таких як бюджети, плани розвитку чи звіти. Це підвищує довіру до місцевої влади [5], [9];

– гнучкість налаштувань: програмне забезпечення CityGrows легко адаптується до потреб різних населених пунктів, незалежно від їхнього розміру чи структури управління. Користувачі можуть налаштовувати процеси відповідно до місцевих законодавчих вимог та особливостей [5], [9];

– аналітика та звітність: програмне забезпечення CityGrows забезпечує інструменти для збору даних, аналізу роботи та створення звітів. Це допомагає місцевій владі краще розуміти, які сфери потребують покращення, і приймати обґрунтовані рішення [5], [9];

– зниження витрат: завдяки цифровізації процесів та зменшенню паперової роботи, програмне забезпечення CityGrows допомагає суттєво скоротити адміністративні витрати [5], [9];

– інтеграція з іншими системами: програмне забезпечення CityGrows підтримує інтеграцію з іншими цифровими платформами, такими як фінансові системи, платформи електронного документообігу, геоінформаційні системи або бази даних мешканців. Це створює єдину екосистему управління, яка дозволяє працювати з різними типами інформації в межах одного інтерфейсу програмне забезпечення CityGrows [5], [9];

– зручний інтерфейс і мобільний доступ: програмне забезпечення CityGrows має інтуїтивно зрозумілий дизайн, який дозволяє легко

користуватися ним навіть людям без технічних навичок. Воно також підтримує мобільні пристрої, що дає змогу працівникам та громадянам використовувати його на ходу [5], [9];

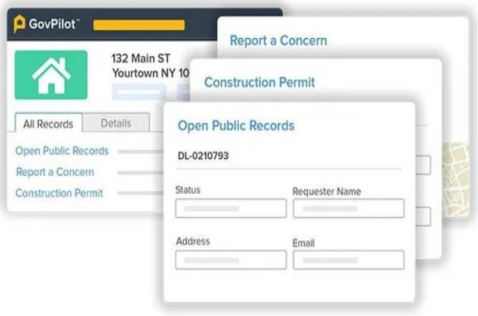
– оптимізація часу: завдяки автоматизації завдань, таких як обробка запитів, створення документів чи проведення платежів, CityGrows значно скорочує час виконання адміністративних процедур. Це важливо як для працівників місцевої влади, так і для мешканців [5], [9].

Серед недоліків програмного забезпечення CityGrows можна відзначити обмежену функціональність для великих міст, обмежені можливості налаштувань та потреба у навчанні персоналу. Впровадження CityGrows потребує навчання співробітників місцевих адміністрацій. Цей процес може бути тривалим і вимагати додаткових витрат часу та ресурсів. Крім того, як і всі хмарні платформи, програмне забезпечення CityGrows має ризик витоку або несанкціонованого доступу до конфіденційної інформації. Це викликає занепокоєння, особливо коли йдеться про дані громадян та фінансову інформацію. У випадках, коли в організації вже використовуються інші спеціалізовані платформи, інтеграція CityGrows може виявитися складною або потребувати значних ресурсів [5], [9].

Програмне забезпечення GovPilot (рис. 1.2) представляє собою інноваційну хмарну платформу, створену для оптимізації управління населеними пунктами. Програмне забезпечення GovPilot забезпечує широкий спектр функцій, які допомагають місцевим органам влади автоматизувати процеси, покращити комунікацію з громадянами та підвищити ефективність адміністративної роботи. Завдяки своїм можливостям GovPilot допомагає модернізувати управління містами та селищами, забезпечуючи прозорість і доступність послуг. GovPilot є потужним інструментом для розвитку громад і підвищення якості життя мешканців. Вона сприяє модернізації муніципального управління, покращуючи доступ до послуг, мінімізуючи бюрократію та забезпечуючи відкритість у взаємодії між громадянами та владою [5], [10].



[Solutions](#)
[Pricing](#)
[Resources](#)
[About Us](#)
[Get Demo](#)
[Sign in](#)





Unify

Eliminate Data Silos

GovPilot's PropertyProfile eliminates data and communications silos by displaying parcel level detail including cross departmental records. Data updates in real-time and is accessible from anywhere with a WIFI connection, 24/7.

[Learn how to Unify Data with GovPilot >>](#)



Engage

Deliver Convenience

In today's world, constituents expect convenient digital services and the ability to engage with their government. Digital forms, automated workflows, and mobile apps - [GovAlert](#) and [GovInspect](#) - deliver ease of use, transparency, and responsiveness.

[Learn How to Engage with GovPilot >>](#)

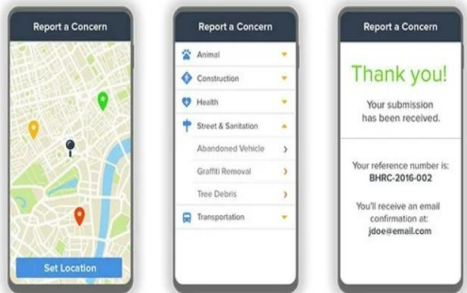


Рисунок 1.2 – Програмне забезпечення GovPilot [10]

Програмне забезпечення GovPilot має такі особливості [5], [10]:

- автоматизація адміністративних процесів: програмне забезпечення GovPilot дозволяє автоматизувати рутинні адміністративні завдання, такі як видача дозволів, управління заявами, обробка ліцензій та контроль за виконанням регуляторних вимог. Це зменшує навантаження на працівників місцевої влади та скорочує час виконання завдань [5], [10];
- цифровізація документообігу: програмне забезпечення GovPilot дає змогу перевести документообіг у цифровий формат, що забезпечує швидкий доступ до важливої інформації, зменшує ризик втрати документів і підвищує загальну ефективність роботи [5], [10];
- покращення обслуговування громадян: програмне забезпечення GovPilot пропонує користувачам зручний інтерфейс для подання заявок, оплати послуг, подання скарг і отримання інформації в режимі реального часу. Це сприяє зручнішій взаємодії громадян із місцевою владою [5], [10];

– управління фінансами та бюджетом: програмне забезпечення GovPilot допомагає місцевим органам влади оптимізувати управління бюджетом, забезпечуючи прозоре використання коштів, контроль витрат і фінансове планування [5], [10];

– аналітика та звітність: GovPilot надає інструменти для аналізу даних та формування звітів, що дає змогу органам влади приймати обґрунтовані рішення на основі актуальної інформації [5], [10];

– підтримка комунікації між громадянами та владою: програмне забезпечення GovPilot спрощує обмін інформацією між мешканцями та місцевою владою, дозволяючи повідомляти про проблеми, отримувати відповіді та відстежувати статус звернень у режимі реального часу [5], [10];

– гнучкість та адаптивність: програмне забезпечення GovPilot розроблено таким чином, щоб відповідати специфічним потребам кожного населеного пункту. Завдяки модульній структурі, його можна налаштувати під конкретні завдання місцевих адміністрацій [5], [10];

– інтеграція з іншими системами та платформами: програмне забезпечення GovPilot легко інтегрується з уже існуючими системами місцевої адміністрації, такими як платіжні системи, інформаційні системи або рішення для управління зверненнями громадян. Це дозволяє органам влади створювати єдину екосистему, що об'єднує всі аспекти управління та підтримки діяльності населеного пункту [5], [10];

– інструменти для управління надзвичайними ситуаціями: програмне забезпечення GovPilot підтримує функції оповіщення громадян у разі надзвичайних ситуацій, таких як природні катаклізми, техногенні аварії чи інші критичні події. Це дозволяє оперативно інформувати мешканців і забезпечувати їхню безпеку [5], [10];

– модулі для управління нерухомістю та землекористуванням: програмне забезпечення GovPilot включає інструменти для управління даними про нерухомість, земельні ділянки та зонування. Це допомагає місцевим

органам влади краще контролювати процеси, пов'язані з забудовою, орендою чи продажем земель [5], [10];

- мобільна доступність: програмне забезпечення GovPilot доступно через вебсайт та мобільні версії, що дозволяє користувачам отримувати доступ до її функцій із будь-якого пристрою. Це особливо корисно для працівників, які виконують польові завдання, або для громадян, які хочуть отримати послуги дистанційно [5], [10];

- забезпечення прозорості: використання GovPilot підвищує прозорість діяльності місцевих органів влади, забезпечуючи доступ громадян до інформації про проекти, витрати та діяльність адміністрації [5], [10].

Серед недоліків програмного забезпечення GovPilot можна відзначити складність адаптації для невеликих муніципалітетів з обмеженими ресурсами, адже інтеграція може вимагати значних фінансових витрат та технічної підготовки персоналу. Деякі користувачі також зазначають, що інтерфейс може бути перевантаженим для початківців, що ускладнює швидке освоєння платформи. Залежність від стабільного Інтернет з'єднання може створювати труднощі у віддалених регіонах з поганою інфраструктурою. Додатково, специфічні функції, необхідні окремим муніципалітетам, можуть потребувати індивідуального налаштування, що збільшує час впровадження [5], [10].

Програмне забезпечення FREY (рис. 1.3) є інноваційною програмою, створеною для оптимізації управлінських процесів у населених пунктах. Його основна мета — забезпечити ефективне надання адміністративних послуг, підтримку комунікації між органами влади та громадами, а також сприяти впровадженню цифрових рішень у повсякденну діяльність муніципалітетів. FREY поєднує зручність використання, функціональність і високу адаптивність, що дозволяє легко інтегрувати його в структуру управління будь-якого населеного пункту [5], [11]. FREY також забезпечує зручність для жителів населеного пункту, надаючи їм доступ до онлайн сервісів, таких як реєстрація заявок, перегляд стану їх виконання, оплата послуг та отримання інформації про діяльність місцевої влади [5], [11].



Software Solutions Since 1974

FREY Municipal Software is a team of public sector market leaders with extensive firsthand experience in our clients business. FMS has spent decades developing

Рисунок 1.3 – Програмне забезпечення FREY [11]

Однією з ключових переваг FREY є автоматизація адміністративних процесів, таких як обробка запитів громадян, управління дозволами, збором податків чи іншими фінансовими операціями. Це дозволяє значно скоротити час виконання завдань і мінімізувати помилки, пов'язані з ручним введенням даних. Завдяки інтегрованим інструментам аналітики та звітності, програмне забезпечення допомагає органам влади приймати обґрунтовані рішення на основі реальних даних [5], [11].

Іншою важливою можливістю програмного забезпечення FREY є підтримка проектів сталого розвитку, таких як моніторинг екологічного стану, управління інфраструктурою та ресурсами, а також впровадження розумних технологій для підвищення якості життя в населених пунктах. Програмне забезпечення може бути адаптоване до конкретних потреб муніципалітетів, що робить його універсальним інструментом для забезпечення ефективного управління [5], [11].

Програмне забезпечення FREY має такі особливості [5], [11]:

– адаптивність та модульна структура: програмне забезпечення FREY побудоване за модульним принципом, що дозволяє налаштовувати функціональність відповідно до специфічних потреб конкретної громади. Користувачі можуть додавати або видаляти функції, забезпечуючи оптимальну конфігурацію для своїх завдань [5], [11];

– комплексне управління комунальними послугами: програмне забезпечення FREY підтримує управління інформацією про клієнтів, облік платежів, контроль споживання та автоматичну генерацію звітів. Це спрощує роботу з комунальними платежами та сприяє прозорості у фінансових операціях [5], [11];

– інтегроване бюджетне управління: програмне забезпечення FREY підтримує управління фондами та ведення бухгалтерського обліку. У режимі реального часу обробляються транзакції, оновлюються записи та виконуються перевірки для забезпечення точності даних. Ця функціональність відповідає вимогам як державних установ, так і неприбуткових організацій [5], [11];

– автоматизація обліку заробітної плати: програмне забезпечення FREY інтегрується з бюджетним модулем і дозволяє автоматично обробляти зарплатні транзакції, розподіляти витрати та створювати відповідну фінансову звітність. Це сприяє зниженню ризику помилок і економії часу [5], [11];

– зручність використання та інтеграція з іншими платформами: програмне забезпечення FREY підтримує інтеграцію з популярними інструментами аналізу, такими як Excel і Crystal Reports, що дозволяє створювати звіти та проводити детальний аналіз фінансових даних [5], [11];

– масштабованість для різних типів операцій: програмне забезпечення FREY може ефективно підтримувати як невеликі спільноти, так і великі адміністрації, адаптуючись до обсягів роботи та складності операцій [5], [11];

– реальний час і високий рівень точності: усі модулі програмного забезпечення FREY працюють у режимі реального часу, що гарантує миттєве оновлення даних та їхню актуальність. Це особливо важливо для точного фінансового планування та моніторингу ресурсів [5], [11];

– прозорість та звітність: програмне забезпечення FREY надає інструменти для створення звітів, що відповідають стандартам, забезпечуючи прозорість у використанні бюджетних коштів і комунальних ресурсів [5], [11];

– спрямованість на потреби громади: функціонал FREY орієнтований на вирішення повсякденних завдань муніципальних установ, таких як

управління комунальними платежами, фінансовий контроль і обробка зарплатних даних. Це дозволяє підвищити ефективність роботи адміністрацій та покращити якість обслуговування населення [5], [11].

Серед недоліків програмного забезпечення FREY можна відзначити певну складність початкового налаштування, що може вимагати додаткових ресурсів і часу для навчання персоналу. Крім того, висока залежність від модульної структури іноді ускладнює інтеграцію нових функцій у вже існуючі системи. Певним обмеженням є також необхідність регулярних оновлень для забезпечення сумісності з сучасними технологіями, що може спричиняти тимчасові перебої у роботі. Ще однією проблемою може стати складність адаптації до специфічних вимог деяких муніципалітетів, якщо вони мають унікальні процедури або нестандартні операційні процеси [5], [11].

Порівняльну характеристику програмного забезпечення для підтримки вебсайтів населених пунктів наведено у таблиці 1.1.

Таблиця 1.1 – Порівняння програмного забезпечення для підтримки вебсайтів населених пунктів

Критерій порівняння	CityGrows [9]	GovPilot [10]	FREY [11]
Адаптація до специфічних потреб населеного пункту	+	+—	+—
Низька вартість	—	—	+—
Підтримка мобільних пристроїв	+	+	—
Можливість створення вебсайтів для населених пунктів	+	+	+—
Підтримка опису туристичних місць	+	+—	—
Масштабованість	+—	+	+
Зручність налаштування	+	+—	+—

За результатами проведеного аналізу можна зробити висновок, що у наш час існує досить багато програмних засобів для підтримки вебсайтів населених пунктів, зокрема, для подання інформації про туристичні місця. Проте деякі програмні засоби та вебсайти туристичних місць населених пунктів характеризуються недостатньо широким функціоналом. Також поширеною проблемою є відсутність актуальної інформації або складність її оновлення через застарілий або незручний інтерфейс, внаслідок чого туристи можуть отримувати застарілі дані про події, об'єкти чи послуги. Відсутність адаптації вебсайтів туристичних місць населених пунктів до мобільних пристроїв унеможливорює комфортне використання ресурсу з телефонів чи планшетів, що є важливим у сучасному світі, де мобільний Інтернет займає значну частку обсягу передачі даних. Ще одним недоліком може стати відсутність багатомовної підтримки, що обмежує доступ до сайту для іноземних туристів. Крім того, ускладнений інтерфейс або незрозуміла навігація знижують задоволеність користувачів і роблять пошук потрібної інформації тривалим і виснажливим [5]-[11].

Тому актуальною є розробка програмного забезпечення вебсайту туристичних місць населеного пункту.

При розробці програмного забезпечення вебсайту туристичних місць населеного пункту необхідно забезпечити такі функціональні вимоги:

- підтримка можливості подання узагальненої інформації у вигляді списку про туристичні місця населеного пункту;
- можливість ознайомлення користувача з детальною інформацією про туристичні місця населеного пункту;
- можливість надання інформації користувачу про туристичні місця у текстовому вигляді та за допомогою зображень;
- можливість додавання та видалення інформації про туристичні місця адміністратором системи.

1.3 Висновки за розділом 1

Відзначено, що програмне забезпечення для вебсайтів населених пунктів створюється для полегшення організації муніципальних процесів і вирішення адміністративних завдань. Показано, що програмне забезпечення для підтримки діяльності населених пунктів забезпечує оптимізацію роботи місцевих органів влади, дозволяючи підвищити їх ефективність та покращити якість наданих послуг. Завдяки впровадженню таких цифрових рішень, муніципальні структури можуть більш раціонально використовувати ресурси та скорочувати витрати часу на виконання повторюваних завдань. Відзначено, що програмні засоби для підтримки діяльності населених пунктів надають органам влади зручні інструменти для планування інфраструктури та управління критичними послугами, а також подання інформації про туристичні місця населеного пункту. Проаналізовано процес розробки програмного забезпечення для підтримки діяльності населених пунктів.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато програмних засобів для підтримки вебсайтів населених пунктів, зокрема, для подання інформації про туристичні місця. Проте деякі програмні засоби та вебсайти туристичних місць населених пунктів характеризуються недостатньо широким функціоналом. Також поширеною проблемою є відсутність актуальної інформації або складність її оновлення через застарілий або незручний інтерфейс, внаслідок чого туристи можуть отримувати застарілі дані про події, об'єкти чи послуги. Відсутність адаптації вебсайтів туристичних місць населених пунктів до мобільних пристроїв унеможливорює комфортне використання ресурсу з телефонів чи планшетів, що є важливим у сучасному світі, де мобільний Інтернет займає значну частку обсягу передачі даних. Ще одним недоліком може стати відсутність багатомовної підтримки, що обмежує доступ до сайту для іноземних туристів. Крім того, ускладнений інтерфейс або незрозуміла навігація знижують задоволеність користувачів і роблять пошук потрібної інформації тривалим і

виснажливим. Тому актуальною є розробка програмного забезпечення вебсайту туристичних місць населеного пункту.

Сформульовано функціональні вимоги до програмного забезпечення вебсайту туристичних місць населеного пункту.

2 МАТЕРІАЛИ І МЕТОДИ

2.1 Вибір мови програмування

При розробці програмного забезпечення вебсайту туристичних місць населеного пункту було використано мову програмування JavaScript [12], [13].

JavaScript є мовою програмування, що відповідає специфікації ECMAScript і належить до високорівневих мов, її відрізняє наявність синтаксису на основі дужок, гнучкої обробки тексту, підтримки об'єктів та загальних функцій [12], [13].

Сьогодні JavaScript відіграє ключову роль у веброботці поряд із HTML і CSS. Майже всі сучасні вебсайти використовують JavaScript для забезпечення інтерактивного користувацького інтерфейсу та впровадження стороннього функціоналу. Завдяки інтеграції мови програмування JavaScript у найпопулярніші веббраузери, код виконується безпосередньо на пристроях користувачів [12], [13].

Ця мова дозволяє впроваджувати структурні рішення для програмування, а також функціонал, орієнтований на події. JavaScript активно використовується для роботи з об'єктною моделлю документа і підтримує засоби для обробки даних, таких як текст, часові мітки та різноманітні структури. Специфікація ECMAScript охоплює як вхідні, так і вихідні операції, що включають графіку, роботу з мережею або збереження інформації. Самі ж браузери та операційні середовища забезпечують доступ до необхідних інтерфейсів вводу-виводу через інтерфейси прикладного програмування (Application Programming Interface, API) [12], [13].

Спочатку JavaScript функціонував виключно у браузерах для забезпечення вебфункціоналу, проте сучасні технології дозволяють застосовувати цю мову програмування й поза межами вебсередовища. Тепер мову JavaScript використовують для серверної розробки, створення окремих програм та інших системних компонентів. Хоча JavaScript і Java мають схожу назву та певні схожості в структурі найменувань, ці дві мови програмування

суттєво відрізняються одна від одної за своїми принципами та сферою використання [12], [13].

Обґрунтування вибору мови програмування для розробки програмного забезпечення вебсайту туристичних місць населеного пункту наведено у таблиці 2.1.

Таблиця 2.1 – Обґрунтування вибору мови програмування для розробки програмного забезпечення вебсайту туристичних місць населеного пункту

Критерій порівняння мов програмування	Мова програмування		
	JavaScript	Python	Java
Багатофункціональність	+	+	+—
Велика кількість фреймворків і бібліотек для веброзробки	+	+	—
Здатність працювати як на стороні клієнта	+	—	—
Зручність для створення ПЗ вебсайту туристичних місць населеного пункту	+	—	—
Активна спільнота розробників	+	+	+—

Отже, для реалізації програмного забезпечення вебсайту туристичних місць населеного пункту обрано мову програмування JavaScript через її багатофункціональність, популярність і здатність задовольнити сучасні вимоги до веброзробки. JavaScript є однією з основних технологій для створення динамічних та інтерактивних вебсайтів, завдяки своїй клієнтській природі вона дозволяє створювати інтуїтивний і швидкий користувацький інтерфейс. Іншою суттєвою перевагою мови JavaScript є її здатність працювати як на стороні клієнта, так і на стороні сервера, що дозволяє створювати повноцінні вебзастосунки, де клієнтська та серверна частини

реалізовані однією мовою. Такий підхід спрощує розробку, зменшує витрати на навчання команди і забезпечує краще взаєморозуміння між розробниками.

2.2 Вибір фреймворку для створення програмного забезпечення вебсайту туристичних місць населеного пункту

В якості фреймворку для створення програмного забезпечення вебсайту туристичних місць населеного пункту було обрано React [14], [15].

Фреймворк React є відкритою JavaScript-бібліотекою, що використовується для ефективного вирішення завдань у сфері веброзробки. Зі зростанням популярності соціальних платформ виникла потреба у створенні продуктивних та гнучких вебзастосунків, здатних обробляти великі обсяги динамічних даних. React орієнтований на забезпечення швидкого та масштабованого процесу розробки, дозволяючи взаємодіяти з інтерфейсом користувача без необхідності додаткових витрат ресурсів. Його структура відповідає архітектурному шаблону Model–view–controller (MVC), де він може використовуватись як частина представлення. Крім того, React легко інтегрується з іншими JavaScript-бібліотеками або комплексними фреймворками. Ця бібліотека забезпечує зручну розробку як простих додатків, так і складних зовнішніх інтерфейсів, що значно розширює її функціональні можливості [14], [15].

Основою React є використання компонентів, які часто створюються з допомогою JavaScript Syntax eXtension (JSX) – спеціального синтаксису, що поєднує можливості JavaScript та XML-подібного коду. JSX, хоч і нагадує PHP, є результатом вдосконалення інструментів на базі XHP від Facebook. Однак розробка не обмежується лише JSX, оскільки для роботи з React також застосовується чистий JavaScript [14], [15].

Завдяки своїй сумісності та потужним можливостям React забезпечує гнучкість у проєктах різного масштабу, а додаткові інструменти, як-от Redux,

допомагають оптимізувати управління станом програми, що робить розробку ще більш ефективною [14], [15].

Обґрунтування вибору фреймворку для створення програмного забезпечення вебсайту туристичних місць населеного пункту наведено у таблиці 2.2.

Таблиця 2.2 – Обґрунтування вибору фреймворку для створення програмного забезпечення вебсайту туристичних місць населеного пункту

Критерій порівняння фреймворків	Фреймворки		
	React	Svelte	Vue.js
Висока продуктивність	+	+-	+-
Гнучкість	+	+-	+
Потужні інструменти для веброзробки	+	+-	+
Зручність створення адаптивних вебсайтів	+	+-	+-
Простота підтримки коду	+	-	+

Таким чином, для створення програмного забезпечення вебсайту туристичних місць населеного пункту обрано фреймворк React завдяки його високій продуктивності, гнучкості та потужним засобам веброзробки. Гнучка архітектура React дозволяє створювати модульні компоненти, які можна легко повторно використовувати в різних частинах вебсайту, що скорочує час розробки та спрощує підтримку коду. React має багату екосистему інструментів і бібліотек, які спрощують реалізацію складних функцій, що дозволяє легко створити багатосторінкову структуру вебсайту з можливістю швидкого переходу між сторінками без перезавантаження, а також ефективно обробляти дані, що важливо для відображення інформації в реальному часі.

2.3 Висновки за розділом 2

Для реалізації програмного забезпечення вебсайту туристичних місць населеного пункту обрано мову програмування JavaScript через її багатофункціональність, популярність і здатність задовольнити сучасні вимоги до веброзробки. JavaScript є однією з основних технологій для створення динамічних та інтерактивних вебсайтів, завдяки своїй клієнтській природі вона дозволяє створювати інтуїтивний і швидкий користувацький інтерфейс. Іншою суттєвою перевагою мови JavaScript є її здатність працювати як на стороні клієнта, так і на стороні сервера, що дозволяє створювати повноцінні вебзастосунки, де клієнтська та серверна частини реалізовані однією мовою. Такий підхід спрощує розробку, зменшує витрати на навчання команди і забезпечує краще взаєморозуміння між розробниками.

Для створення програмного забезпечення вебсайту туристичних місць населеного пункту обрано фреймворк React завдяки його високій продуктивності, гнучкості та потужним засобам веброзробки. Гнучка архітектура React дозволяє створювати модульні компоненти, які можна легко повторно використовувати в різних частинах вебсайту, що скорочує час розробки та спрощує підтримку коду. React має багату екосистему інструментів і бібліотек, які спрощують реалізацію складних функцій, що дозволяє легко створити багатосторінкову структуру вебсайту з можливістю швидкого переходу між сторінками без перезавантаження, а також ефективно обробляти дані, що важливо для відображення інформації в реальному часі.

3 ОПИС ПРОГРАМИ

3.1 Структура програмного забезпечення вебсайту туристичних місць населеного пункту

Структуру програмного забезпечення вебсайту туристичних місць населеного пункту наведено на рис. 3.1.

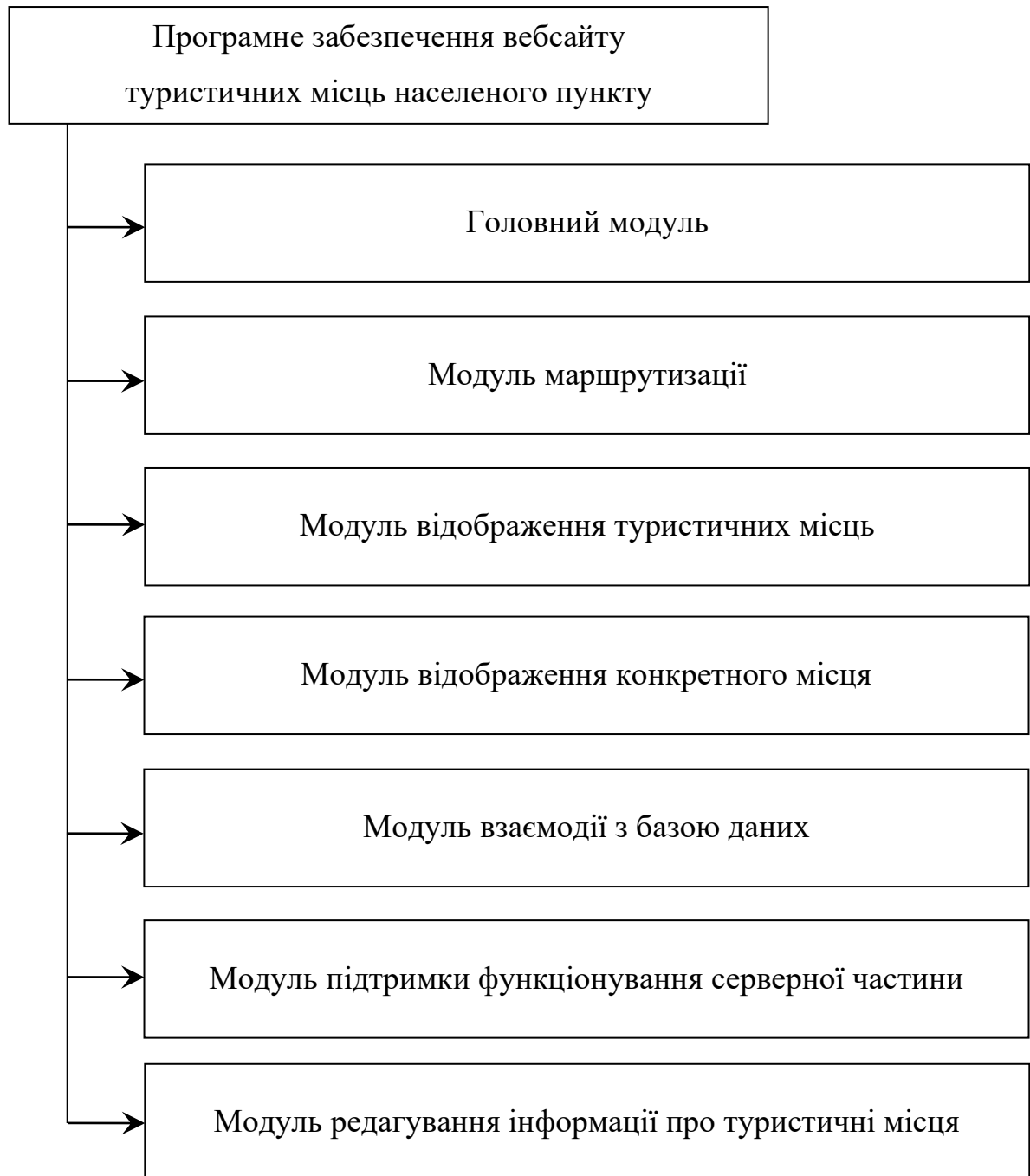


Рисунок 3.1 – Структура програмного забезпечення вебсайту туристичних місць населеного пункту

Структура програмного забезпечення вебсайту туристичних місць населеного пункту побудована на основі модульного підходу, що дозволяє зручно організувати код, спростити підтримку та забезпечити розширюваність функціоналу. Програмне забезпечення складається з кількох основних модулів, кожен з яких виконує певну роль у роботі вебсайту.

Головний модуль програмного забезпечення вебсайту туристичних місць населеного пункту відповідає за запуск програми та відображення сторінок. У цьому модулі відбувається ініціалізація інтерфейсу, а також підключення маршрутизації для переходу між різними сторінками. Він забезпечує відображення верхньої та нижньої частини сайту, а також основного контенту, який змінюється залежно від URL-шляху.

Модуль маршрутизації програмного забезпечення вебсайту туристичних місць населеного пункту визначає, який контент має бути відображено на сторінці, залежно від введеного шляху. Цей компонент керує динамічним завантаженням сторінок, передаючи необхідні параметри для відображення загального списку туристичних місць або окремого об'єкта.

Відображення інформації про туристичні місця забезпечується модулем, що обробляє масив даних і створює розмітку для кожного місця. Цей модуль відповідає за генерацію блоків із туристичними місцями та навігацію між загальним списком і детальними сторінками.

Модуль відображення конкретного місця займається відображенням конкретного об'єкта, перевіряючи наявність даних у базі та виводячи відповідний контент або повідомлення про відсутність інформації.

Модуль взаємодії з базою даних програмного забезпечення вебсайту туристичних місць населеного пункту виконує операції зі зчитування та запису інформації у базу даних. Він забезпечує взаємодію між програмним забезпеченням і даними, збереженими у базі даних, дозволяючи підтримувати актуальність інформації на вебсайті.

Модуль підтримки функціонування серверної частини відповідає за обробку запитів від клієнтів. Він об'єднує серверну логіку та маршрути для

API, що дозволяє налаштувати зручну взаємодію між клієнтською та серверною частинами вебсайту. Сховище стану, побудоване за допомогою централізованого управління, забезпечує збереження даних про туристичні місця та методи їх оновлення. Це дозволяє ефективно управляти змінами у додатку та підтримувати синхронізацію даних між різними компонентами.

Модуль редагування інформації про туристичні місця відповідає за додавання нових об'єктів до бази даних, а також редагування відповідної інформації. У ньому реалізується логіка для створення місць і виконання певних дій при завантаженні сторінки. Цей модуль об'єднується з контейнером, який передає необхідні дані та функції для роботи з новими елементами.

Завдяки такій структурі програмне забезпечення вебсайту туристичних місць населеного пункту працює стабільно та ефективно, а модульна організація дозволяє спростувати його розширення та підтримку в майбутньому.

3.2 Функціонування програмного забезпечення вебсайту туристичних місць населеного пункту

Функціонування програмного забезпечення вебсайту туристичних місць населеного пункту подамо за допомогою схеми, зображеної на рис. 3.2. Для функціонування програмного забезпечення вебсайту туристичних місць необхідно забезпечити коректну роботу сервера та наявність бази даних. Після запуску програми у браузері відкривається сторінка з переліком доступних місць. Користувачі можуть переглядати інформацію про вже додані об'єкти, переходячи на сторінки конкретних місць. Для цього достатньо обрати необхідний пункт зі списку, після чого відкриється сторінка з детальним описом об'єкта.



Рисунок 3.2 – Схема функціонування програмного забезпечення вебсайту туристичних місць населеного пункту

Варіанти використання користувачем програмного забезпечення вебсайту туристичних місць населеного пункту подано за допомогою діаграми прецедентів (рис. 3.3).

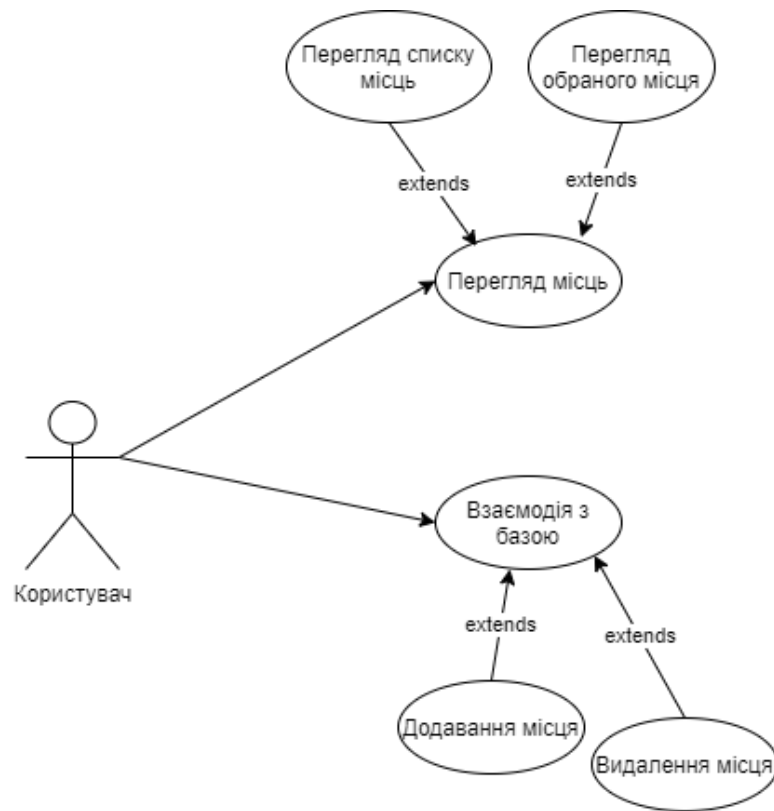


Рисунок 3.3 – Діаграма прецедентів програмного забезпечення вебсайту туристичних місць населеного пункту

Взаємодію між компонентами вебсайту туристичних місць населеного пункту у процесі його функціонування подано за допомогою схеми, зображеної на рис. 3.4. Сервер забезпечує обробку запитів, які надсилаються до модуля взаємодії з базою даних, що дозволяє отримувати або змінювати необхідну інформацію. Локальне сховище проєкту управляється через єдиний редьюсер, який містить дані про туристичні місця та методи для їх обробки. Функціонування вебсайту базується на роботі вебсторінок. Перша сторінка призначена для відображення загального списку туристичних пам'яток. Кожен об'єкт представлений зображенням та короткою інформацією і дозволяє перейти до детальної сторінки конкретного місця. Друга сторінка містить детальну інформацію про вибрану пам'ятку, включаючи більше зображень і розширений опис. Вся інформація для обох сторінок надходить з бази даних, після чого обробляється для відображення відповідного контенту.

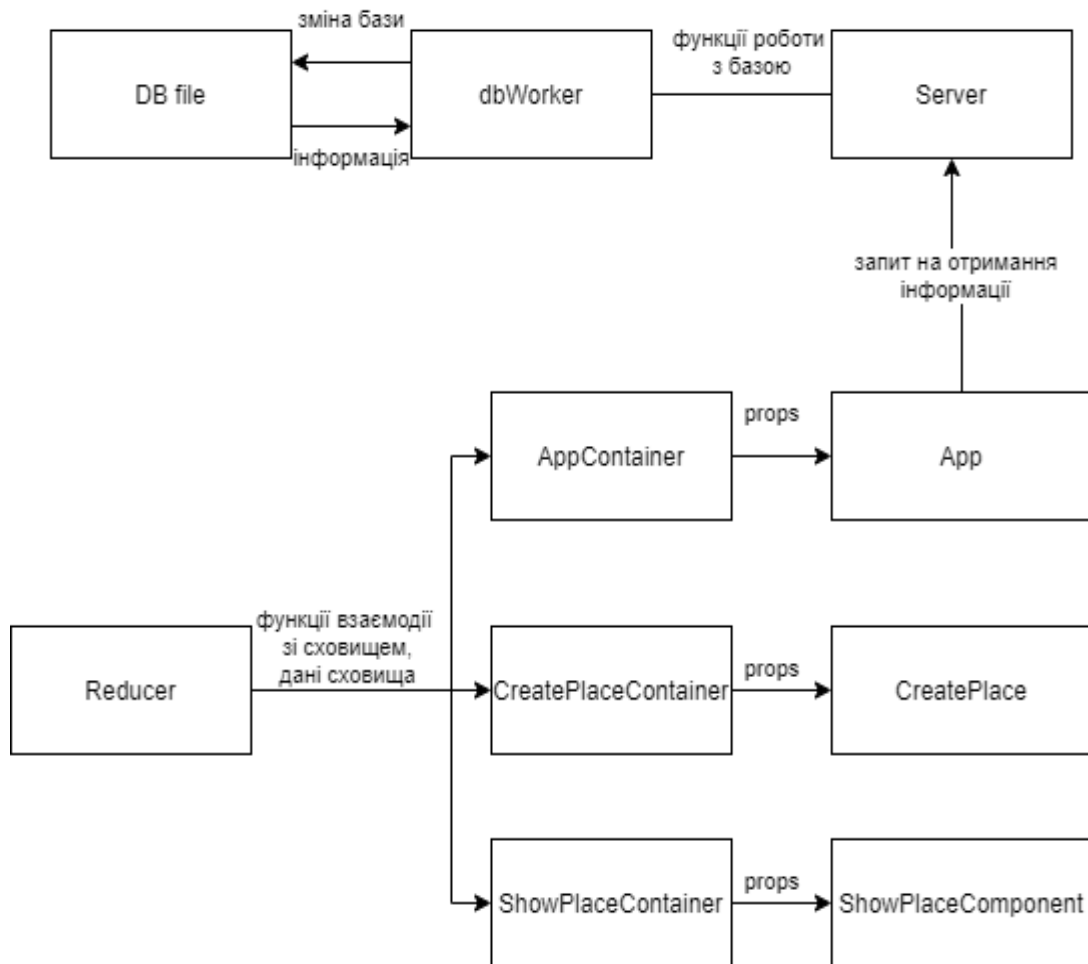


Рисунок 3.4 – Схема взаємодії між компонентами вебсайту туристичних місць населеного пункту

За навігацію між сторінками відповідає маршрутизатор, який перенаправляє користувача на необхідний розділ сайту залежно від обраного шляху. Контент кожної сторінки формується на основі окремих компонентів, які відповідають за відтворення основних елементів інтерфейсу. Таким чином, структура програмного забезпечення забезпечує ефективне відображення та оновлення даних про туристичні місця населеного пункту.

3.3 Рішення щодо зберігання даних програмного забезпечення вебсайту туристичних місць населеного пункту

Для зберігання даних про туристичні об'єкти використовується файл типу json. У відповідному файлі зберігається інформація про туристичні місця

населеного пункту, зокрема, відповідні зображення, коротка та повна інформація про туристичне місце, посилання на вебсайт, адреса місця та ін. Структура json файлу наведена у табл. 3.1.

Таблиця 3.1 – Структура json файлу для зберігання даних вебсайту туристичних місць населеного пункту

Назва поля	Призначення
imagS	Основне зображення туристичного місця НП
summary	Короткий опис туристичного місця НП
description	Детальний опис туристичного місця НП
title	Назва туристичного місця
galleryImages	Набір зображень туристичного місця
link	Фрагмент посилання для формування повної адреси сторінки туристичного місця на сайті
websiteUrl	Посилання на офіційний вебсайт туристичного місця (за наявності)
location	Розташування туристичного місця

3.4 Проєктування інтерфейсу програмного забезпечення вебсайту туристичних місць населеного пункту

Проєктування інтерфейсу взаємодії користувача з програмним забезпеченням вебсайту туристичних місць населеного пункту виконано з урахуванням вимог технічного завдання. При розробленні програмного забезпечення вебсайту туристичних місць населеного пункту враховані функціональні вимоги до програми:

- підтримка можливості подання узагальненої інформації у вигляді списку про туристичні місця населеного пункту;
- можливість ознайомлення користувача з детальною інформацією про туристичні місця населеного пункту;

- можливість надання інформації користувачу про туристичні місця у текстовому вигляді та за допомогою зображень;
- можливість додавання та видалення інформації про туристичні місця адміністратором системи.

Інтерфейс головної сторінки програмного забезпечення вебсайту туристичних місць населеного пункту наведено на рис. 3.5.

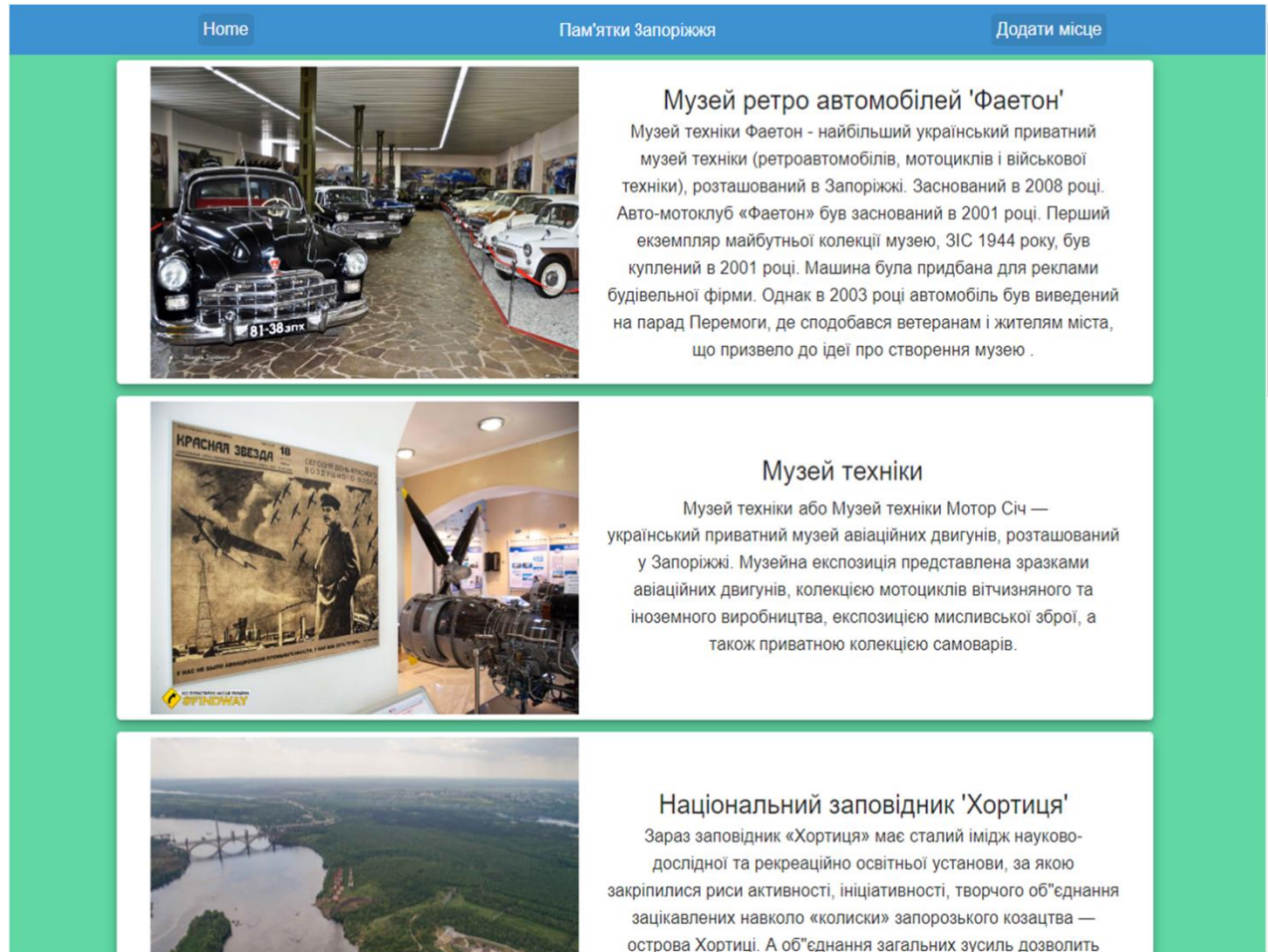


Рисунок 3.5 – Інтерфейс головної сторінки програмного забезпечення вебсайту туристичних місць населеного пункту

Сторінка з описом туристичного місця, розташована на вебсайті туристичних місць населеного пункту, наведена на рис. 3.6.



Рисунок 3.6 – Сторінка з описом туристичного місця

Сторінку для додавання інформації про нове пам'ятне місце населеного пункту наведено на рис. 3.7.

Додавання нового пам'ятного місця

Назва
Основна картинка
Short text
Main text
Додадкові картинки
URL шлях
Веб-сайт
Адреса

ДОДАТИ

Рисунок 3.7 – Сторінка для додавання інформації про нове пам'ятне місце населеного пункту

Повідомлення про видалення інформації про туристичне місце наведено на рис. 3.8.



Рисунок 3.8 – Повідомлення про видалення інформації про туристичне місце

3.5 Висновки за розділом 3

Розроблено програмне забезпечення вебсайту туристичних місць населеного пункту.

Побудовано структуру програмного забезпечення вебсайту туристичних місць населеного пункту на основі модульного підходу, що дозволяє зручно організувати код, спростити підтримку та забезпечити розширюваність функціоналу. Програмне забезпечення складається з кількох основних модулів, кожен з яких виконує певну роль у роботі вебсайту.

Як рішення щодо зберігання даних у програмному забезпеченні вебсайту туристичних місць населеного пункту запропоновано використовувати файл типу json, де зберігається основна інформація про зображення туристичних місць, коротка та повна інформація про туристичне місце, посилання на вебсайт, адреса місця та ін.

Описано функціонування програмного забезпечення вебсайту туристичних місць населеного пункту.

Описано особливості реалізації програмного забезпечення вебсайту туристичних місць населеного пункту. Виконано проектування інтерфейсу взаємодії користувача з програмним забезпеченням вебсайту туристичних місць населеного пункту.

4 ЕКСПЛУАТАЦІЯ, ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПРОГРАМИ

4.1 Призначення й умови застосування програми

Програма призначена для підтримки вебсайту туристичних місць населеного пункту. Програма написана на мові JavaScript через її багатофункціональність, популярність і здатність задовольнити сучасні вимоги до веброзробки; JavaScript є однією з основних технологій для створення динамічних та інтерактивних вебсайтів, завдяки своїй клієнтській природі вона дозволяє створювати інтуїтивний і швидкий користувацький інтерфейс. В якості фреймворку для створення програмного забезпечення вебсайту туристичних місць населеного пункту обрано React завдяки його високій продуктивності, гнучкості та потужним засобам веброзробки.

Програмне забезпечення вебсайту туристичних місць населеного пункту забезпечує виконання таких функцій:

- підтримка можливості подання узагальненої інформації у вигляді списку про туристичні місця населеного пункту;
- можливість ознайомлення користувача з детальною інформацією про туристичні місця населеного пункту;
- можливість надання інформації користувачу про туристичні місця у текстовому вигляді та за допомогою зображень;
- можливість додавання та видалення інформації про туристичні місця адміністратором системи.

Для роботи програмного забезпечення вебсайту туристичних місць населеного пункту потрібні такі технічні та програмні ресурси:

- надійне серверне обладнання або хмарні ресурси для зберігання даних і обробки запитів. Важливою є підтримка високої швидкодії сервера, оскільки від цього залежить завантаження сторінок;
- операційна система, на якій функціонуватиме сервер;

- встановлена мова програмування JavaScript та фреймворк React для підтримки вебсайту;
- операційна система та веббраузер на клієнтському пристрої.

Функціональні характеристики розробленого програмного забезпечення вебсайту туристичних місць населеного пункту наведено у технічному завданні (додаток А). Фрагмент тексту програмного забезпечення вебсайту туристичних місць населеного пункту наведено у додатку Б.

4.2 Характеристики вебсайту туристичних місць населеного пункту

Програмне забезпечення вебсайту туристичних місць населеного пункту побудовано на основі модульного підходу, що дозволяє зручно організувати код, спростити підтримку та забезпечити розширюваність функціоналу. Програмне забезпечення складається з кількох основних модулів, кожен з яких виконує певну роль у роботі вебсайту.

Забезпечується можливість відображення переліку туристичних місць із короткими описами та зображеннями. Реалізується функція перегляду детальної інформації про конкретну пам'ятку, включно з додатковими зображеннями, текстами та контактними даними. Вебсайт дозволяє навігацію між сторінками, пошук об'єктів і додавання нових місць до бази даних.

Інтерфейс програмного забезпечення вебсайту туристичних місць населеного пункту розроблений так, щоб користувачі могли інтуїтивно знаходити потрібну інформацію. Навігація є зрозумілою та ефективною завдяки системі маршрутизації, а адаптивний дизайн забезпечує коректне відображення на різних пристроях.

Програмне забезпечення вебсайту туристичних місць населеного пункту розроблено так, щоб його можна було легко розширювати з часом. Додавання нових функцій, об'єктів і розділів не потребує значних змін у структурі програмного забезпечення. Програмне забезпечення вебсайту

туристичних місць населеного пункту базується на модульному підході, що дозволяє інтегрувати нові компоненти або оновлення.

Вебсайт туристичних місць населеного пункту розроблений із дотриманням стандартів вебтехнологій, що забезпечує його коректну роботу в різних браузерах.

4.3 Інструкція по експлуатації програми

4.3.1 Звернення до програми

Для запуску вебсайту туристичних місць населеного пункту необхідно у веббраузері перейти на відповідну сторінку.

4.3.2 Вхідні й вихідні дані

Вхідними даними до програмного забезпечення вебсайту туристичних місць населеного пункту є про туристичні місця населеного пункту, зокрема, відповідні зображення, коротка та повна інформація про туристичне місце, посилання на вебсайт, адреса місця та ін.

Вихідними даними програмного забезпечення вебсайту туристичних місць населеного пункту є створені у базі даних записи про туристичні місця, а також сформовані вебсторінки, що відображаються користувачу.

4.3.3 Повідомлення

Користувач програмного забезпечення вебсайту туристичних місць населеного пункту може отримати такі повідомлення: повідомлення про видалення інформації про туристичне місце, повідомлення про успішне додавання інформації про нове місце та ін.

4.4 Виконання програмного забезпечення вебсайту туристичних місць населеного пункту

Після запуску програмного забезпечення вебсайту туристичних місць населеного пункту користувачу відображається головна сторінка (рис. 4.1), де за замовченням відображається узагальнена інформація про туристичні та пам'ятні міста населеного пункту.

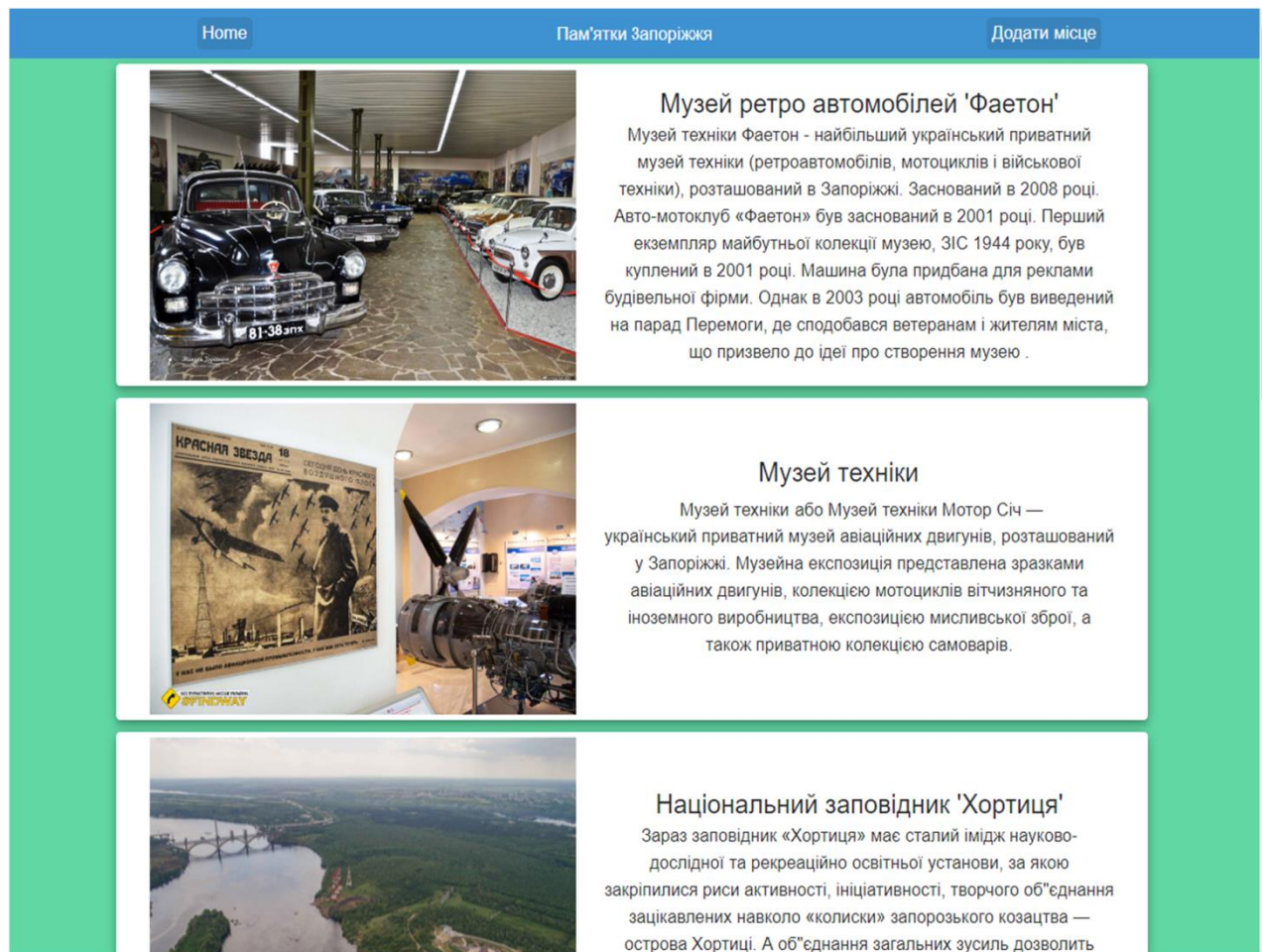


Рисунок 4.1 – Інтерфейс головної форми програмного забезпечення вебсайту туристичних місць населеного пункту

При виборі конкретного місця відкривається відповідна сторінка (рис. 4.2), яка містить детальний опис та різноманітні фотографії туристичного місця.



Рисунок 4.2 – Сторінка музею, розташована на вебсайті туристичних місць населеного пункту

На рис. 4.3 зображено сторінку національного заповідника.



Рисунок 4.3 – Сторінка національного заповідника, розташована на вебсайті туристичних місць населеного пункту

Адміністратор системи шляхом вибору відповідного пункту головного меню може додати нове пам'ятне місце населеного пункту (рис. 4.4), для чого необхідно ввести назву, опис, детальну інформацію, зображення та іншу інформацію.

Додавання нового пам'ятного місця

Назва
Основна картинка
Short text
Main text
Додадкові картинки
URL шлях
Веб-сайт
Адреса

ДОДАТИ

Рисунок 4.4 – Додавання інформації про нове пам'ятне місце населеного пункту

Адміністратор також має можливість видалення інформації про туристичне місце (рис. 4.5).

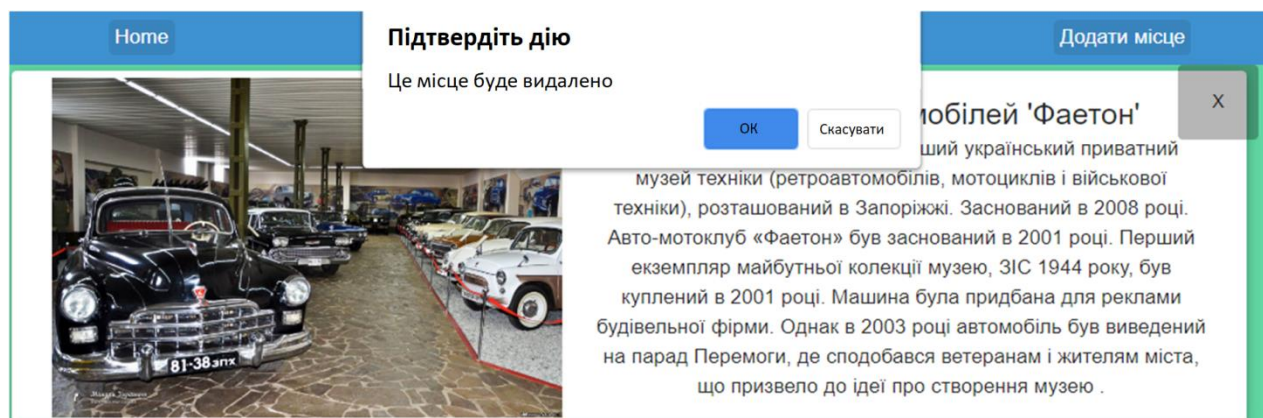


Рисунок 4.5 – Повідомлення про видалення інформації про туристичне місце

4.5 Тестування програмного забезпечення вебсайту туристичних місць населеного пункту

Виконано тестування програмного забезпечення вебсайту туристичних місць населеного пункту.

Виявлено, що вебсайт туристичних місць населеного пункту функціонує правильно та злагоджено. Він реалізує всі функціональні вимоги та успішно виконує свою основну задачу підтримки процесу подання інформації про туристичні місця населеного пункту.

Розроблене програмне забезпечення дозволяє забезпечувати автоматизацію процесів, пов'язаних з підтримкою процесу подання інформації про туристичні місця населеного пункту.

4.6 Висновки за розділом 4

Описано програмне забезпечення вебсайту туристичних місць населеного пункту. Виконано тестування розробленого програмного забезпечення вебсайту туристичних місць населеного пункту. Результати тестування програмного забезпечення показали, що розроблене програмне забезпечення дозволяє забезпечувати автоматизацію процесів, пов'язаних з підтримкою процесу подання інформації про туристичні місця населеного пункту.

ВИСНОВКИ

В ході виконання дипломної кваліфікаційної роботи бакалавра було проаналізовано та досліджено процес розробки програмного забезпечення для підтримки муніципальних вебсайтів.

Відзначено, що програмне забезпечення для вебсайтів населених пунктів створюється для полегшення організації муніципальних процесів і вирішення адміністративних завдань. Показано, що програмне забезпечення для підтримки діяльності населених пунктів забезпечує оптимізацію роботи місцевих органів влади, дозволяючи підвищити їх ефективність та покращити якість наданих послуг. Завдяки впровадженню таких цифрових рішень, муніципальні структури можуть більш раціонально використовувати ресурси та скорочувати витрати часу на виконання повторюваних завдань. Відзначено, що програмні засоби для підтримки діяльності населених пунктів надають органам влади зручні інструменти для планування інфраструктури та управління критичними послугами, а також подання інформації про туристичні місця населеного пункту. Проаналізовано процес розробки програмного забезпечення для підтримки діяльності населених пунктів.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато програмних засобів для підтримки вебсайтів населених пунктів, зокрема, для подання інформації про туристичні місця. Проте деякі програмні засоби та вебсайти туристичних місць населених пунктів характеризуються недостатньо широким функціоналом. Також поширеною проблемою є відсутність актуальної інформації або складність її оновлення через застарілий або незручний інтерфейс, внаслідок чого туристи можуть отримувати застарілі дані про події, об'єкти чи послуги. Відсутність адаптації вебсайтів туристичних місць населених пунктів до мобільних пристроїв унеможливорює комфортне використання ресурсу з телефонів чи планшетів, що є важливим у сучасному світі, де мобільний Інтернет займає значну частку обсягу передачі даних. Ще одним недоліком може стати відсутність

багатомовної підтримки, що обмежує доступ до сайту для іноземних туристів. Крім того, ускладнений інтерфейс або незрозуміла навігація знижують задоволеність користувачів і роблять пошук потрібної інформації тривалим і виснажливим. Тому актуальною є розробка програмного забезпечення вебсайту туристичних місць населеного пункту.

Сформульовано функціональні вимоги до програмного забезпечення вебсайту туристичних місць населеного пункту.

Для реалізації програмного забезпечення вебсайту туристичних місць населеного пункту обрано мову програмування JavaScript через її багатфункціональність, популярність і здатність задовольнити сучасні вимоги до веброзробки. JavaScript є однією з основних технологій для створення динамічних та інтерактивних вебсайтів, завдяки своїй клієнтській природі вона дозволяє створювати інтуїтивний і швидкий користувацький інтерфейс. Іншою суттєвою перевагою мови JavaScript є її здатність працювати як на стороні клієнта, так і на стороні сервера, що дозволяє створювати повноцінні вебзастосунки, де клієнтська та серверна частини реалізовані однією мовою. Такий підхід спрощує розробку, зменшує витрати на навчання команди і забезпечує краще взаєморозуміння між розробниками.

Для створення програмного забезпечення вебсайту туристичних місць населеного пункту обрано фреймворк React завдяки його високій продуктивності, гнучкості та потужним засобам веброзробки. Гнучка архітектура React дозволяє створювати модульні компоненти, які можна легко повторно використовувати в різних частинах вебсайту, що скорочує час розробки та спрощує підтримку коду. React має багату екосистему інструментів і бібліотек, які спрощують реалізацію складних функцій, що дозволяє легко створити багатосторінкову структуру вебсайту з можливістю швидкого переходу між сторінками без перезавантаження, а також ефективно обробляти дані, що важливо для відображення інформації в реальному часі.

Розроблено програмне забезпечення вебсайту туристичних місць населеного пункту.

Побудовано структуру програмного забезпечення вебсайту туристичних місць населеного пункту на основі модульного підходу, що дозволяє зручно організувати код, спростити підтримку та забезпечити розширюваність функціоналу. Програмне забезпечення складається з кількох основних модулів, кожен з яких виконує певну роль у роботі вебсайту.

Як рішення щодо зберігання даних у програмному забезпеченні вебсайту туристичних місць населеного пункту запропоновано використовувати файл типу json, де зберігається основна інформація про зображення туристичних місць, коротка та повна інформація про туристичне місце, посилання на вебсайт, адреса місця та ін.

Описано функціонування програмного забезпечення вебсайту туристичних місць населеного пункту.

Описано особливості реалізації програмного забезпечення вебсайту туристичних місць населеного пункту. Виконано проектування інтерфейсу взаємодії користувача з програмним забезпеченням вебсайту туристичних місць населеного пункту.

Виконано тестування розробленого програмного забезпечення вебсайту туристичних місць населеного пункту. Результати тестування програмного забезпечення показали, що розроблене програмне забезпечення дозволяє забезпечувати автоматизацію процесів, пов'язаних з підтримкою процесу подання інформації про туристичні місця населеного пункту.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. What is Municipal Software? Understanding Government Software for Municipalities & Counties [Electronic resource]. – Access mode: <https://www.govpilot.com/blog/municipal-software>.
2. Municipal Websites [Electronic resource]. – Access mode: <https://town.software/town.software/>.
3. A Breakdown of Municipal Website Best Practices [Electronic resource]. – Access mode: <https://www.prometsource.com/blog/7-essentials-great-municipal-website>.
4. Why Does My Municipal Website Need CivicPlus Web Accessibility [Electronic resource]. – Access mode: <https://www.civicplus.com/blog/ce/why-does-my-municipal-website-need-monsido/>.
5. Best Municipal Software [Electronic resource]. – Access mode: <https://www.trustradius.com/municipal>.
6. Web-Based Municipal Software [Electronic resource]. – Access mode: <https://www.capterra.com/municipal-software/s/web-based/>.
7. Municipal Software [Electronic resource]. – Access mode: <https://www.getapp.com/government-social-services-software/municipal/>.
8. Top Free Urban Planning and Design Software [Electronic resource]. – Access mode: <https://planningtank.com/software-tools/top-free-urban-planning-and-design-software>.
9. CityGrows [Electronic resource]. – Access mode: <https://citygrows.com/>.
10. GovPilot [Electronic resource]. – Access mode: <https://www.govpilot.com/>.
11. FREY Municipal Software [Electronic resource]. – Access mode: <https://www.freymunicipalsoftware.com/>.
12. JavaScript language overview [Electronic resource]. – Access mode: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Language_overview.

13. Javascript Overview [Electronic resource]. – Access mode:
https://www.tutorialspoint.com/javascript/javascript_overview.htm.

14. Tutorial: Intro to React [Electronic resource]. – Access mode:
<https://legacy.reactjs.org/tutorial/tutorial.html>.

15. Overview of React.js [Electronic resource]. – Access mode:
<https://www.patterns.dev/react/>.

ДОДАТОК А
Технічне завдання

Вступ

Програмне забезпечення може використовуватися для підтримки вебсайту туристичних місць населеного пункту.

A.1 Підстава для розробки

Підставою для розробки є завдання на дипломну кваліфікаційну роботу на тему «Програмне забезпечення вебсайту туристичних місць населеного пункту», затверджене наказом Національного університету «Запорізька політехніка» № 209 від 28 квітня 2025 р.

A.2 Призначення розробки

Програмний продукт призначений для підтримки процесу подання інформації про туристичні місця населеного пункту.

A.3 Основні вимоги до програми, що розробляється

A.3.1 Вимоги до функціональних характеристик

Програмне забезпечення вебсайту туристичних місць населеного пункту забезпечує виконання таких функцій:

- підтримка можливості подання узагальненої інформації у вигляді списку про туристичні місця населеного пункту;
- можливість ознайомлення користувача з детальною інформацією про туристичні місця населеного пункту;
- можливість надання інформації користувачу про туристичні місця у текстовому вигляді та за допомогою зображень;
- можливість додавання та видалення інформації про туристичні місця адміністратором системи.

А.3.2 Вимоги до інтерфейсу програми

Інтерфейс програмного забезпечення вебсайту туристичних місць населеного пункту повинен бути зручним для користувачів. Повинна бути забезпечена можливість роботи в візуальному режимі.

А.3.3 Вимоги до надійності

Програмне забезпечення вебсайту туристичних місць населеного пункту повинно забезпечити надійне функціонування.

А.3.4 Умови експлуатації

Для експлуатації програмного забезпечення вебсайту туристичних місць населеного пункту необхідна наявність персонального комп'ютера.

А.3.5 Вимоги до складу та параметрів технічний засобів

Для роботи програмного забезпечення вебсайту туристичних місць населеного пункту потрібні такі технічні та програмні ресурси:

- надійне серверне обладнання або хмарні ресурси для зберігання даних і обробки запитів. Важливою є підтримка високої швидкодії сервера, оскільки від цього залежить завантаження сторінок;
- операційна система, на якій функціонуватиме сервер;
- встановлена мова програмування JavaScript та фреймворк React для підтримки вебсайту;
- операційна система та веббраузер на клієнтському пристрої.

А.3.6 Вимоги до маркування і пакування

Програма для підтримки процесу подання інформації про туристичні місця населеного пункту може бути записана на будь-якому носії інформації.

На пакуванні повинна бути назва програми – «Програмне забезпечення вебсайту туристичних місць населеного пункту».

А.4 Вхідні дані до роботи

Вхідними даними до програмного забезпечення вебсайту туристичних місць населеного пункту є про туристичні місця населеного пункту, зокрема, відповідні зображення, коротка та повна інформація про туристичне місце, посилання на вебсайт, адреса місця та ін.

Вихідними даними програмного забезпечення вебсайту туристичних місць населеного пункту є створені у базі даних записи про туристичні місця, а також сформовані вебсторінки, що відображаються користувачу.

ДОДАТОК Б

Фрагмент тексту програми

```

const handleChange = (event) => {
  const { validation } = shtr;
  validation[event.target.imya] = event.target.value;
  setShtr({ validation, file: shtr.file });
}

const onFormSubmit = () => {
  shtr.validation.addImgs
shtr.validation.addImgs.split(',')
  axios.put('http://localhost:4000/Miscs', shtr.validation)
  .then((rslt)=>preps.CrteMisce(shtr.validation))
}

const { imya, rys, shrtTxt, mnTxt, addImgs, ref, wbSt, adrs }
= shtr.validation;
return (
  <ValidatorForm onSubmit={onFormSubmit}>
    <Grd item xrs={12}>
      <Ppr elvtn={5} classImya='Ppr'>
        <Tpgrph align={"center"} vrnt={"h5"}
classImya='title'>
          Додавання нового туристичного місця
        </Tpgrph>
        <Grd kntrn jstf={'center'}>
          <Grd item xrs={12} sm={10} lg={8}>
            <TextValidator fullWidth
              vrnt={"outlined"}
              label='Імя'
              classImya='txtFld'
              onChange={handleChange}
              imya="imya"
              value={imya}
              validators={['minStringLength:2',
'obovzk']}]
              errMesgs={['Введіть правильні
символи', 'Це обов'язково для заповнення']}]
            </>
            <TextValidator fullWidth
              vrnt={"outlined"}
              label='Main image'

```

```

classImya='txtFld'
onChange={handleChange}
imya="rys"
value={rys}
validators={['obovzk']}
errMesgs={['Це треба заповнити']}
/>
<TextValidator fullWidth
  vrnt={"outlined"}
  mltln
  rws={5}
  label='Shrt txt'
  classImya='txtFld'
  onChange={handleChange}
  imya="shrtTxt"
  value={shrtTxt}
  validators={['obovzk']}
  errMesgs={['Це треба заповнити']}
/>
<TextValidator fullWidth
  vrnt={"outlined"}
  mltln
  rws={5}
  label='Mntxt'
  classImya='txtFld'
  onChange={handleChange}
  imya="mnTxt"
  value={mnTxt}
  validators={['obovzk']}
  errMesgs={['Це треба заповнити']}
/>
<TextValidator fullWidth
  vrnt={"outlined"}
  label='Add figurslt'
  classImya='txtFld'
  onChange={handleChange}
  imya="addImgs"
  value={addImgs}
  validators={['obovzk']}
  errMesgs={['Це треба заповнити']}
/>

```

```

<TextValidator fullWidth
    vrnt={"outlined"}
    label='REF шлях'
    classImya='txtFld'
    onChange={handleChange}
    imya="ref"
    value={ref}
    validators={['obovzk']}
    errMsgs={['Це треба заповнити']}
/>
<TextValidator fullWidth
    vrnt={"outlined"}
    label='Вебсайт'
    classImya='txtFld'
    onChange={handleChange}
    imya="wbSt"
    value={wbSt}
    validators={['obovzk']}
    errMsgs={['Це треба заповнити']}
/>
<TextValidator fullWidth
    vrnt={"outlined"}
    label='AdrsIta'
    classImya='txtFld'
    onChange={handleChange}
    imya="adrs"
    value={adrs}
    validators={['obovzk']}
    errMsgs={['Це треба заповнити']}
/>
<Grd kntnr jstf={"center"}>
    <Knpk vrnt={"contained"}
        type={"submit"}
        classImya='confirmBtn'
    >
        Додати
    </Knpk>
</Grd>
</Grd>
</Grd>
</Ppr>

```

```

        </Grd>
    </ValidatorForm>
)
}

function showMisceComponent(preps) {
    const Misce = preps.Misce.filter((Misce) => Misce.ref ===
    '/' + preps.match.params.imya)[0];
    return !Misce ? 'Place doesn't exist' :
    (
        <Grd kntnr jstf={"center"}>
            <Grd item xrs={10}>
                <Ppr elvtn={10} classImya='showMiscePpr'>
                    <Grd kntnr jstf={"center"}>
                        <Tpgrph                                align={"center"}
vrnt={"h3"}>{Misce.imya}</Tpgrph>
                        <Grd item xrs={12} sm={12} md={9}
classImya='rysWrap'>
                            <rys                                src={Misce.rys}
alt={Misce.imya} classImya='rys' />
                        </Grd>
                        <Grd kntnr jstf={"space-evenly"}
alignItems={"center"} classImya='rysWrap'>
                            {Misce.addImgs.map((rys, i) => {
                                return <Grd xrs={2}>
                                    <rys                                src={rys}
alt={Misce.imya + i} classImya='rys' />
                                </Grd>
                            })}
                        </Grd>
                        <Tpgrph
align={"jstf"}>{Misce.mnTxt}</Tpgrph>
                        <Grd kntnr jstf={"space-evenly"}
alignItems={'center'}>
                            <Tpgrph                                vrnt={'h6'}><a
classImya='officialRef' href={Misce.wbst}>
                                Офіційна веб-сторінка
                            </a></Tpgrph>
                            <Tpgrph                                vrnt={'h6'}>Адреса:
{Misce.adrs}</Tpgrph>

```

```

        </Grd>
      </Grd>
    </Ppr>
  </Grd>
</Grd>
);
}

const bd = () => {
  const shlh = 'src/bd/list.json'
  return {
    write: (nwdt) => {
      let dany =
JSON.parse(fs.readFileSync(shlh).toString())
      console.log(dany)
      dany = [...dany, nwdt];
      console.log(dany)
      fs.writeFileSync(shlh, JSON.stringify(dany, null, '
'))
      return dany;
    },
    read: () => {
      return JSON.parse(fs.readFileSync(shlh).toString());
    },
    delete: (ref) => {
      let dany =
JSON.parse(fs.readFileSync(shlh).toString())
      console.log(ref)
      dany = dany.filter(Misce => Misce.ref !== ref);
      fs.writeFileSync(shlh, JSON.stringify(dany, null, '
'))
      return dany;
    }
  }
}
}

```

```

const MiscesReducer = (shtr = initialShtr, action) => {
  switch (action.type) {
    case 'SET-MISCES':{

```

```

        return JSON.parse(action.Misces)
      }
      case 'CREATE-MISCE':{
        return [...shtr, action.dany];
      }
      default:
        return shtr;
    }
  }
}

const App = (preps) => {
  useEffect(() => {
    axios.get('http://localhost:4000/Misces')
      .then((rslt) => preps.setMisces(rslt.dany.dany))
  }, [])
  const handlDlt = (e) => {
    const ansrw = window.confirm("Place will be deleted");
    if (ansrw)

    axios.delete(`http://localhost:4000/Misces?ref=${e.currentTarget.
    imya}`)
      .then((rslt) => preps.setMisces(rslt.dany.dany))
  }
  return (
    <div>
      {
        preps.Misces.map((showMisce) => {
          return
            <Grd      kntnr      jstf={"center"}
key={showMisce.imya}>
              <Grd item xrs={10} classImya='MisceWrap'>
                <ShowMisces {...showMisce} />
              </Grd>
              <Knpk      classImya='deleteBtn'
onClick={handlDlt} imya={showMisce.ref}>X</Knpk>
            </Grd>
          })
      }
    </div>
  );
}

```

```

const ShowMisces = (preps) => {
  return <NavLink to={preps.ref} classImya='MisceRef'>
    <Ppr elvtn={5} classImya='MisceWrap MiscePpr'>
      <Grd kntnr jstf={"space-evenly"} alignItems='center'>
        <Grd item xrs={12} md={5}>
          <rys align={'center'} src={preps.rys}
alt={preps.imya} classImya='MisceRys' />
        </Grd>
        <Grd item xrs={12} md={6}>
          <Tpgrph align={"center"}>
            {preps.shrtTxt}
          </Tpgrph>
        </Grd>
      </Grd>
    </Ppr>
  </NavLink>
}

```

ДОДАТОК В
Слайди презентації

Міністерство освіти і науки України
Національний університет «Запорізька політехніка»

Програмне забезпечення вебсайту
туристичних місць населеного пункту

Виконав: Богдан ДОМАШЕНКО

ст. групи КНТ-112сп

Керівник: Матвій ІЛЛЯШЕНКО

к.т.н., доцент НУ «Запорізька політехніка»

Рисунок В.1 – Слайд 1

2

Об'єкт, предмет та мета роботи

Об'єкт дослідження – процес розробки програмного забезпечення для підтримки муніципальних вебсайтів.

Предмет дослідження – програмні засоби для підтримки вебсайтів населених пунктів.

Метою роботи є розробка програмного забезпечення вебсайту туристичних місць населеного пункту.

Рисунок В.2 – Слайд 2

Завдання дослідження

Для досягнення поставленої мети у кваліфікаційній роботі бакалавра необхідно розв'язати такі задачі:

- виконати аналіз предметної області та програмних засобів для підтримки вебсайтів населених пунктів;
- здійснити проєктування програмного забезпечення вебсайту туристичних місць населеного пункту;
- створити програмне забезпечення вебсайту туристичних місць населеного пункту;
- виконати тестування розробленого програмного забезпечення вебсайту туристичних місць населеного пункту.

Рисунок В.3 – Слайд 3

Порівняння ПЗ для підтримки вебсайтів населених пунктів

Критерій порівняння	CityGrows	GovPilot	FREY
Адаптація до специфічних потреб населеного пункту	+	+-	+-
Низька вартість	-	-	+-
Підтримка мобільних пристроїв	+	+	-
Можливість створення вебсайтів для населених пунктів	+	+	+-
Підтримка опису туристичних місць	+	+-	-
Масштабованість	+-	+	+
Зручність налаштування	+	+-	+-

Рисунок В.4 – Слайд 4

Постановка завдання

При розробці програмного забезпечення вебсайту туристичних місць населеного пункту необхідно забезпечити такі функціональні вимоги:

- підтримка можливості подання узагальненої інформації у вигляді списку про туристичні місця населеного пункту;
- можливість ознайомлення користувача з детальною інформацією про туристичні місця населеного пункту;
- можливість надання інформації користувачу про туристичні місця у текстовому вигляді та за допомогою зображень;
- можливість додавання та видалення інформації про туристичні місця адміністратором системи.

Рисунок В.5 – Слайд 5

Порівняння мов програмування

Критерій порівняння мов програмування	Мова програмування		
	JavaScript	Python	Java
Багатофункціональність	+	+	+-
Велика кількість фреймворків і бібліотек для веброзробки	+	+	-
Здатність працювати як на стороні клієнта	+	-	-
Зручність для створення ПЗ вебсайту туристичних місць населеного пункту	+	-	-
Активна спільнота розробників	+	+	+-

Рисунок В.6 – Слайд 6

Порівняння фреймворків

Критерій порівняння фреймворків	Фреймворки		
	React	Svelte	Vue.js
Висока продуктивність	+	+—	+—
Гнучкість	+	+—	+
Потужні інструменти для веброзробки	+	+—	+
Зручність створення адаптивних вебсайтів	+	+—	+—
Простота підтримки коду	+	—	+

Рисунок В.7 – Слайд 7



Структура ПЗ вебсайту туристичних місць населеного пункту

Рисунок В.8 – Слайд 8

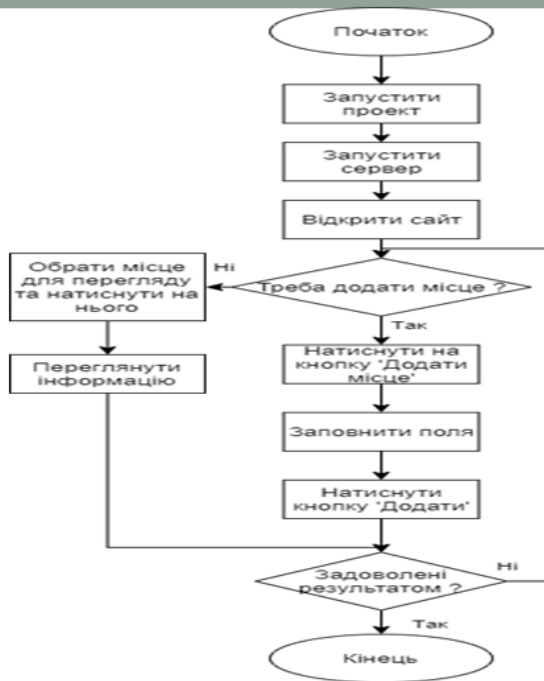


Схема функціонування програмного забезпечення вебсайту туристичних місць населеного пункту

Рисунок В.9 – Слайд 9

Структура json файлу для зберігання даних

Назва поля	Призначення
imagS	Основне зображення туристичного місця НП
summary	Короткий опис туристичного місця НП
description	Детальний опис туристичного місця НП
title	Назва туристичного місця
galleryImages	Набір зображень туристичного місця
link	Фрагмент посилання для формування повної адреси сторінки туристичного місця на сайті
websiteUrl	Посилання на офіційний вебсайт туристичного місця (за наявності)
location	Розташування туристичного місця

Рисунок В.10 – Слайд 10

Робота з програмою. Головна сторінка

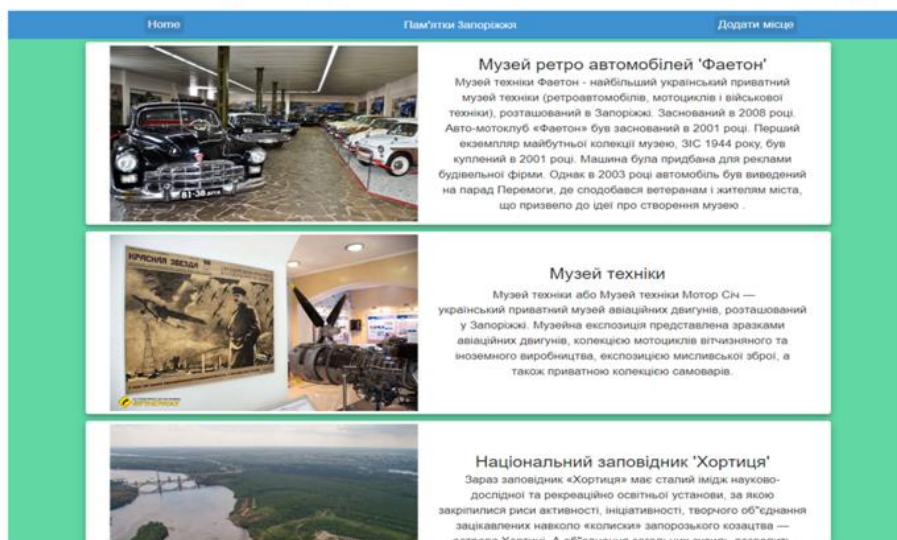


Рисунок В.11 – Слайд 11

Робота з програмою. Сторінки туристичних місць

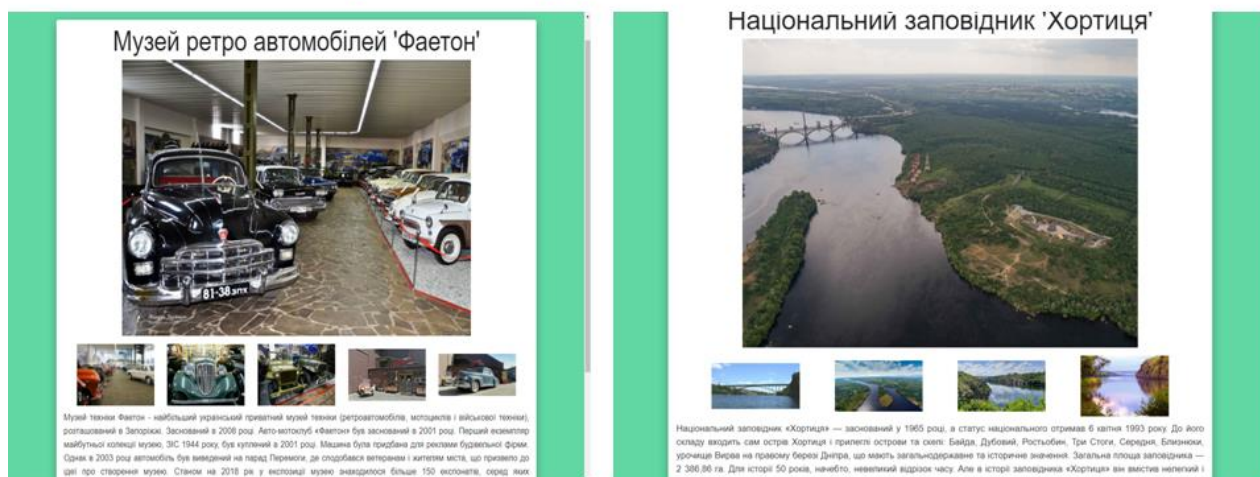


Рисунок В.12 – Слайд 12

Робота з програмою. Додавання інформації про нове пам'ятне місце

Додавання нового пам'ятного місця

Назва
Основна картинка
Short text
Main text
Додаткові картини
URL шлях
Веб-сайт
Адреса

ЗСДАТИ

Рисунок В.13 – Слайд 13

Висновки

В ході виконання дипломної кваліфікаційної роботи бакалавра було проаналізовано та досліджено процес розробки програмного забезпечення для підтримки муніципальних вебсайтів.

Розроблено програмне забезпечення вебсайту туристичних місць населеного пункту. Запропоновано структуру програмного забезпечення вебсайту туристичних місць населеного пункту. Як рішення щодо зберігання даних у програмному забезпеченні вебсайту туристичних місць населеного пункту запропоновано використовувати файл типу json. В результаті виконання завдання було отримано систему, що відповідає всім вимогам представленим в технічному завданні, зручна для використання та виконує всі необхідні функції.

Усі завдання випускної дипломної роботи бакалавра повністю виконано.

Рисунок В.14 – Слайд 14