

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК
СЕКЦІЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему: «Web-додаток підтримки вибору туристичного маршруту
пам'ятками Сумщини»

за спеціальністю 122 «Комп'ютерні науки»,
освітньо-професійна програма «Інформаційні технології проектування»

Виконавець роботи: студентка групи ІТ-61-8 Головенко Анастасія Олегівна

**Кваліфікаційна робота бакалавра
захищена на засіданні ЕК
з оцінкою**

_____ «__» _____ 2020 р.

Науковий керівник

(підпис)

к.т.н., ст.викл., Кузнецов Е.Г.

(науковий ступінь, вчене звання, прізвище та ініціали)

Голова комісії

(підпис)

Шифрін Д. М.

(науковий ступінь, вчене звання, прізвище та ініціали)

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів
без відповідних посилань.

Студент _____

(підпис)

Суми-2020

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук
Секція інформаційних технологій проектування
Спеціальність 122 «Комп'ютерні науки»
Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ
Зав. секцією ІТП

_____ В. В. Шендрик
«__» _____ 2020 р.

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ

Головенко Анастасія Олегівна

1 Тема роботи Web-додаток підтримки вибору туристичного маршруту пам'ятками Сумщини

керівник роботи Кузнєцов Едуард Геннадійович, к.т.н., старший викладач,

затверджені наказом по університету від « 14 » травня 2020 р. № 0576-III

2 Строк подання студентом роботи «1» червня 2020 р.

3 Вхідні дані до роботи відомості про культурні пам'ятки Сумщини

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) аналіз предметної області, моделювання та проектування додатку, розробка Web-додатку

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) актуальність, мета роботи, задачі, аналіз аналогів, порівняння розроблюваного додатку та аналогів, функціональні вимоги, моделювання процесу підтримки вибору туристичного маршруту, декомпозиція процесу підтримки вибору туристичного маршруту, діаграма варіантів використання, архітектура додатку, засоби реалізації, отримання даних зі стороннього ресурсу, модель даних у вигляді ER-діаграми, демонстрація роботи, висновки, оприлюднення роботи.

РЕФЕРАТ

Тема кваліфікаційної роботи бакалавра: «Web-додаток підтримки вибору туристичного маршруту пам'ятка Сумщини».

Пояснювальна записка складається зі вступу, 3 розділів, висновків, списку використаних джерел із 23 найменувань, 7 додатків. Загальний обсяг роботи – 76 сторінок, у тому числі 50 сторінок основного тексту, 2 сторінки списку використаних джерел, 23 сторінки додатків.

Кваліфікаційну роботу бакалавра присвячено розробці Web-додатку підтримки вибору туристичного маршруту пам'ятками Сумщини.

В першому розділі була проаналізована предметна область та визначена мета, об'єкт та предмет роботи. Огляд існуючих програмних продуктів дозволив визначити переваги та недоліки для їх врахування при розробці додатку. Були визначені задачі для виконання роботи, а також засоби реалізації, за допомогою яких буде проведена їх реалізація.

Другий розділ присвячено виконанню структурно-функціонального моделювання, опису варіантів використання Web-додатку та проектуванню моделі бази даних.

У третьому розділі описується процес розробки додатку, наводиться його архітектура та наводяться зображення, які демонструють використання Web-додатку.

Результатом виконання роботи є розроблений Web-додаток підтримки вибору туристичного маршруту пам'ятками Сумщини.

Ключові слова: Web-додаток, збір даних, туристична пам'ятка, мапа, маршрут.

ЗМІСТ

ВСТУП.....	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	7
1.1 Огляд останніх досліджень і публікацій.....	7
1.2 Аналіз програмних продуктів – аналогів.....	8
1.3 Постановка задачі.....	13
1.4 Вибір засобів реалізації.....	14
2 ПРОЕКТУВАННЯ WEB-ДОДАТКУ ПІДТРИМКИ ВИБОРУ ТУРИСТИЧНОГО МАРШРУТУ ПАМ'ЯТКАМИ СУМЩИНИ.....	17
2.1 Структурно-функціональне моделювання.....	17
2.2 Моделювання варіантів використання web-додатку підтримки вибору туристичного маршруту пам'ятками Сумщини.....	19
2.3 Проектування моделі бази даних.....	21
3 РОЗРОБКА WEB-ДОДАТКУ ПІДТРИМКИ ВИБОРУ ТУРИСТИЧНОГО МАРШРУТУ ПАМ'ЯТКАМИ СУМЩИНИ.....	29
3.1 Архітектура Web-додатку.....	29
3.2 Реалізація Web-додатку.....	30
3.3 Використання Web-додатку.....	35
ВИСНОВКИ.....	49
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	51
ДОДАТОК А ТЕХНІЧНЕ ЗАВДАННЯ.....	53
ДОДАТОК Б ПЛАНУВАННЯ РОБІТ.....	59
ДОДАТОК В ЛІСТИНГ ЗБЕРЕЖЕННЯ ПАМ'ЯТОК ДО БАЗИ ДАНИХ.....	71
ДОДАТОК Г ЛІСТИНГ ШАБЛОНУ СТОРІНКИ ДЕТАЛЬНОЇ ІНФОРМАЦІЇ ПРО ПАМ'ЯТКУ.....	72
ДОДАТОК Д ЛІСТИНГ ВІДОБРАЖЕННЯ СТВОРЕННЯ МАРШРУТУ.....	74
ДОДАТОК Е ЛІСТИНГ ВІДОБРАЖЕННЯ ЗБЕРЕЖЕНОГО МАРШРУТУ... ..	75
ДОДАТОК Ж ОПРИЛЮДНЕННЯ РЕЗУЛЬТАТІВ РОБОТИ.....	76

ВСТУП

Інформаційні технології міцно закріпились у житті людини та щодня його полегшують. Завдяки їм не виходячи з дому можна навчатись, працювати, отримувати різноманітні консультації спеціалістів тощо. Це значно економить час, тому є потреба у тому, щоб безліч послуг були доступна саме через електронні засоби, зокрема, туристичні сервіси.

Кожного дня люди відправляються у подорожі заради нових вражень, дослідництва або по роботі, тому необхідно мати уявлення куди планується подорож. Туристичні агентства організовують подорожі й за рахунок їх роботи можлива економія часу. Але вони не можуть коригувати ці маршрути, тому зараз набувають популярності індивідуальні подорожі. Такі тури складені безпосередньо за індивідуальними побажаннями, це маршрути тими місцями, відвідати які цікаво для кожного. Індивідуальний туризм передбачає самостійний вибір міста, готелю, транспорту чи закладу харчування. Подорож буде сплановано самостійно за обраним маршрутом і в зручні терміни, поїздка буде включати цікаві екскурсії для кожного. Тому метою роботи є розробка Web-додатку підтримки вибору туристичного маршруту пам'ятками Сумщини, який дасть змогу користувачеві не лише отримувати дані про місто чи визначні пам'ятки, а й створювати власний маршрут із цікавих для нього місць. Для досягнення мети необхідно вирішити наступні задачі:

- проаналізувати предметну область та визначити актуальність роботи;
- провести огляд продуктів-аналогів;
- визначити вимоги до додатку й виконати планування робіт;
- розробити модель додатку створення туристичного маршруту;
- обрати засоби реалізації та середовище розробки;
- реалізувати Web-додаток з врахуванням можливостей засобів реалізації й вимог до функціоналу та провести її тестування.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх досліджень і публікацій

За останні роки туристична галузь в Україні почала розвиватись і стала масовим соціально-економічним явищем. Її швидкому розвитку сприяє розширення політичних, економічних, наукових і культурних зв'язків між Україною та іншими державами.

Розвиток туризму в Україні суттєво впливає на транспорт, торгівлю, зв'язок, будівництво, сільське господарство, виробництво товарів народного споживання та є одним із найбільш перспективних напрямків структурної перебудови економіки. Використання у повному обсязі туристичного потенціалу дасть можливість нашій країні стати однією з провідних країн світу, що матиме позитивний вплив на економічний і соціальний стан країни в цілому, стимулюватиме ряд важливих галузей економіки, сприятиме зміцненню нового позитивного іміджу України на світовій арені [1].

Вирішення завдання створення туристичного маршруту є неможливим без використання інформаційних технологій. Складність побудови туристичного маршруту полягає у наданні користувачеві можливості його прокладання цікавими туристичними пам'ятками. Такого роду завдання можна класифікувати як задачу комбінаторної оптимізації, рішенням якої буде задача комівояжера в її незамкненому варіанті [23]. Вирішення задач оптимізації виконують за допомогою різноманітних алгоритмів, але вони мають наступні недоліки:

- всі алгоритми часто мають обмеження локальних рішень;
- у якості вихідного використовується лише один варіант рішення;
- кожен метод є досить чутливим до вибору умов.

1.2 Аналіз програмних продуктів – аналогів

Щоб спланувати туристичну поїздку на пошук відповідного житла, недорогих авіаквитків, продумування логістики тощо потрібно витратити чимало сил та часу. Сервіси з організації поїздок взяли на себе цю клопітку частину роботи. Сьогодні послуги з планування самостійних поїздок надають кілька компаній, серед яких – Free Travel [2], TripAdvisor [3], а також численні невеликі сервіси та приватні особи.

Сервіс Free Travel дозволяє планувати та організувати індивідуальні подорожі. На сторінках сайту можна знайти довідкову інформацію для мандрівника про транспорт, місця проживання чи визначні пам'ятки. На головній сторінці (рис. 1.1) розміщені посилання щодо оформлення закордонного паспорта, страховки мандрівника тощо.

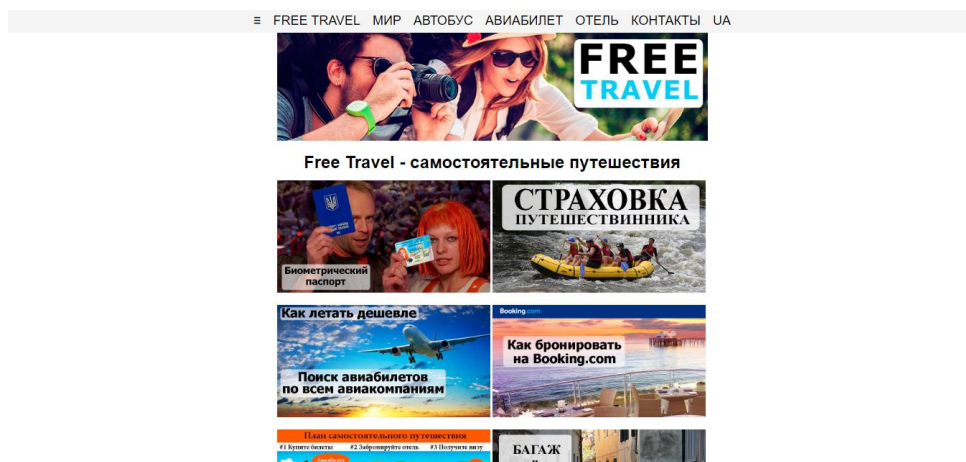


Рисунок 1.1 – Головна сторінка сервісу Free Travel

Сторінки сервісу мають довідниковий характер, оскільки про кожну країну чи місто є інформація про грошову одиницю, часовий пояс тощо (рис. 1.2).

Ще одним туристичним сервісом є американський сайт подорожей TripAdvisor. Він допоможе продумати основні нюанси поїздки, подивитися перелік визначних пам'яток й забронювати готельний номер чи столик у закладі громадського харчування. Послуги на сайті є безкоштовними для користувачів завдяки створенню ними більшої частини контенту. Саме тому сервіс можна вважати міжнародною базою експертних і користувальницьких відгуків. Відображення головної сторінки сайту представлено на рис. 1.4.

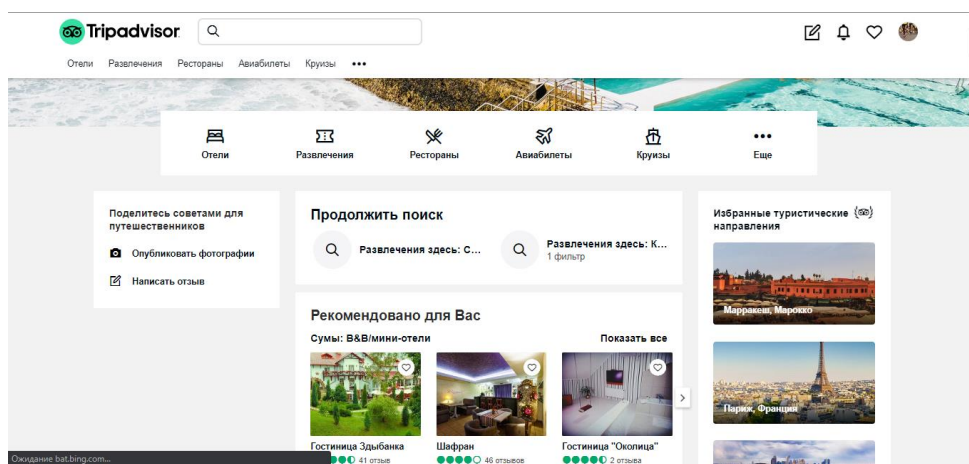


Рисунок 1.4 – Головна сторінка сервісу TripAdvisor

В додатку розміщена актуальна інформація, оскільки сервіс співпрацює з туристичними центрами, готелями й ресторанами. Саме тому на сторінках сайту можна бронювати та купувати квитки на літаки (рис. 1.5), поїзди, автобуси за низькими цінами, бронювати номери в готелях і столики в місцях громадського харчування.

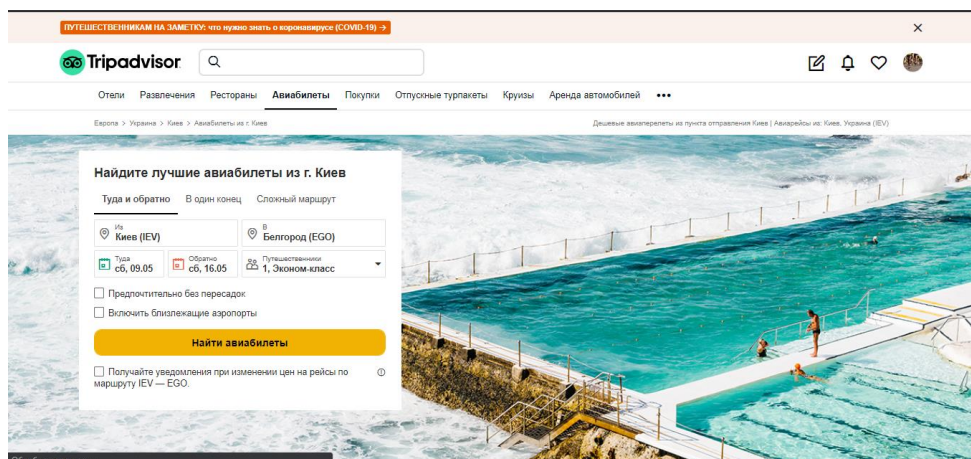


Рисунок 1.5 – Сторінка пошуку авіаквитків сервісу TripAdvisor

Сервіс дозволяє виконувати пошук культурних пам'яток та місць відпочинку з їх описами та фотографіями, а також переглядати величезну кількість відгуків про заклади та місця відпочинку (рис. 1.6).

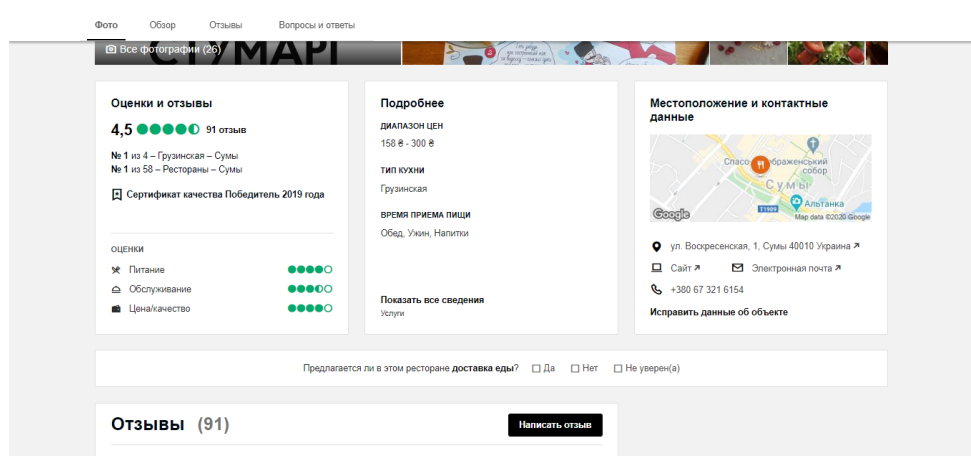


Рисунок 1.6 – Типова сторінка закладу сервісу TripAdvisor

На сторінках даних сервісів розміщена інформація про визначні місця з різних частин світу. Така велика кількість інформації може бути як і корисною та цікавою, так і заплутати користувача. Сервіс TripAdvisor, на відміну від Free Travel, передбачає повну характеристику закладу чи пам'ятки, що, безумовно, буде корисним для користувачів. Для зручності та розуміння місцезнаходження того чи іншого визначного місця доцільно реалізувати його відображення на мапі, що присутнє в обох додатках.

Обрані додатки позиціонують себе як туристичні сервіси, але кожен має свої недоліки. Сервіс Free Travel важко назвати планувальником маршруту, оскільки на його сторінках присутня інформація рекомендаційного характеру, що не дає повної картини для організації подорожі. На сторінках сайту TripAdvisor можна ознайомитися з туристичними пам'ятками, переглянути до них відгуки реальних користувачів, а також додати деякі місця до обраних для майбутньої подорожі, що дозволяє попередньо продумати основні деталі поїздки. Головним недоліком цих додатків є те, що з їх допомогою немає можливості розробити туристичний маршрут. Підтримка вибору туристичного маршруту дозволила б користувачам досить швидко зрозуміти траєкторію свого руху, спланувати час та можливості.

Для визначення переваг і недоліків кожного додатку, а також для розуміння актуальності розробки слід представити порівняння характеристик розроблюваного Web-додатку з аналогами – таблиця 1.1.

Таблиця 1.1 – Порівняння характеристик розроблюваного Web-додатку з його аналогами

Порівняльна характеристика	Розроблюваний Web-додаток	Сервіс Free Travel	Сервіс TripAdvisor
Інформація про пам'ятки	Місцевого рівня	Глобального рівня	Глобального рівня
Детальна інформація про місця відпочинку	+	-	+
Перегляд місцерозташування пам'яток на мапі через інтерфейс сайту	+	+	+
Наявність особистого кабінету	+	-	+
Створення та збереження маршруту	+	-	-

1.3 Постановка задачі

Робота полягає у розробці Web-додатку підтримки вибору туристичного маршруту пам'ятками Сумщини. Об'єктом роботи є процес надання послуг у створенні туристичного маршруту пам'ятками Сумщини. Предметом дослідження є інтернет-технології прокладання маршруту використовуючи електронні мапи.

Для досягнення поставленої мети спершу необхідно проаналізувати предметну область, щоб виділити її сутності, визначити початкові вимоги до системи та визначити межі проекту. Аналіз існуючих програмних продуктів дозволяє виділити їх позитивні сторони для подальшого впровадження в додаток, а також уникнути помилок та недоробок, які присутні в аналогах. Потім слід визначити та встановити вимоги до додатку, яких треба досягти під час розробки. Моделювання додатку дозволяє візуалізувати систему в її поточному або бажаному стані, визначити структуру та поведінку системи тощо. Вибір засобів реалізації та середовища розробки є однією з важливих задач, оскільки вибір оптимальних засобів дозволить полегшити процес виконання поставленого завдання.

Створений Web-додаток повинен надавати змогу користувачеві не лише отримувати інформацію про місто та його пам'ятки, а й створювати власний маршрут із цікавих для нього місць. Інформація для додатку буде збиратись з інших Інтернет-джерел та розподіляться на категорії для зручного перегляду. Додаток повинен бути інтуїтивно зрозумілим, адаптивним та кросбраузерним, щоб кожна людина, маючи доступ до мережі Інтернет, мала можливість ним користуватись без перешкод.

Для того, щоб мандрівник мав змогу скласти власний маршрут подорожі містом необхідно розробити відповідний модуль. Для зручності користування додатком кожен матиме змогу за бажанням зберігати складений маршрут.

Окрім основних функцій корисно буде додати можливість реєстрування користувачів, де буде зберігатись інформація про бажані або вже створені маршрути.

Отже, ціль виконання кваліфікаційної роботи бакалавра полягає у розробці Web-додатку підтримки вибору туристичного маршруту пам'ятками Сумщини. Для досягнення визначеної цілі необхідно виконати такі задачі:

- створити модуль автоматичного збору інформації;
- створити модуль прокладання маршруту;
- створити модуль збереження створеного маршруту.

Технічне завдання на розробку продукту наведено у додатку А.

1.4 Вибір засобів реалізації

Оскільки для додатку не потрібна велика кількість додаткових модулів, які створюють додаткове навантаження на сервер, було обране самописне рішення створення засобами HTML (HyperText Markup Language), CSS (Cascading Style Sheets), PHP (Hypertext Preprocessor), та JavaScript.

При розробці, використовуючи HTML і CSS необхідно розуміти те, що HTML-код задає розміщення та порядок абзаців, графічних зображень, рядків і осередків в таблиці. Каскадні таблиці стилів задають уявлення сторінки – шрифти, колір тексту, границі у таблицях тощо.

Для оформлення сторінок було обрано використання безкоштовного фреймворку Bootstrap 3, який часто використовується для верстки адаптивних дизайнів сайтів та веб-додатків. Його використання дозволяє скоротити час розробки сторінки. Фреймворк містить класи стилів тексту, зображень, таблиць та іншого контенту та їх вирівнювань, а також компоненти для створення кнопок, навігаційного меню, різноманітних форм тощо. Використані компоненти фреймворку можна налаштувати та змінювати під власні потреби, що дозволяє створювати унікальні рішення стосовно дизайну сторінок [6][8].

При розробці Web-сайтів використовують мови програмування двох великих груп. До першої належать ті, код яких виконується у браузері на комп'ютері

користувача сайту. Це JavaScript, VBScript (Visual Basic Script). Програми, написані цими мовами, вбудовують у код Web-сторінок або виносять в окремий файл з викликом з Web-сторінки. До другої групи потрапляють ті мови, за допомогою яких розроблені програми виконуються на тому комп'ютері, де розташований Web-сервер. Ця група набагато ширше, до неї відноситься PHP, яка відрізняється своєю простотою, оскільки не треба піклуватися про права доступу до файлів сценарію або прописувати точні шляхи до різних модулів. Для тестування PHP-програм на локальному комп'ютері достатньо завантажити відповідний Web-сервер і підключити до нього модуль PHP.

Для розробки функціоналу було обране рішення скористатися фреймворком – деяким набором бібліотек, що означає набір файлів, в яких зберігаються підпрограми, об'єкти, функції тощо. Використання фреймворків дозволяє прискорити розробку та підвищити її зручність. Спілкування з фреймворком відбувається через програмний інтерфейс, який містить безліч функцій та конструкцій. Сайти, розроблені за допомогою фреймворків, мають розширений функціонал та дозволяють встановлювати різні модулі, блоки, пакети тощо [4].

Одним з кращих PHP-фреймворків для розробки можна вважати Laravel. Цьому свідчить аналіз зростання зацікавленості PHP-розробників в Laravel в порівнянні з такими фреймворками як Symfony, Codeigniter, yii, zend тощо. Laravel є безкоштовним PHP-фреймворком загального призначення з відкритим кодом [5].

Функціонал Laravel є простим для розуміння та використання. Більшість його функцій, спираючись на загальноприйняті стандарти написання коду, роблять його інтуїтивно зрозумілим, працюють не вимагаючи додаткових налаштувань. Фреймворк має закінчену та документацію, яка постійно оновлюється [21].

За допомогою Laravel є можливість створити різномасштабні проекти, він дає змогу вирішити різного роду завдання, а також тестувати, оновлювати чи вносити зміни в роботу сайту. Фреймворк має просту та інтуїтивно зрозумілу адміністративну панель та надійний захист бази даних.

Інформацію про культурні пам'ятки Сумщини зручніше отримати за допомогою парсеру. Він виконує функцію завантаження контенту деяких сторінок та

вилучення з нього даних [9]. Для роботи був використаний PHP-парсер hQuery з підтримкою PHP 5.3 і вище, який швидко аналізує сторінку та здійснює пошук. Він має jQuery подібний стиль обходу об'єктної моделі документа. Однією з важливих його переваг є низьке використання пам'яті, а також можливість обробляти великі HTML-документи [18].

Для відображення місця знаходження туристичної пам'ятки та для відображення створення маршруту було обрано використання безкоштовного картографічного сервісу API Google Maps. Сервіс представлений у вигляді набору протоколів [10]. Для повноцінної роботи з ним необхідно отримати власний ключ, з яким компанія Google може відстежувати додатки, а у разі потреби – зв'язатися з власником додатку [7].

Для користування сервісом користувачеві потрібно задати своє місце розташування та прописати дані, які потрібно відобразити на карті. На заданому користувачем місці з'явиться позначення, при його натисканні буде відображена вся актуальна інформація стосовно об'єкту. Однією з переваг API Google Maps є те, що його можна оформити під загальну тематику сайту або розширити існуючі можливості за потребою.

2 ПРОЕКТУВАННЯ WEB-ДОДАТКУ ПІДТРИМКИ ВИБОРУ ТУРИСТИЧНОГО МАРШРУТУ ПАМ'ЯТКАМИ СУМЩИНИ

2.1 Структурно-функціональне моделювання

Процес моделювання додатку під час виконання роботи дозволяє: виконати візуалізацію систему в її поточному чи бажаному стані; визначити структуру та поведінку системи; отримати шаблон, що дозволяє сконструювати систему у майбутньому; документувати прийняті рішення, використовуючи отримані моделі [11].

Методологію графічного опису систем і процесів діяльності організації як безлічі взаємозалежних функцій називають IDEF0 [12]. З її допомогою можна досліджувати функції організації не пов'язуючи їх з об'єктами, які забезпечують їх реалізацію. Моделювання Web-додатку підтримки вибору туристичного маршруту пам'ятками Сумщини в IDEF0 представлено на рис. 2.1.

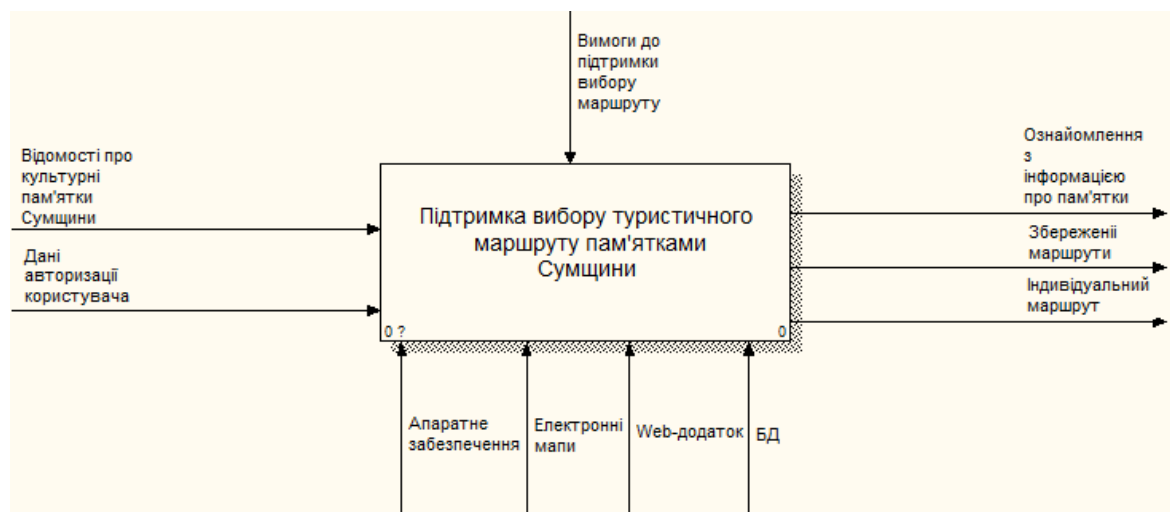


Рисунок 2.1 – IDEF0 Web-додатку

Кожна IDEF0-діаграма містить у собі блоки та дуги. Функції системи, що моделюється, зображують у блоках. Дуги пов'язують блоки разом і дозволяють відображати взаємодію та взаємозв'язки для між ними [13].

Відомості про культурні пам'ятки Сумщини використовуються і перетворюються роботою для отримання результату. Дані авторизації користувача необхідні для отримання повноцінного функціоналу додатку. Стрілка управління містить вимоги до підтримки вибору маршруту, які керують діями роботи. За допомогою використання додатку, апаратного та програмного забезпечення, а також бази даних та електронних карт виконується підтримка вибору маршруту. У результаті можна отримати інформацію про визначні пам'ятки, індивідуальний туристичний маршрут, а також збережені раніше створені маршрути.

Після опису системи виконується функціональна декомпозиція, що означає розбив системи на підсистеми [14]. Декомпозиція процесу підтримки вибору туристичного маршруту пам'ятками Сумщини представлена на рис. 2.2.

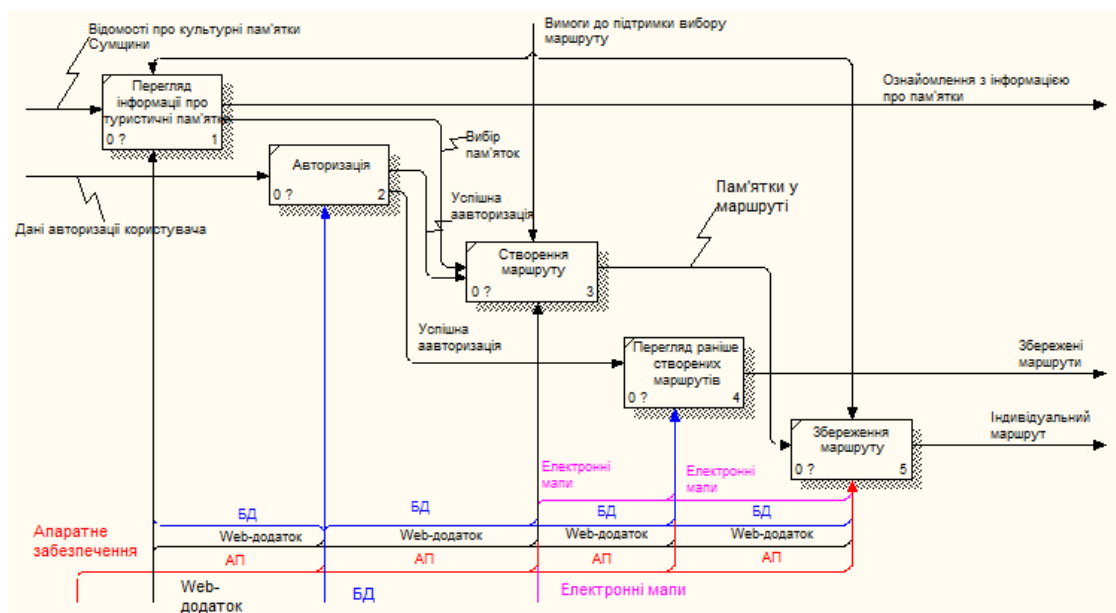


Рисунок 2.2 – Декомпозиція IDEF0-діаграми Web-додатку

Ця діаграма містить наступні блоки:

- перегляд інформації про туристичні пам'ятки;
- авторизація;

- створення маршруту;
- перегляд раніше створених маршрутів;
- збереження маршруту.

Для кожної діяльності необхідна наявність Web-додатку, апаратного забезпечення, бази даних. Відомості про культурні пам'ятки є вхідними даними до діяльності «перегляд інформації про туристичні пам'ятки». Для «авторизації» використовуються дані авторизації користувача. Для «створення маршруту» необхідний вхід до особистого кабінету та вибір туристичних пам'яток. «Перегляд раніше створених маршрутів» передбачає вхід до особистого кабінету. Набір вибраних туристичних пам'яток після створення маршруту використовуються для «збереження».

2.2 Моделювання варіантів використання web-додатку підтримки вибору туристичного маршруту пам'ятками Сумщини

Для об'єктного моделювання, для моделювання бізнес-процесів, системного проектування та відображення організаційних структур використовується мова графічного опису UML (Unified Modeling Language). Вона є відкритим стандартом, графічні позначення використовують для створення абстрактної моделі системи, званої UML-моделлю. Її мета – визначати, візуалізувати, проектувати та документувати програмні системи.

Моделювання UML має уявлення процесу спуску від загальної та абстрактної концептуальної моделі вихідної системи від логічної до фізичної моделі відповідної системи. Щоб досягти ці цілі спершу потрібно збудувати модель у формі діаграми варіантів використання (Use Case Diagram), яка описує функціональне призначення системи чи що система робитиме в процесі свого функціонування. Діаграму варіантів використання вважають вихідним концептуальним поданням або концептуальною моделлю системи в процесі її проектування і розробки.

Діаграма варіантів використання складається з акторів, елементів зв'язку з системою зовні, а також самих варіантів використання та їх зв'язків між іншими частинами діаграми.

Діаграма варіантів використання Web-додатку в UML представлена на рисунку 2.3.

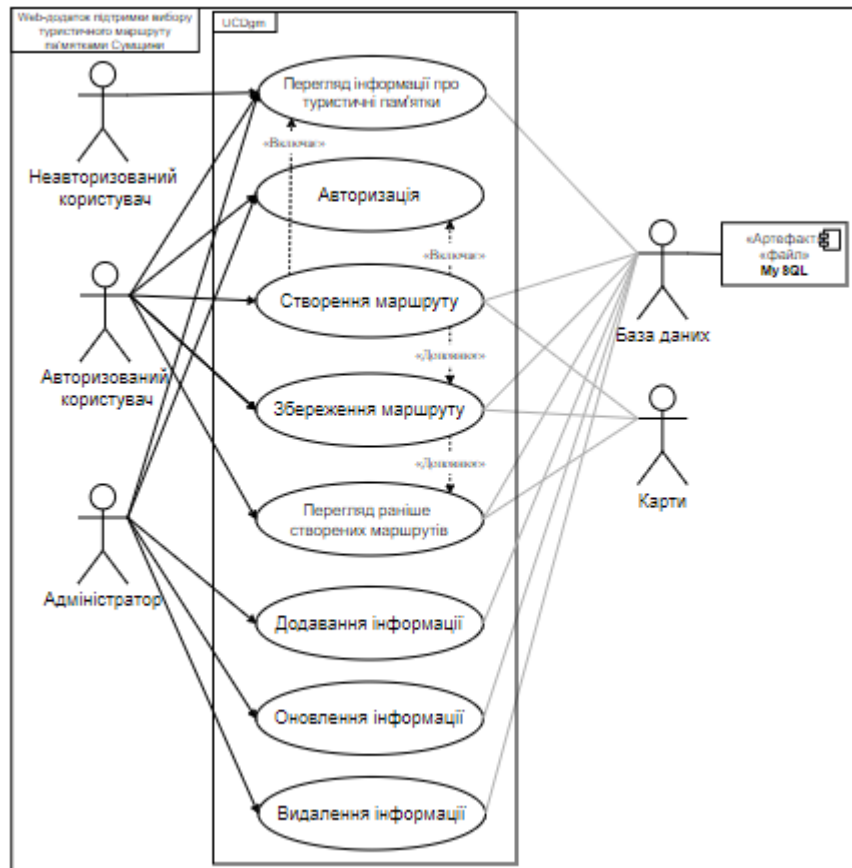


Рисунок 2.3 – Діаграма варіантів використання Web-додатку

База даних та електронні карти є елементами зовнішнього зв'язку в діаграмі. База даних містить інформацію про визначні пам'ятки, дані користувачів, а також створені та збережені маршрути. За допомогою карт відбувається створення та збереження маршруту чи його перегляд.

Варіант використання перегляд інформації про туристичні пам'ятки дозволяє користувачеві ознайомитися з інформацією про визначні пам'ятки.

Варіант використання авторизація дозволяє користувачеві працювати з додатком в залежності від наданих прав доступу.

Варіант використання створення маршруту передбачає створення індивідуального туристичного маршруту шляхом вибору деяких культурних пам'яток з переліку.

Варіант використання перегляд раніше створених маршрутів дає можливість авторизованому користувачу за наявності переглянути раніше створені ним маршрути.

Варіант використання збереження маршруту означає збереження створеного маршруту.

Варіант використання додавання інформації означає додавання даних про туристичні пам'ятки.

Варіант використання оновлення інформації означає оновлення даних про туристичні пам'ятки.

Варіант використання видалення інформації означає видалення існуючих даних про туристичні пам'ятки або даних про користувачів.

Механізми розширення UML призначені для того, щоб розробники мали можливість адаптувати мову моделювання до своїх конкретних потреб не змінюючи при цьому його мета модель. Наявність механізмів розширення принципово відрізняє UML від IDEF0, IDEF1X, IDEF3, DFD тощо, адже їх можна визначити як сильно типізовані, так як вони не допускають довільну інтерпретацію семантики елементів моделей. UML є слабо універсальною мовою через те, що допускає таку інтерпретацію [15].

2.3 Проектування моделі бази даних

Одним з важливих компонентів будь якого додатку чи інформаційної системи є база даних – структурований та організований набір даних, який описує характеристики будь-якої віртуальної або фізичної системи. За допомогою

використання бази даних можна виконувати експлуатацію додатку, виконувати його підтримку, модифікувати тощо.

Сукупність структур даних та операцій їх обробки називають моделлю даних. Вона виявляє сутності, атрибути та зв'язки з іншими даними, а також надає концептуальне представлення даних і зв'язку між ними. Моделі даних створюють для того, щоб забезпечити розробника концептуальною схемою бази даних. При їх створенні задіяний метод семантичного моделювання, інструментом якого слугує використання діаграм сутність-зв'язок. Під час їх створення використовують графічне зображення сутностей предметної області, їх властивостей (атрибутів), і взаємозв'язків між сутностями [16].

Проектування бази даних дозволяє зрозуміти логіку роботи додатку та за допомогою яких дій може бути досягнутий результат. Створена модель дає можливість концептуально визначити набори даних, які використовуються у додатку.

Web-додаток служить для підтримки вибору туристичного маршруту. Він містить інформацію про пам'ятки міста Суми. За допомогою додатку користувачі можуть отримувати відомості про культурні пам'ятки Сумщини, створювати туристичні маршрути цими пам'ятками, а також їх зберігати.

В процесі проектування бази даних було виділено наступні сутності:

- категорії (categories);
- всі культурні пам'ятки (places);
- зображення (images);
- користувачі (users);
- обрані пам'ятки (place_user);
- пам'ятки у маршруті (place_route);
- маршрути користувача (route_user);
- всі маршрути (routes).

На рисунку 2.4 приведені представлення моделі бази даних додатку з атрибутами сутностей та зв'язками між таблицями.

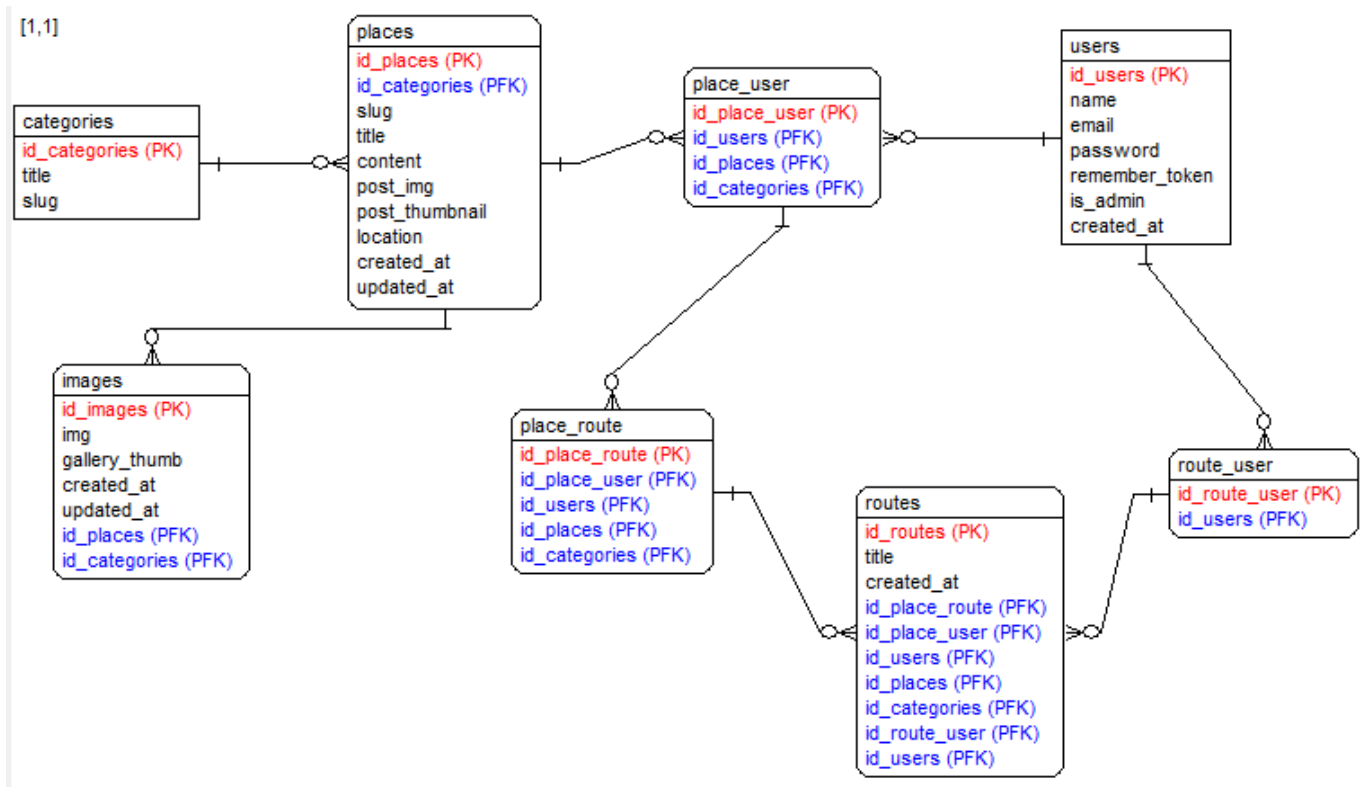


Рисунок 2.4 – Представлення моделі бази даних додатку

Зв'язок сутності «Категорії» з сутністю «Всі культурні пам'ятки» означає, що до однієї категорії може належати декілька пам'яток, але пам'ятка повинна мати власну категорію.

Зв'язок сутності «Всі культурні пам'ятки» з сутністю «Зображення» означає, що у однієї пам'ятки може бути декілька зображень, але зображення повинно відноситися до пам'ятки.

Зв'язок сутності «Користувачі» з сутністю «Обрані пам'ятки» означає, що у користувача може бути декілька обраних пам'яток.

Зв'язок сутності «Всі культурні пам'ятки» з сутністю «Обрані пам'ятки» означає, що обрана пам'ятка може бути обраною декількома користувачами.

Зв'язок сутності «Обрані пам'ятки» з сутністю «Пам'ятки у маршруті» означає, що обрана пам'ятка може бути додана до маршруту.

Зв'язок сутності «Користувачі» з сутністю «Маршрути користувача» означає, що у користувача може бути декілька створених маршрутів.

Зв'язок сутності «Пам'ятки у маршруті» з сутністю «Всі маршрути» означає, що обрані пам'ятки можуть бути додані до маршрутів.

Зв'язок сутності «Маршрути користувача» з сутністю «Всі маршрути» означає, що створені користувачем маршрути можуть бути збережені та додані до списку усіх створених маршрутів.

Проаналізувавши сутності, використовувані в моделі додатку, можна перейти до реалізації структури бази даних. Для цього слід представити назви необхідних таблиць, атрибутів, типів, їх призначення та обмеження (табл. 2.1).

Таблиця 2.1 – Структура бази даних

Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
Категорії	id_categories	Ідентифікатор категорії	bigint(20)	PK	not null
	title	Назва категорії	varchar(255)		not null
	slug	Семантичний URL	varchar(255)		not null
Всі культурні пам'ятки	id_places	Ідентифікатор пам'ятки	int(20)	PK	not null
	id_categories	Ідентифікатор категорії	bigint(20)	PFK	not null
	slug	Семантичний URL	varchar(255)		not null
	title	Назва пам'ятки	varchar(255)		not null
	content	Опис пам'ятки	text		
	post_img	Зображення	varchar(255)		
	post_thumbnail	Мініатюра зображення	varchar(255)		

Продовження таблиці 2.1 – Структура бази даних

Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
Всі культурні пам'ятки	location	Місце розташування пам'ятки	varchar(255)		
	created_at	Коли додана пам'ятка	timestamp		
	updated_at	Коли оновлена пам'ятка	timestamp		
Зображення	id_images	Ідентифікатор зображення	bigint(20)	PK	not null
	id_places	Ідентифікатор пам'ятки	int(20)	PFK	not null
	id_categories	Ідентифікатор категорії	bigint(20)	PFK	not null
	img	Зображення	varchar(255)		
	gallery_thumb	Зображення в галереї	varchar(255)		
	created_at	Коли додано зображення	timestamp		
	updated_at	Коли оновлено зображення	timestamp		
Користувачі	id_users	Ідентифікатор користувача	bigint(20)	PK	not null
	name	Ім'я користувача	varchar(255)		not null
	email	Електронна пошта	varchar(255)		not null
	password	Пароль	varchar(255)		not null

Продовження таблиці 2.1 – Структура бази даних

Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
Користувачі	remember_token	Запам'ятати користувача	varchar(255)		
	is_admin	Чи є користувач адміністратором	tinyint(1)		
	created_at	Коли зареєстрований користувач	timestamp		
Обрані пам'ятки	id_place_user	Ідентифікатор обраної пам'ятки користувача	bigint(20)	PK	not null
	id_users	Ідентифікатор користувача	bigint(20)	PFK	not null
	id_places	Ідентифікатор пам'ятки	int(20)	PFK	not null
	id_categories	Ідентифікатор категорії	bigint(20)	PFK	not null
Пам'ятки у маршруті	id_place_route	Ідентифікатор пам'ятки у маршруті	bigint(20)	PK	not null
	id_place_user	Ідентифікатор обраної пам'ятки користувача	bigint(20)	PFK	not null

Продовження таблиці 2.1 – Структура бази даних

Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
Пам'ятки у маршруті	id_users	Ідентифікатор користувача	bigint(20)	PFK	not null
	id_places	Ідентифікатор пам'ятки	int(20)	PFK	not null
	id_categories	Ідентифікатор категорії	bigint(20)	PFK	not null
Маршрути користувача	id_route_user	Ідентифікатор маршруту користувача	bigint(20)	PK	not null
	id_users	Ідентифікатор користувача	bigint(20)	PFK	not null
Всі маршрути	id_routes	Ідентифікатор маршрутів	bigint(20)	PK	not null
	title	Назва маршруту	varchar(255)		not null
	created_at	Коли створений маршрут	timestamp		
	id_place_route	Ідентифікатор пам'ятки у маршруті	bigint(20)	PFK	not null
	id_place_user	Ідентифікатор обраної пам'ятки користувача	bigint(20)	PFK	not null
	id_users	Ідентифікатор користувача	bigint(20)	PFK	not null

Продовження таблиці 2.1 – Структура бази даних

Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
Всі маршрути	id_places	Ідентифікатор пам'ятки	int(20)	PFK	not null
	id_categories	Ідентифікатор категорії	bigint(20)	PFK	not null
	id_route_user	Ідентифікатор маршруту користувача	bigint(20)	PFK	not null

3 РОЗРОБКА WEB-ДОДАТКУ ПІДТРИМКИ ВИБОРУ ТУРИСТИЧНОГО МАРШРУТУ ПАМ'ЯТКАМИ СУМЩИНИ

3.1 Архітектура Web-додатку

Архітектура додатку (рис. 3.1) дозволяє отримати представлення структури його сторінок або програмної частини.

Вона складається з бази даних на основі MySQL, яка містить дані про культурні пам'ятки, отримані за допомогою парсера hQuery зі стороннього ресурсу. Ці дані передаються на сервер та оброблюються фреймворком Laravel. У Laravel для роботи з базою даних використовуються моделі. З їх допомогою дані передаються на контролери, де відбувається їх обробка. Далі дані передаються на шаблони. Після виконання цих дій можна бачити відображення сторінок створеного додатку.

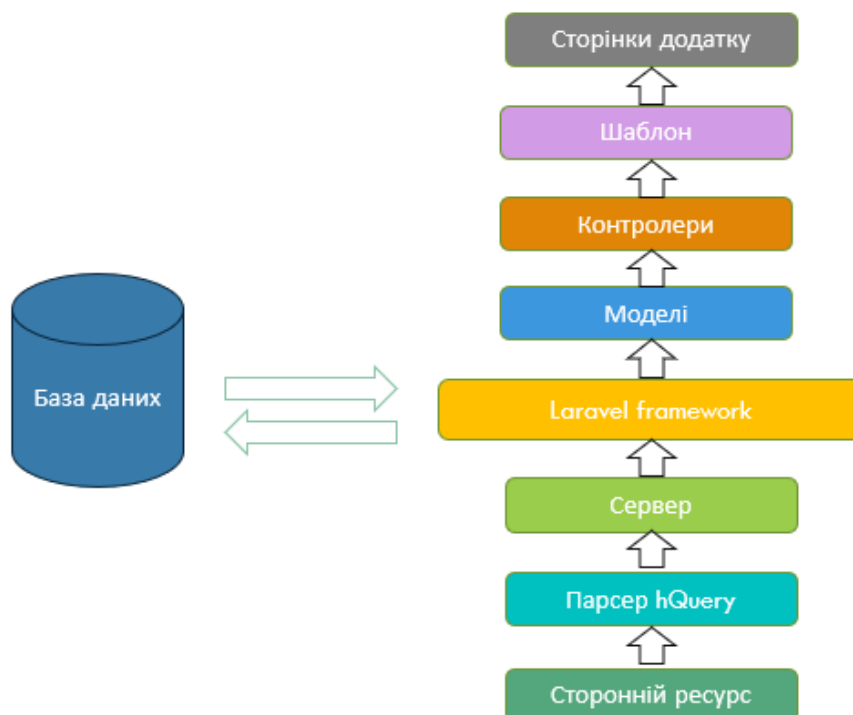


Рисунок 3.1 – Архітектура Web-додатку

3.2 Реалізація Web-додатку

Реалізація Web-додатку підтримки вибору туристичного маршруту включає в себе розробку сторінок, їх зовнішнього вигляду та створення функціональних модулів.

Першим кроком було розроблення шаблону (рис. 3.2), що означало виконання HTML-розмітки сторінок та їх стилістичне оформлення. При розробці були використані наступні компоненти Bootstrap:

- компонент navbar та супутні класи для горизонтального адаптивного меню додатку;
- класи text right, left, center для вирівнювання [20].

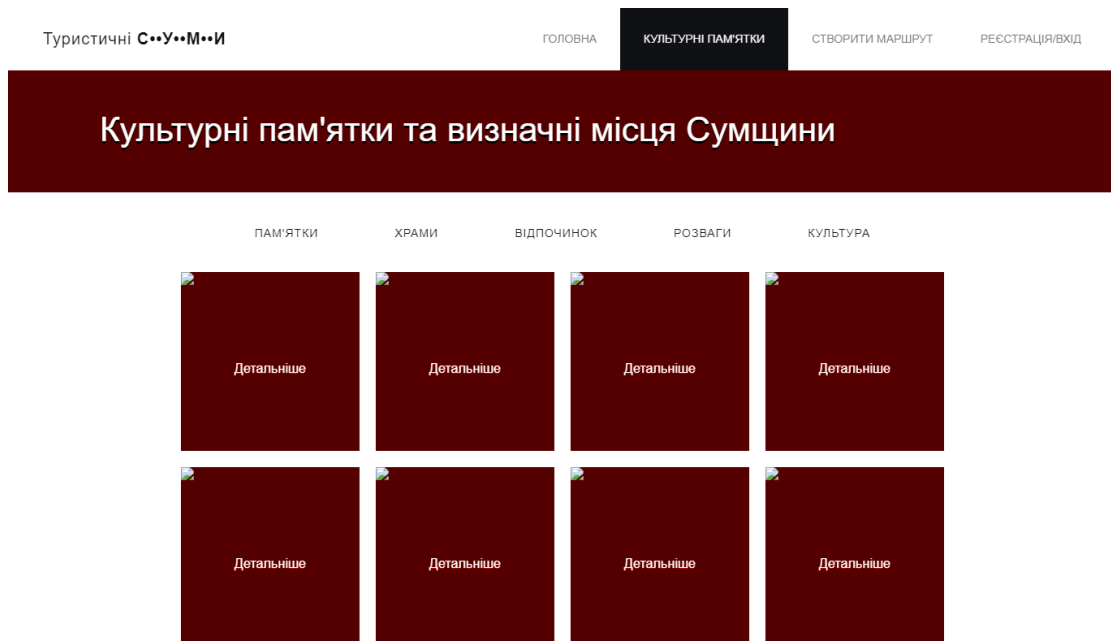


Рисунок 3.2 – Фрагмент шаблону сторінки з культурними пам'ятками

Для збору даних про культурні пам'ятки Сумщини був обраний онлайн путівник [17], який містить достовірну інформацію про визначні місця Сумщини.

Попередньо через інспектор об'єктів на сторінках сайту-джерела були виділені класи з необхідними для парсингу даними (рис. 3.3-3.4):

- блок списку пам'яток `newsListItemsHolder`;
- клас окремої туристичної пам'ятки `newsListItem`;
- клас заголовку туристичної пам'ятки `newsListItemTextTitle`;
- клас локації туристичної пам'ятки `newsListItemTextAddress`;
- клас інформації про окрему пам'ятку `tour-description-text`.

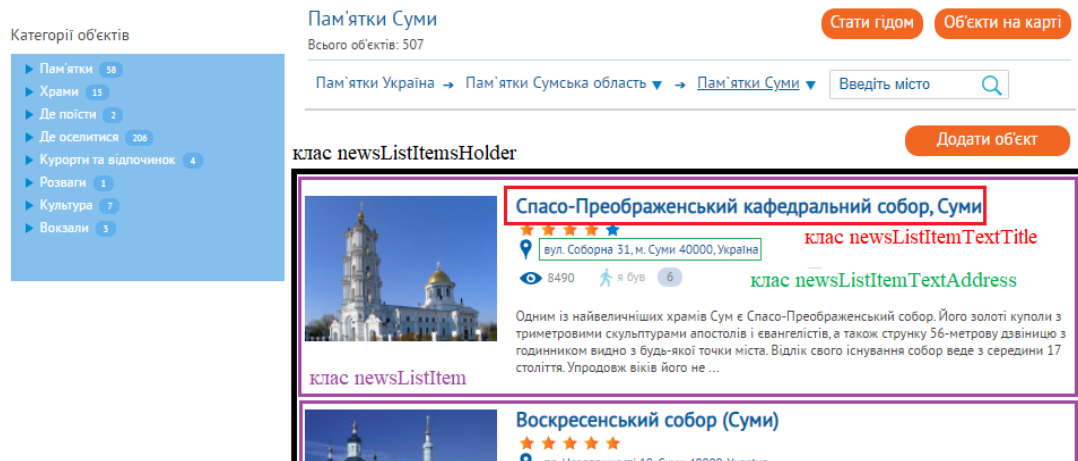


Рисунок 3.3 – Аналіз фрагменту сторінки сайту-джерела

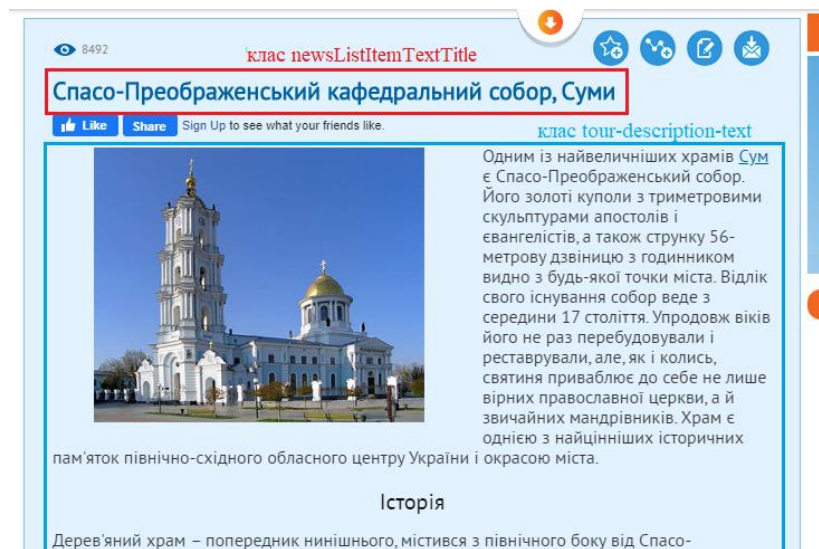


Рисунок 3.4 – Аналіз фрагменту сторінки сайту-джерела

Використовуючи РНР бібліотеку `hQuery` за допомогою функції `Create` контролера `ParseController` (рис. 3.5) з обраної сторінки збираються дані про туристичні пам'ятки. Після цього зібрані дані передаються до шаблону парсера для

подальшої обробки. За допомогою перебору отриманих об'єктів (рис. 3.6) були отримані заголовки, посилання на внутрішні сторінки кожної пам'ятки, а також місцезнаходження і зображення. Використання регулярних виразів дозволяє провести очищення даних від непотрібних об'єктів.

```
public function create(Request $request)
{
    $url = $request->input('url');
    $doc = hQuery::fromFile($url)->find("#newsListItemsHolder .newsListItem");
    $user = \Auth::user();
    $categories = Category::all();
    if($user && $user->is_admin == 1) {
        return view('admin.parser.create', compact('user', 'doc', 'categories'));
    }
    else {
        return redirect('/login');
    }
}
```

Рисунок 3.5 – Функція збору даних із сайту-джерела

```
<?php
foreach($doc as $key => $value) {
    $single_page = $value->find("a.newsListItemTextTitle.poiItemTitle")->attr('href');
    $dom1 = hQuery::fromFile($single_page);
    ...
    ...<?=$value->find("a.newsListItemTextTitle.poiItemTitle")->text();?>
    ...<?preg_replace('/\s?<ins[^\s]*?>.*?<\ins>\s?/si', '',
        preg_replace('/\s?<script[^\s]*?>.*?</script>\s?/si', '',
        preg_replace('/<div.*?>(div.*?)(?1)*?</div>|.)*?</div>/is', '',
        $dom1->find("div.tour-description-text")->html());
    ?>
    ...
    ...<?php if($dom1->find(".poi-image-holder img")) {
        echo $dom1->find(".poi-image-holder img")->attr('src'); }
        else { echo '...'; }
    ?>
    ...
    ...<?php
        $images = $dom1->find(".thslide_list ul li img");
        $links = array();
        if($images) {
            foreach ($images as $key => $a) {
                $links[$key] = $a->attr('src');}
            }
        ?>
    ...<?=$value->find("div.newsListItemIcons.newsListItemTextAddress")->text();?>
    ...
    ...<?php
}
?>
```

Рисунок 3.6 – Фрагмент перебору отриманих об'єктів сайту-джерела

Після цього необхідно організувати зберігання оброблених даних до бази даних. Воно виконується за допомогою функції Store контролера ParseController (додаток В). Після збереження інформації про культурні пам'ятки (рис. 3.7) їх вивід на шаблон сторінки додатку відбувається за допомогою функції Index контролера PlacesController (рис. 3.8). Шаблон сторінки окремої визначної пам'ятки розміщено у додатку Г.

cat_id	slug	title	content	post_img	post_thumbnail	location	created_at	updated_at
1	budinok-muzej-antona-chehova	Будинок-музей Антона Чехова	<p>На мальовничій околиці Сум – Луці розміщено Буд...	img-budinok-muzej-antona-chehova-full.jpg	thumb-budinok-muzej-antona-chehova-220x220.jpg	вул. Чехова 79, Суми 40000, Україна	2020-05-27 15:41:03	2020-05-27 15:41:03
1	sums-ka-al-tanka	Сумська альтанка	<a href="#" class="descriptionAutoTranslateTitle" ...	img-sums-ka-al-tanka-full.jpg	thumb-sums-ka-al-tanka-220x220.jpg	пл. Покровська, м. Суми 40000, Сумська обл., Украі...	2020-05-27 15:41:09	2020-05-27 15:41:09
1	sums-kiy-oblasniy-hudozhniy-muzej-im-nikanora-onac...	Сумський обласний художній музей ім. Никанора Онац...	<p>Сумський обласний музей ім. Никанора Онацького ...	img-sums-kiy-oblasniy-hudozhniy-muzej-im-nikanora-...	thumb-sums-kiy-oblasniy-hudozhniy-muzej-im-nikanor...	вул. Покровська площа 1, Суми 40000, Україна	2020-05-27 15:41:12	2020-05-27 15:41:12
1	sums-kiy-oblasniy-krayeznavchiy-muzej	Сумський обласний краєзнавчий музей	<p>В музеї зберігається майже 45 тисяч оригіналини...	img-sums-kiy-oblasniy-krayeznavchiy-muzej-full.jpg	thumb-sums-kiy-oblasniy-krayeznavchiy-muzej-220x22...	вул. Герасима Кондратьєва 2, Суми 40030, Україна	2020-05-27 15:41:18	2020-05-27 15:41:18
1	muzej-istoriyi-sums-kogo-polku	Музей історії Сумського полку	<a href="#" class="descriptionAutoTranslateTitle" ...	img-muzej-istoriyi-sums-kogo-polku-full.jpg	thumb-muzej-istoriyi-sums-kogo-polku-220x220.jpg	вул. Воскресенська 2, м. Суми 40000, Сумська обл.,...	2020-05-27 15:41:23	2020-05-27 15:41:23
1	pam-yatnik-cukru-sumi	Пам'ятник цукру, Суми	<a href="#" class="descriptionAutoTranslateTitle" ...	img-pam-yatnik-cukru-sumi-full.jpg	thumb-pam-yatnik-cukru-sumi-220x220.jpg	пл. Покровська, м. Суми 40000, Сумська обл., Украі...	2020-05-27 15:41:26	2020-05-27 15:41:26
1	mauritan	Майдан Незалежності	<a href="#"	img-mauritan	thumb-mauritan	пл. Незалежності	2020-05-27 15:41:30	2020-05-27 15:41:30

Рисунок 3.7 – Фрагмент заповненої таблиці бази даних з інформацією про культурні пам'ятки

```
public function index()
{
    $user = \Auth::user();
    $places = Place::paginate(12);
    $categories = Category::all();
    return view('front-end.archive', compact('places', 'user', 'categories'));
}
```

Рисунок 3.8 – Функція виводу інформації про культурні пам'ятки

Авторизований користувач повинен мати змогу додавати будь-яку пам'ятку до обраних пам'яток для подальшого створення маршруту. Тому було розроблено окрему функцію Store у контролері ParseController для виконання цієї дії (рис. 3.9).

```

public function store(Request $request)
{
    $user = \Auth::user();
    $place = $request->input('place_id');
    if(!($user->places()->where('place_id', $place)->exists())) {
        $user->places()->attach($place);
    }
    return redirect('/places');
}

```

Рисунок 3.9 – Функція додавання до обраних культурної пам’ятки

Створення туристичного маршруту обраними пам’ятками відбувається за рахунок функції Create контролера RoutesController (рис. 3.10). Побудова маршруту на карті відбувається за допомогою скрипту, який наведено у додатку Д.

```

public function create()
{
    $user = \Auth::user();
    $places = Place::all();
    return view('front-end.account.createroute', compact('places', 'user'));
}

```

Рисунок 3.10 – Функція створення туристичного маршруту

Побудова збереженого маршруту відбувається за допомогою скрипту, який наведено у додатку Е.

Після розробки додатку його було завантажено на хостинг HOSTiQ, який вважають одним з найкращих хостингів на ринку в Україні. Для завантаження додатку було обрано тарифний план хостингу HQ10 з реєстрацією домену водночас. Для додатку було обране доменне ім’я touristsumy.online (рис. 3.11).

The screenshot shows a web form titled "Оберіть доменне ім'я" (Choose domain name). It has two radio buttons: "Я хочу зареєструвати новий домен" (I want to register a new domain) which is selected, and "У мене вже є домен" (I already have a domain). Below the buttons is a text input field containing "www. touristsumy.online" and a "Перевірити" (Check) button. At the bottom, there is a green checkmark and the text "Вітаємо! Домен доступний для реєстрації." (Congratulations! Domain is available for registration).

Рисунок 3.11 – Реєстрація доменного імені

Після сплати послуг наступним кроком було створення бази даних на хостингу, якій було присвоєне ім'я tourist_sumytour. Далі за допомогою FTP-клієнта FileZilla відбулося підключення до серверу хостингу та завантажено до папки public_html файлів додатку (рис. 3.12).

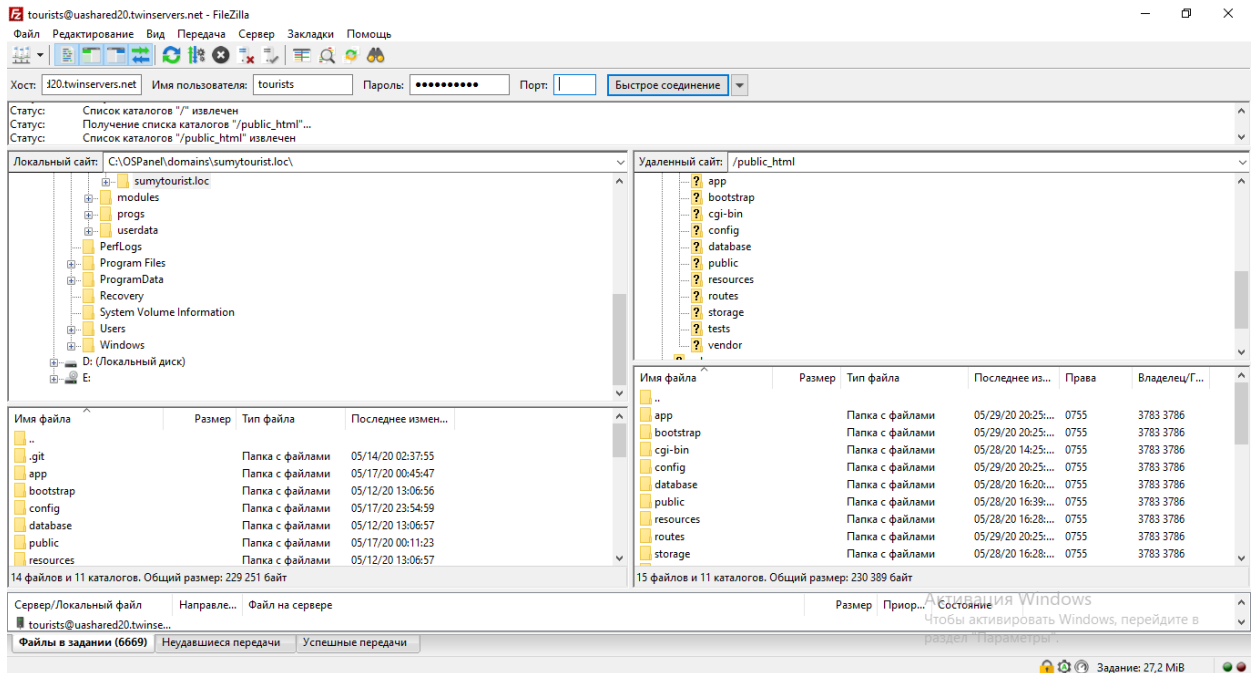


Рисунок 3.12 – Підключення до хостингу за протоколом FTP

Після підключення бази даних та налаштувань установки Web-додаток було встановлено для подальшого повноцінного функціонування.

3.3 Використання Web-додатку

Доступ до Web-додатку можна здійснити за посиланням – <http://touristsumy.online>. При його завантаженні користувач потрапляє на головну сторінку додатку (рис. 3.13), де розміщена довідкова інформація та посилання для перегляду визначних пам'яток (рис. 3.14) або створення маршруту (рис. 3.20). Також

у головному меню є можливість переходу на сторінку авторизації/реєстрації. (рис. 3.19) У підвалі сайту розміщена інформація про розробника.

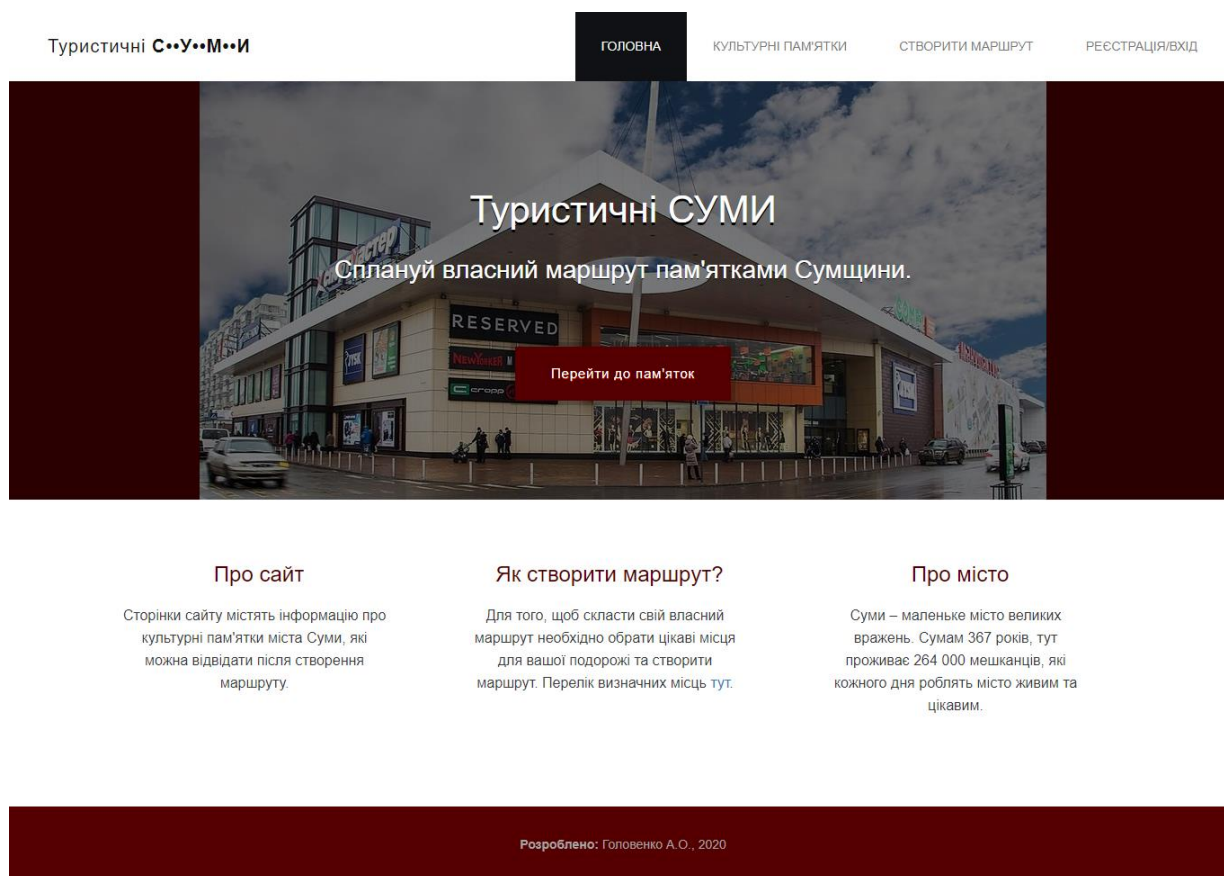


Рисунок 3.13 – Головна сторінка розроблюваного додатку

Після переходу до розділу меню з культурними пам'ятками користувач потрапляє на сторінку з їх переліком (рис. 3.14), розподіленими на категорії: «Пам'ятки», «Храми», «Відпочинок», «Розваги», «Культура».

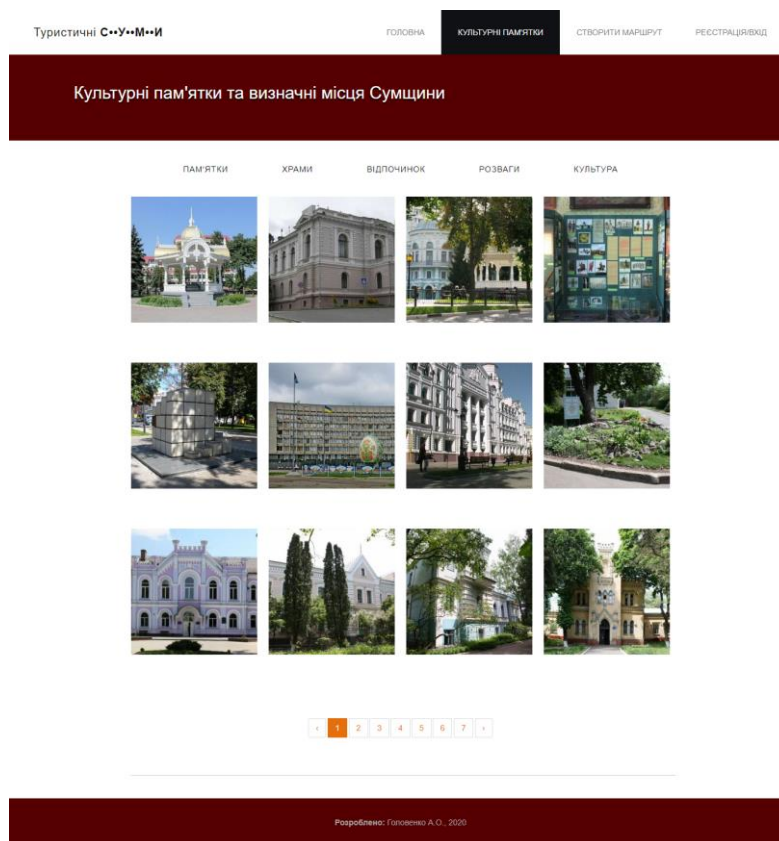


Рисунок 3.14 – Сторінка розділу «Культурні пам’ятки» розроблюваного додатку категорії «Пам’ятки»

При наведенні курсору на окрему картку пам’ятки з’являється її назва (рис. 3.15). При натисненні на зображення потрібної для перегляду пам’ятки відкривається сторінка з її детальним описом (рис. 3.16-3.18).

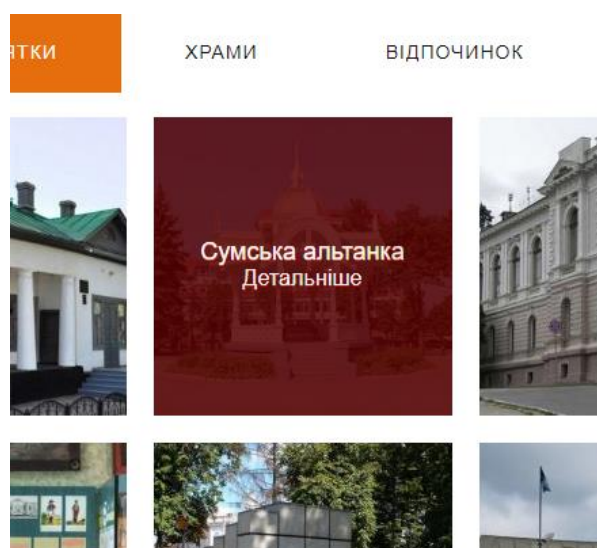



Рисунок 3.15 – Результат наведення курсору миші на окрему пам’ятку

Для кожної туристичної пам'ятки передбачена окрема сторінка для отримання детальної інформації, за її наявності на сайті, звідки беруться дані, – галереї зображень, історичної довідки та місце розташування на карті (рис. 3.16-3.18)

Туристичні С•У•М•И ГОЛОВНА КУЛЬТУРНІ ПАМ'ЯТКИ РЕЄСТРАЦІЯ/ВХІД

Сумська альтанка

Додати до обраних




Інформація

У кожному місті є свій символ - незвичайна споруда, виконана в оригінальному і неповторному стилі. Таким символом є місто **Суми** - альтанка.

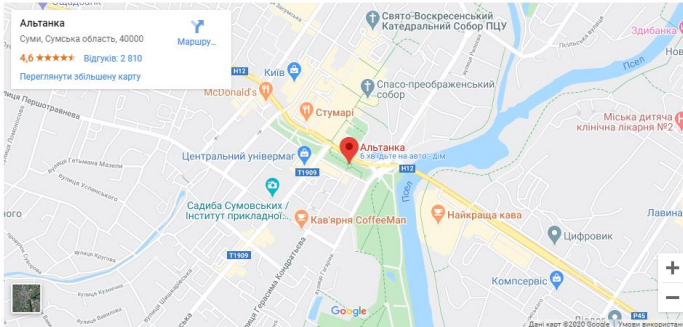
Унікальністю альтанки є **принцип її побудови без єдиного цвяжка**. Вона вся виконана з дерева. Другою особливістю будови стало те, що за всі роки війни воно нітрохи не пострадало. Її будівництво почалося в кінці 19 століття, у зв'язку з тим, що геологи з Санкт-Петербурга шукали в Сумах нафту, але не знайшовши її залишили свердловину. Після чого заповзятливий місчанин І. Лещинський купив цю ділянку землі. Вирішивши скористатися свердловиною в майбутньому або передати її нащадкам, він закрив отвір металеві плитою і поставив на тому місці **виточену альтанку з різьбленого дерева**.

Виконана альтанка у вигляді відкритої тераси з вісьмома гранями на цегельному фундаменті. Дах альтанки вічна гранований купол. Споруда височіє на 10 метрів, а її діаметр становить трохи менше 6 метрів. Красу оздоблення доповнюють різьблені шпиль, мереживні карнизи і різні декоративні візерунки. Альтанку оточує клумба з квітами і бузок.

Рисунок 3.16 – Фрагмент сторінки «Сумська альтанка»



На карті



Розроблено: Головенко А.О., 2020

Рисунок 3.17 – Фрагмент сторінки «Сумська альтанка»

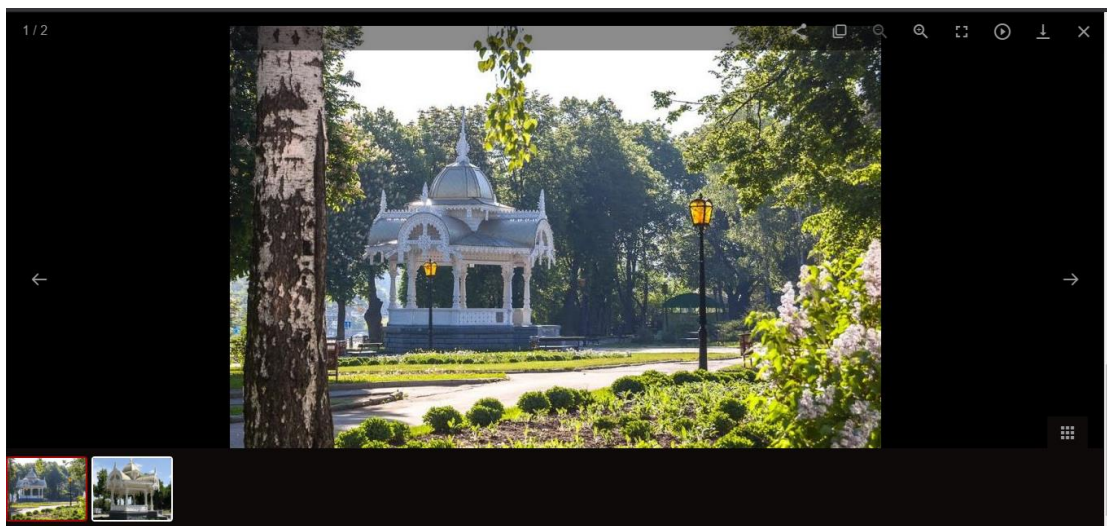


Рисунок 3.18 – Фрагмент сторінки «Сумська альтанка», галерея зображень

Після перегляду інформації про пам'ятку, за допомогою відповідної кнопки, її можна додати до обраних. Для цього користувач повинен бути авторизованим. Якщо раніше авторизація не відбувалась, при додаванні пам'ятки до обраних користувач буде направлений на сторінку авторизації (рис. 3.19).

Рисунок 3.19 – Сторінка авторизації

Після реєстрації/авторизації користувач має можливість додавати до обраних цікаві для нього визначні пам'ятки Сумщини, а в подальшому створювати власний туристичний маршрут. Цікаві для користувача пам'ятки, після додавання до обраних,

знаходяться у розділі «Створити маршрут», який з’являється після авторизації (рис. 3.20).

У цьому розділі є можливість ще раз переглянути детальну інформації про обрані туристичні пам’ятки. Для того, щоб створити власний туристичний маршрут, користувачу потрібно виділити бажані туристичні пам’ятки та натиснути кнопку «Створити маршрут».

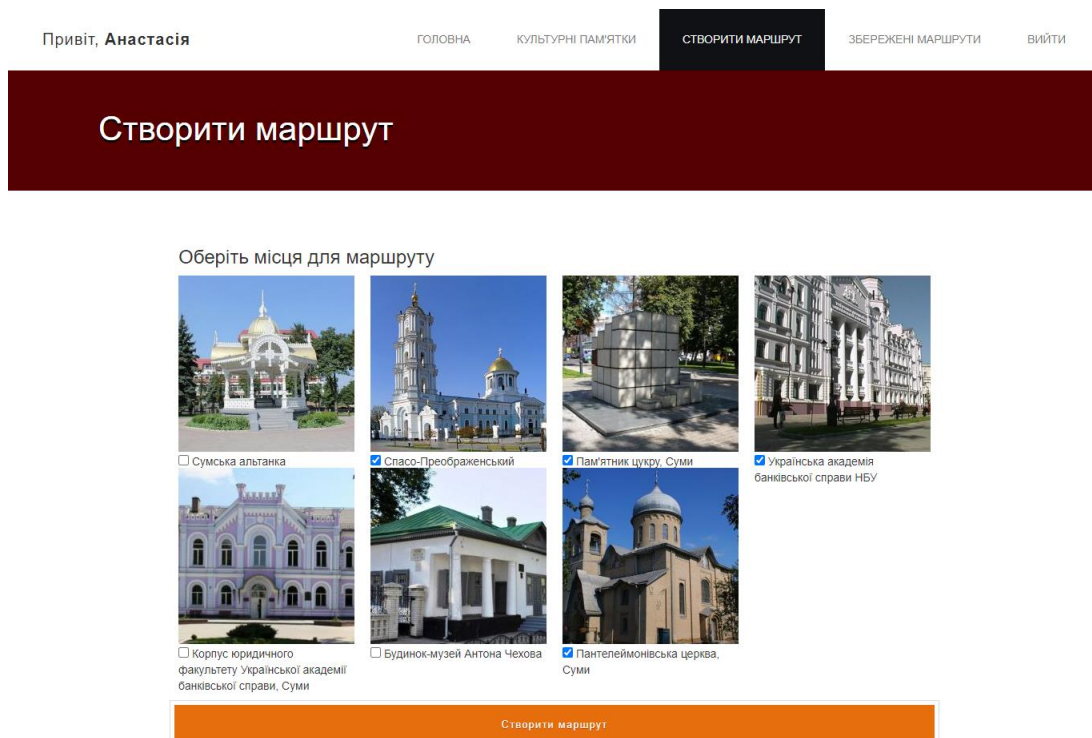
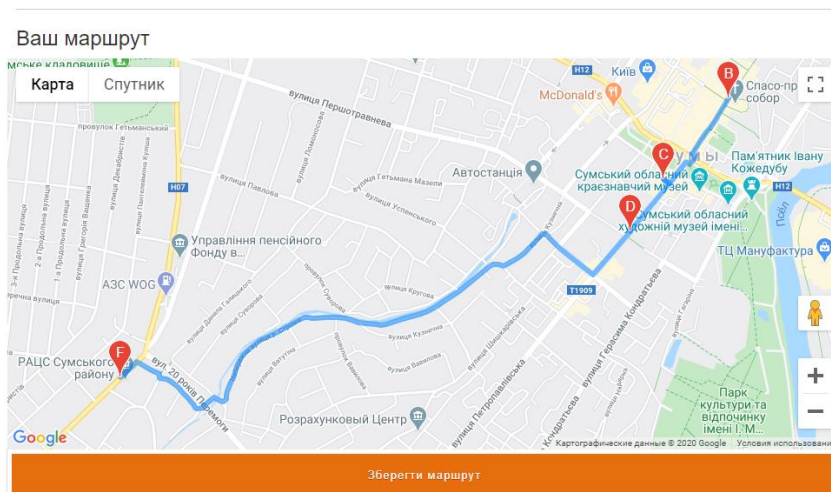


Рисунок 3.20 – Фрагмент сторінки «Створити маршрут»

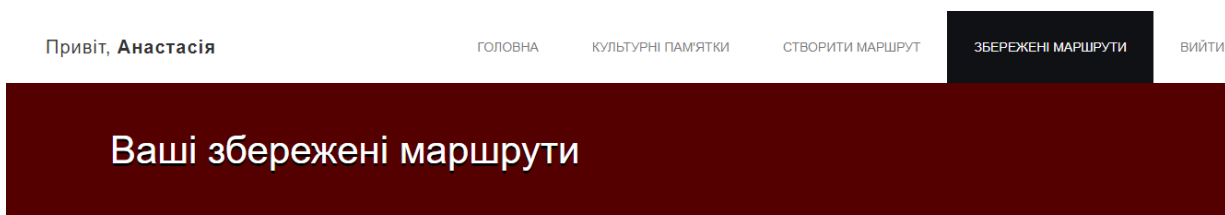
Після виконання цих дій на карті нижче прокладається пішохідний маршрут обраними пам’ятками (рис. 3.21), який можна збільшувати для детального перегляду або розширити на весь екран пристрою.



Розроблено: Головенко А.О., 2020

Рисунок 3.21 – Фрагмент сторінки «Створити маршрут»

Якщо користувача влаштовує створений маршрут, то він може його зберегти для подальшого користування натиснувши відповідну кнопку збереження. Створені та збережені користувачем маршрути можна не лише переглядати, а й видаляти за бажанням у відповідному розділі меню (рис. 3.22).



Номер	Маршрут	Видалити маршрут
1	Сумська альтанка -> Спасо-Преображенський кафедральний собор, Суми -> Пам'ятник цукру, Суми -> Українська академія банківської справи НБУ -> Корпус юридичного факультету Української академії банківської справи, Суми	Видалити
2	Спасо-Преображенський кафедральний собор, Суми -> Пам'ятник цукру, Суми -> Українська академія банківської справи НБУ -> Пантелеймонівська церква, Суми	Видалити

Розроблено: Головенко А.О., 2020

Рисунок 3.22 – Сторінка збережених маршрутів

Сторінки додатку мають адаптивний дизайн. Для демонстрації адаптивності сторінок при різних розмірах екрану були обрані сторінки з культурними пам'ятками та сторінка створення маршруту.

Вигляд фрагменту сторінки з культурними пам'ятками при ширині екрану від 425 до 768 пікселів відображено на рисунку 3.23.

Вигляд сторінки створення маршруту при ширині екрану від 425 до 768 пікселів відображено на рисунку 3.24-3.25.

Вигляд сторінки з культурними пам'ятками при ширині екрану до 425 пікселів відображено на рисунку 3.26.

Вигляд сторінки створення маршруту при ширині екрану до 425 пікселів відображено на рисунку 3.27-3.28.

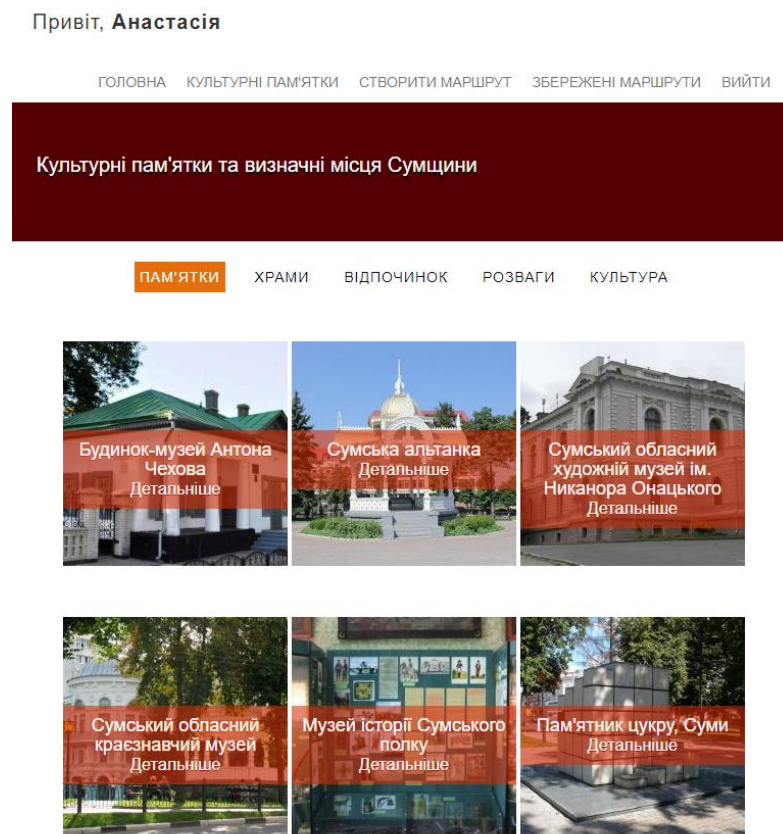


Рисунок 3.23 – Фрагмент сторінки «Культурні пам'ятки»

Привіт, **Анастасія**ГОЛОВНА КУЛЬТУРНІ ПАМ'ЯТКИ **СТВОРИТИ МАРШРУТ** ЗБЕРЕЖЕНІ МАРШРУТИ ВІЙТИ

Створити маршрут

Оберіть місця для маршруту

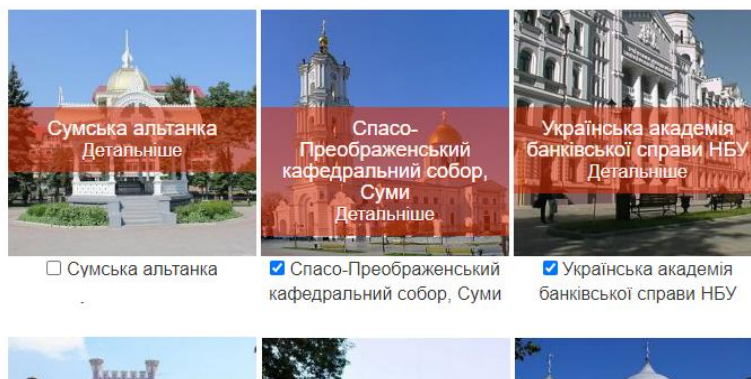


Рисунок 3.24 – Фрагмент сторінки «Створити маршрут»

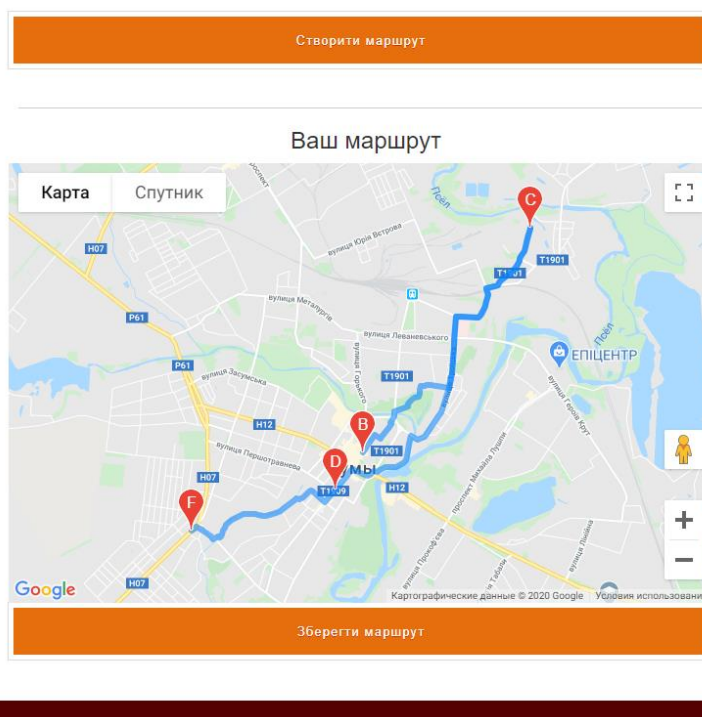


Рисунок 3.25 – Фрагмент сторінки «Створити маршрут»

При зменшенні ширини екрану до 425 пікселів головне меню приховується, блоки з культурними пам'ятками розміщуються один під одним.

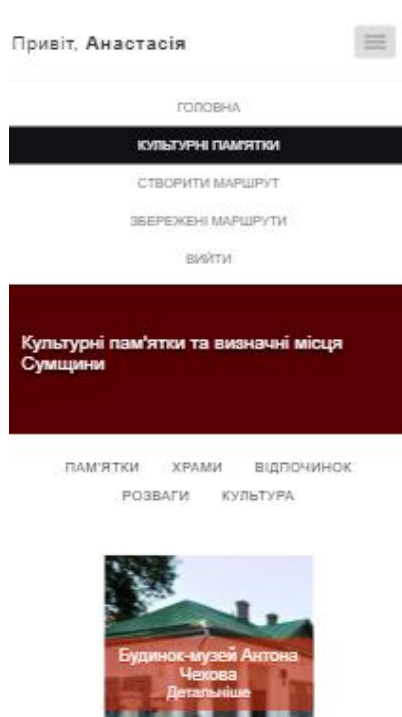


Рисунок 3.26 – Фрагмент сторінки «Культурні пам'ятки»



Рисунок 3.27 – Фрагмент сторінки «Створити маршрут»

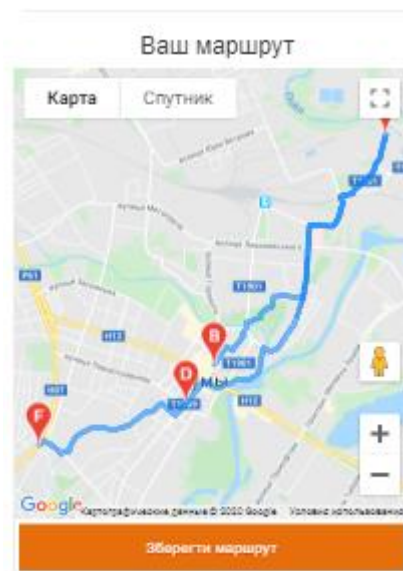


Рисунок 3.28 – Фрагмент сторінки «Створити маршрут»

Для додання, оновлення чи видалення інформації про туристичні пам'ятки необхідно авторизуватися в додатку з даними адміністратора. Після авторизації відривається головна сторінка панелі адміністратора (рис. 3.29), де відображена наявна інформація про туристичні пам'ятки Сумщини.

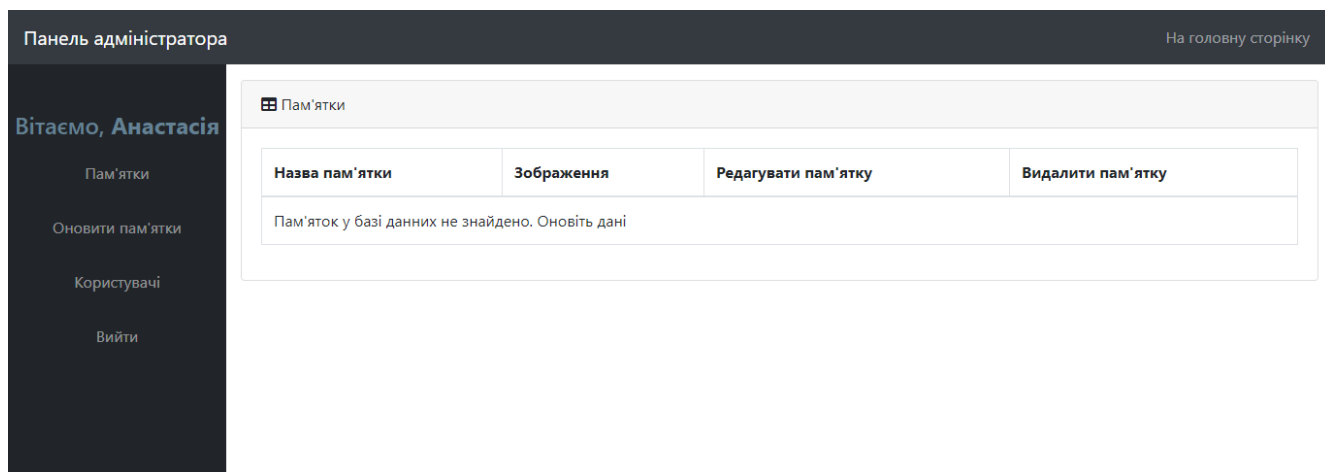


Рисунок 3.29 – Сторінка «Панель адміністратора»

Для додавання даних або їх оновлення необхідно перейти до відповідного розділу меню «Оновити пам'ятки». На цій сторінці розміщена форма (рис. 3.31), до

якої необхідно ввести посилання на сайт-джерело, звідки будуть братися дані про визначні пам'ятки (рис. 3.30).

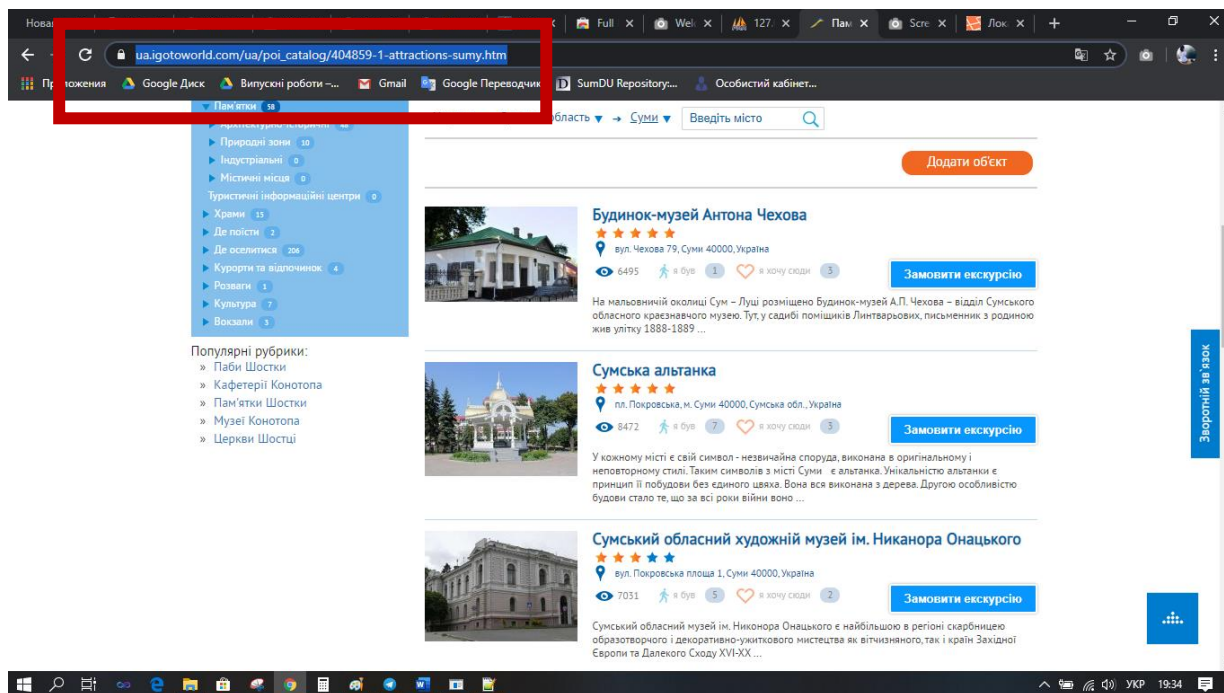


Рисунок 3.30 – Сторінка сайту ua.igotoworld.com

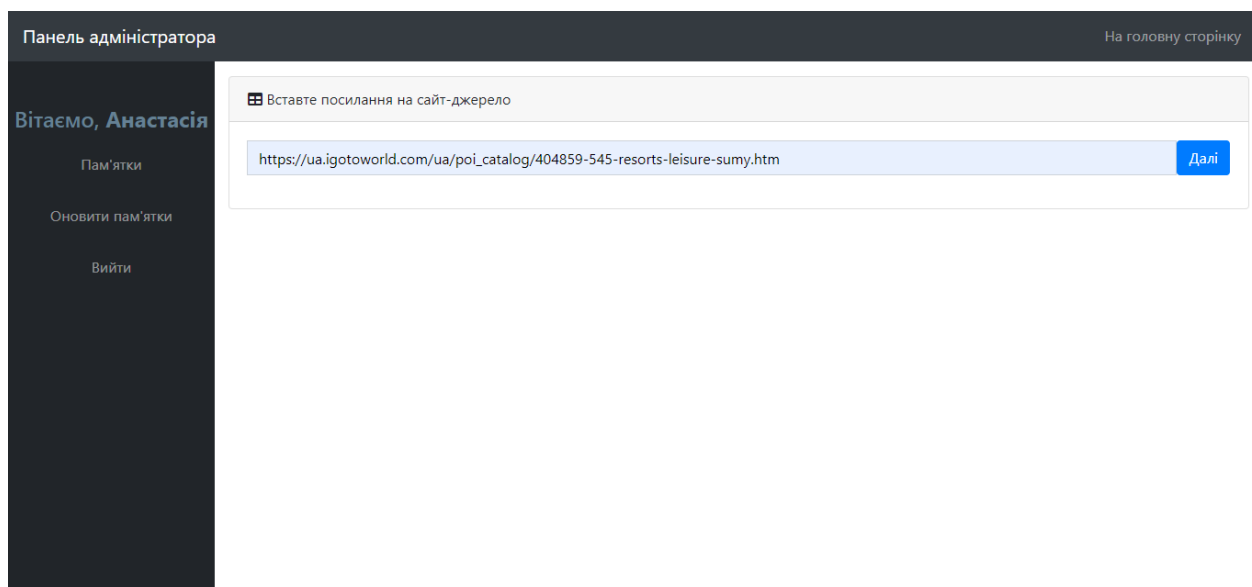


Рисунок 3.31 – Сторінка «Панель адміністратора»

Після відправки форми зібрані дані потрапляють у форму обробки даних (рис. 3.32), де можна відредагувати заголовок пам'ятки, інформацію про неї та обрати для неї категорію. Для того, щоб зберегти ці дані до бази даних, необхідно натиснути «Відправити дані».

Скульптури Ангел і Голгофа	він з кінця 18 століття. Надгробні пам'ятники робили майстри з		вул. 20-річчя Перемоги	Пам'ятки
Пам'ятник Собаці, Суми	пам'ятник Собаці, який стоїть у		вул. Пролетарська 10, С	Пам'ятки
Пам'ятник невідомому солд	солдату є частиною <a href="https://ua.igotoworld.com/u		вул. Герасима Кондрат	Пам'ятки
Пам'ятник Лушпі, Суми	пам'ятник Михайлу Лушпі, який вважається почесним		пр. Тараса Шевченка 25	Пам'ятки

Відправити дані

Рисунок 3.32 – Фрагмент сторінки отримання даних

Після цього дані можна буде переглянути, відредагувати у майбутньому (рис. 3.34) або видалити у пункті меню «Пам'ятки» (рис. 3.33).

Панель адміністратора На головну сторінку

Вітаємо, Анастасія

- Пам'ятки
- Оновити пам'ятки
- Вийти

Пам'ятки

Назва пам'ятки	Зображення	Редагувати пам'ятку	Видалили пам'ятку
Будинок-музей Антона Чехова		Редагувати	Видалили
Сумська альтанка		Редагувати	Видалили
Сумський обласний художній музей ім. Никанора Онацького		Редагувати	Видалили

Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".

Рисунок 3.33 – Фрагмент сторінки «Панель адміністратора»

Для кожної пам'ятки можна змінити назву, інформацію про неї, змінити категорію чи місцезнаходження обравши пам'ятку та натиснувши Редагувати (рис. 3.34).

Панель адміністратора На головну сторінку

Вітаємо, Анастасія

- Пам'ятки
- Оновити пам'ятки
- Вийти

Назва пам'ятки

Контент

```
<a href="#" class="descriptionAutoTranslateTitle" title="Автоматичний переклад"> </a>
<p> У кожному місті є свій символ - незвичайна споруда, виконана в оригінальному і неповторному стилі. Таким символів з місті <a href="https://ua.igotoworld.com/ua/geo_object/404859.htm">Суми</a> <span class="apple-converted-space"> </span> є альтанка. </p><p>
```

Категорія

Локація

Розроблено: Головенко А.О., 2020

Рисунок 3.34 – Сторінка редагування пам'ятки

ВИСНОВКИ

На сьогоднішній день за допомогою всіх сучасних пристроїв людське життя стало набагато простіше, а головне – зручніше. Це також стосується подорожей і туризму. Для спрощення планування подорожей люди можуть користуватися як туристичними агентствами, так і сервісами планування подорожей. Саме тому існує потреба у створенні адаптивного додатку, який дозволить швидко спланувати поїздки без марнування часу та грошей. Такий додаток повинен не лише заощаджувати сили, гроші та час в пошуках місця проведення відпочинку, а й бути зручним для користувачів.

Під час виконання роботи було визначено об'єкт та предмет дослідження. Аналіз предметної області дозволив виявити основні характерні для роботи сутності та їх зв'язки для подальшого моделювання. Аналіз програмних продуктів-аналогів дозволив зрозуміти які недоліки присутні у даних сервісах та як їх уникнути, а також які переваги допоможуть у проектуванні. При виборі засобів реалізації було обрано виконувати розробку використовуючи фреймворк Laravel. Процес моделювання додатку під час виконання роботи дозволив візуалізувати систему, визначити структуру та її поведінку. Під час планування було виділено та структуровано елементарні роботи, які необхідно виконати в ході проекту для оцінки бюджету проекту та ризиків, призначено відповідального та побудовано матрицю відповідальності. За допомогою діаграми Ганта були визначені часові рамки для виконання кожної з робіт, після чого побудовано PDM-мережа. Описання проекту на фазі планування представлено у додатку Б.

Метою роботи було створення Web-додатку підтримки вибору туристичного маршруту пам'ятками Сумщини. Для її досягнення необхідно було реалізувати можливість прокладати туристичний маршрут на карті культурними пам'ятками. Таке рішення дозволяє скоротити час на самостійне планування, а також дає можливість зберегти щойно створений маршрут для подальшого користування. Інформацію про культурні пам'ятки Сумщини необхідно було отримати з надійних

електронних ресурсів, щоб інформація на сторінках додатку була актуальною. Для реалізації був використаний парсер, який дозволяє не лише отримувати дані про визначні місця, а й редагувати їх. Маршрут кожного користувача є унікальним, тому бажаний створити маршрут повинен авторизуватися у додатку. Створений модуль складання маршруту передбачає наявність обраних пам'яток у особистому кабінеті користувача. До моменту збереження маршруту до особистого кабінету пам'ятки можна додавати, видаляти чи взагалі не зберігати складений маршрут. В свою чергу, збережені маршрути користувач може переглянути у особистому кабінеті без можливості зміни, але може його видалити за бажанням.

Надання інформації про туристичні пам'ятки та планування індивідуального маршруту ними може заохотити людей відвідувати місто, що безперечно матиме позитивний соціально-культурний вплив. Оскільки поставлені задачі на початку виконання проєкту були виконанні у повному обсязі, то у результаті роботи був розроблений продукт, який надає підтримку вибору туристичного маршруту пам'ятками Сумщини. Роботу було оприлюднено на науково-технічній конференції ІМА-2020. Теза доповіді розміщена у додатку Ж.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Розвиток туристичної галузі України: особливості та проблеми: веб-сайт. URL: http://tourlib.net/statti_ukr/miroshnyk.htm (дата звернення: 10.04.2020).
2. Free Travel – самостоятельные путешествия: веб-сайт. URL: <https://freetravel.com.ua/> (дата звернення: 11.04.2020).
3. TripAdvisor: веб-сайт. URL: <https://www.tripadvisor.ru/> (дата звернення: 11.04.2020).
4. Что такое фреймворки и для чего они нужны?: веб-сайт. URL: <https://saikt-online.ru/chto-takoe-frejmworki-i-dlya-chego-oni-nuzhny/> (дата звернення: 11.04.2020).
5. 10 Popular PHP frameworks in 2019: веб-сайт. URL: <https://habr.com/ru/post/464417/> (дата звернення: 12.04.2020).
6. Что такое Bootstrap и зачем он нужен?: веб-сайт. URL: <https://itchief.ru/bootstrap/introduction> (дата звернення: 12.04.2020).
7. ЧТО ТАКОЕ API GOOGLE КАРТЫ?: веб-сайт. URL: <https://maps-creator.com/blog/what-is-google-maps-api.html> (дата звернення: 13.04.2020).
8. Краткое введение в Bootstrap: веб-сайт. URL: <https://tproger.ru/translations/bootstrap-short-intro/> (дата звернення: 14.04.2020).
9. Что такое парсинг сайтов?: веб-сайт. URL: <https://msiter.ru/articles/chto-takoe-parsing-saytov> (дата звернення: 15.04.2020).
10. Google Maps API: веб-сайт. URL: <https://habr.com/ru/post/110460/> (дата звернення: 15.04.2020).
11. Методология IDEF0: веб-сайт. URL: https://studme.org/87184/ekonomika/metodologiya_idef0 (дата звернення: 17.04.2020).
12. Методология IDEF0: веб-сайт. URL: <https://itteach.ru/bpwin/metodologiya-idef0> (дата звернення: 17.04.2020).
13. МЕТОДОЛОГИЯ IDEF0: веб-сайт. URL: https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema6/tema6_2 (дата звернення: 17.04.2020).

14. Методология процесса моделирования: веб-сайт. URL: <http://idef.ru/idef.php> (дата звернення: 18.04.2020).
15. Отличия uml от idef0, dfd: веб-сайт. URL: <https://studfile.net/preview/5999051/page:2/> (дата звернення: 19.04.2020).
16. Создание ER-Диаграмм: веб-сайт. URL: <http://inf-teh-lotos.ru/sozdanie-er-diagramm> (дата звернення: 20.04.2020).
17. Онлайн путівник по Україні: веб-сайт. URL: <https://ua.igotoworld.com/> (дата звернення: 15.04.2020).
18. hQuery.php: веб-сайт. URL: <https://github.com/duzun/hQuery.php> (дата звернення: 20.05.2020).
19. Class hQuery: веб-сайт. URL: <http://hquery.duzun.me/docs/class-hQuery.html> (дата звернення: 20.04.2020).
20. Bootstrap 3 - Navbar (горизонтальное меню): веб-сайт. URL: <https://itchief.ru/lessons/bootstrap-3/38-bootstrap-3-navigation-menu> (дата звернення: 15.04.2020).
21. Ларавел по русски: веб-сайт. URL: <https://laravel.ru/> (дата звернення: 18.04.2020).
22. Матрица управления рисками: веб-сайт. URL: <https://econ.wikireading.ru/67771> (дата звернення: 28.03.2020).
23. Жуков Н.С., Папшев В.А. Алгоритм построения городских туристических переходных маршрутов. Труды Международного симпозиума «Надёжность и качество». 2016. Т2. С. 204-207.

ДОДАТОК А

ТЕХНІЧНЕ ЗАВДАННЯ на розробку Web-додатку підтримки вибору туристичного маршруту пам'ятками Сумщини

Суми 2020

A.1 Призначення й мета створення Web-додатку

A.1.1 Призначення Web-додатку

Web-додаток призначений надавати підтримку вибору туристичного маршруту пам'ятками Сумщини.

A.1.2 Мета створення Web-додатку

Ознайомлення користувачів з туристичними пам'ятками міста та надання можливості самостійно скласти унікальний маршрут подорожі пам'ятками Сумщини.

A.1.3 Цільова аудиторія

Цільовою аудиторією Web-додатку можуть бути туристи, мешканці міста або інші зацікавлені відвідувачі додатку.

A.2 Вимоги до Web-додатку в цілому

A.2.1 Вимоги до структури й функціонування Web-додатку

Web-додаток повинен мати інтуїтивно зрозумілий інтерфейс для користувачів з різним рівнем навиків роботи з комп'ютером. Доступ до Web-додатку повинен здійснюватися через мережу Інтернет. Додаток повинен мати коректне відображення у різних браузерах та на різних пристроях. Сторінки сайту повинні бути стилістично взаємопов'язані між собою та мати чіткі визначені функції.

A.2.2 Вимоги до персоналу

Для підтримки сайту й експлуатації веб-інтерфейсу адміністративної панелі від персоналу не повинно вимагатися спеціальних технічних навичок, знання технологій або програмних продуктів, за винятком загальних навичок роботи з персональним комп'ютером і стандартним веб-браузером (наприклад, MS Internet Explorer 7.0 або вище).

A.2.3 Вимоги до розмежування доступу

Інформація, яка розташовується на сайті, є загальнодоступною.

Користувачів сайту можна розділити на 2 групи відповідно до прав доступу:

- відвідувачі;
- адміністратор .

Відвідувачі мають доступ тільки до загальнодоступної частини сайту, де зможуть переглядати місця відпочинку, а також створювати власний туристичний маршрут.

Доступ до адміністративної частини мають користувачі із правами адміністратора.

Адміністратор може додавати, редагувати чи видаляти матеріали розділів та переглядати інформацію в закритих полях.

Доступ до адміністративної частини повинен здійснюватися з використанням унікального логіна й пароля.

А.3 Основні вимоги

А.3.1 Структура сайту

Сайт повинен складатися з наступних розділів:

- сторінка Головна – містить довідкову інформація про сайт;
- сторінка Культурні пам'ятки – основна частина контенту, представлена у вигляді підрозділів з описами культурних пам'яток Сумщини та місць відпочинку;
- сторінка створити маршрут – розділ, де є можливість створити маршрут на основі власних вподобань користувача;
- сторінка Реєстрація/Вхід – реєстрація на сайті або вхід в особистий кабінет.

А.3.2 Навігація

Інтерфейс Web-додатку повинен бути інтуїтивно зрозумілим та мати швидкий і логічний перехід до розділів і сторінок. На сторінках додатку необхідно встановити єдиний спосіб переходу від сторінки до сторінки і від розділу до розділу, а також повернення на домашню сторінку і доступу до основних посилань.

Навігаційні елементи повинні забезпечувати однозначне розуміння користувачем їх змісту: посилання на сторінки повинні бути мати заголовки, умовні позначки відповідати загальноприйнятим. Графічні елементи навігації повинні бути мати альтернативний підпис.

Система повинна забезпечувати навігацію по всіх доступних користувачеві ресурсам і відображати відповідну інформацію. Для навігації повинна

використовуватися система контент-меню. Меню повинне являти собою текстовий блок (список гіперпосилань) у верхній частині сторінки.

А.3.3 Наповнення сайту (контент)

Інформація з кожного розділу додатку повинна зберігатися у таблицях бази даних на сервері.

Модифікація вмісту розділів повинна здійснюватися за допомогою адміністративної панелі, яка без застосування спеціальних навичок програмування повинна передбачати можливість редагування інформаційного вмісту сторінок сайту. Наповнення інформацією повинне проводитися з використанням шаблонів сторінок сайту.

А.3.4 Вимоги до функціональних можливостей

Адміністративна панель сайту повинна надавати можливість додавання, редагування й видалення вмісту сторінок.

А.3.5 Функціональні можливості розділів

На кожній сторінці сайту будуть представлені наступні елементи:

- логотип;
- головне меню;
- авторизація;
- інформація про розробника.

А.3.6 Загальні вимоги

Дизайн сторінок Web-додатку повинен бути витриманим у одному стилі, головні елементи повинні бути помітними, доступ до меню повинен бути на кожній сторінці.

Розташування елементів на сторінці сайту розділу культурних пам'яток схематично показано на рисунку А.1.

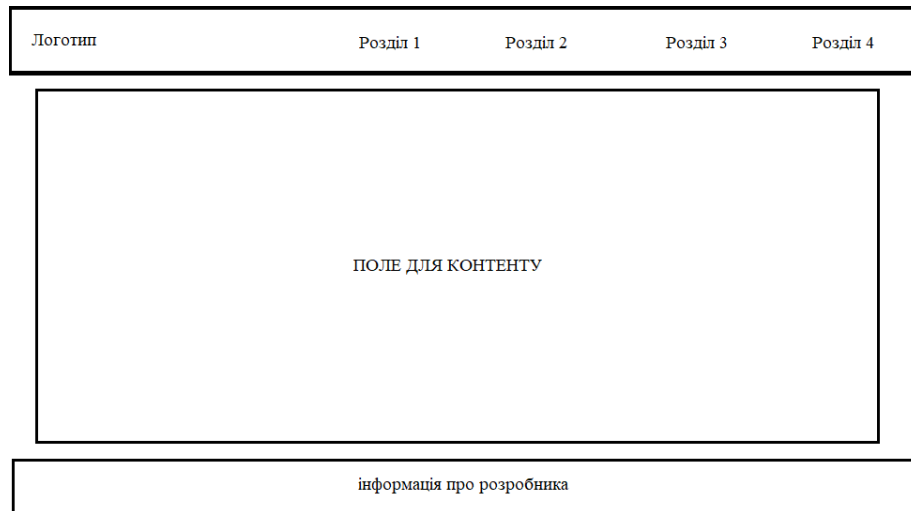


Рисунок А.1 – Типова сторінка

А.3.7 Типові навігаційні й інформаційні елементи

- шапка сайту;
- основне меню;
- додаткове меню;
- основне поле контенту.

А.3.7.1 Шапка сайту

Шапка сайту повинна містити логотип з назвою сайту, головне навігаційне меню, перехід до сторінки авторизації. Логотип є посиланням на головну сторінку сайту.

А.3.7.2 Меню

Головне меню розміщується у верхній частині вікна та наявне на кожній сторінці додатку. Воно містить посилання на головну сторінку, на сторінку з переглядом культурних пам'яток, на сторінку створення маршруту та на сторінку авторизації. Додаткове меню являє собою посилання з підрозділами – видами культурних пам'яток. Воно розташовується у верхній частині вікна у розділі культурних пам'яток.

А.3.7.3 Основне поле контенту

Основне поле контенту повинне розташовуватися в центрі сторінки. У цьому полі відображається основний зміст обраного розділу. Стильове оформлення матеріалів та їх елементів (посилань, заголовків, основного тексту, зображень, форм, таблиць тощо) повинне бути єдиним для всього Web-сайту.

А.3.8 Вимоги до програмного забезпечення

Програмне забезпечення клієнтської частини повинне задовольняти наступним вимогам:

– веб-браузер з підтримкою JavaScript, Flash, cookies: Internet Explorer 7.0 і вище або Firefox 7.5 і вище, або Opera 9.5 і вище, або Safari 6.1 і вище, або Chrome 7 і вище.

ДОДАТОК Б

ПЛАНУВАННЯ РОБІТ

Деталізація мети методом SMART. Сутність деталізації мети проекту за допомогою SMART-методу впливає з розшифровки термінів, які формують його назву: конкретна (Specific), вимірювана (Measurable), досяжна (Achievable), реалістична (Relevant), обмежена у часі (Time-framed).

S – конкретність, специфічність. Створити додаток підтримки вибору туристичного маршруту пам'ятками Сумщини.

M – вимірюваність. Для вільного доступу до Web-додатку, на розробку якого витрачено мінімум ресурсів, всім користувачам необхідно розмістити його на хостингу.

A – узгодженість. Для того, щоб підвищити потік відвідувачів міста, полегшити пошук усієї потрібної інформації про місця відвідувань та створення власного маршруту подорожі потрібно розробити додаток, який дасть змогу вирішити ці задачі одночасно.

R – реалістичність, релевантність. Для виконання проекту необхідні знання HTML, CSS, мови програмування JavaScript, PHP, MySQL та навичок написання документації. Враховуючи доступні ресурсні можливості та обмеження мета є такою, яку можливо досягти.

T – обмеженість в часі. Ціль має часові обмеження. Терміни досягнення мети проекту визначаються за домовленістю замовником та виконавцем.

Опис функціонування продукту проекту. На основі встановленої мети проекту було встановлено рекомендації до функціоналу додатку які мають найбільше значення:

- збір інформації про пам'ятки міста;
- складення маршруту подорожі;

- збереження маршруту;
- кросбраузерність;
- користування додатком маючи доступ до мережі Інтернет.

Планування змісту структури робіт ІТ—проекту. WBS – це графічне подання згрупованих елементів проекту у вигляді пакета робіт, які ієрархічно пов’язані з продуктом проекту. На верхньому першому рівні WBS фіксується продукт проекту. Він повинен відповідати продукту проекту. Наступний II рівень відповідає діям або основним заходам для досягнення продукту проекту.

Потім триває розбивка цих дій доти, поки не відбувається виконання дій елементарних робіт. Елементарні роботи – це роботи, які мають один чіткий результат, який використовується при прийнятті цієї роботи; на які призначений один конкретний відповідальний; на неї можна обчислити витрати праці і тривалість виконання. Зазвичай декомпозиція завершується тоді, коли для розкриття змісту потрібні вузькі фахівці, що знають технологічні особливості їх виконання.

На рисунках Б.1-Б.3 представлена WBS-структура Web-додатку підтримки вибору туристичного маршруту пам’ятками Сумщини.

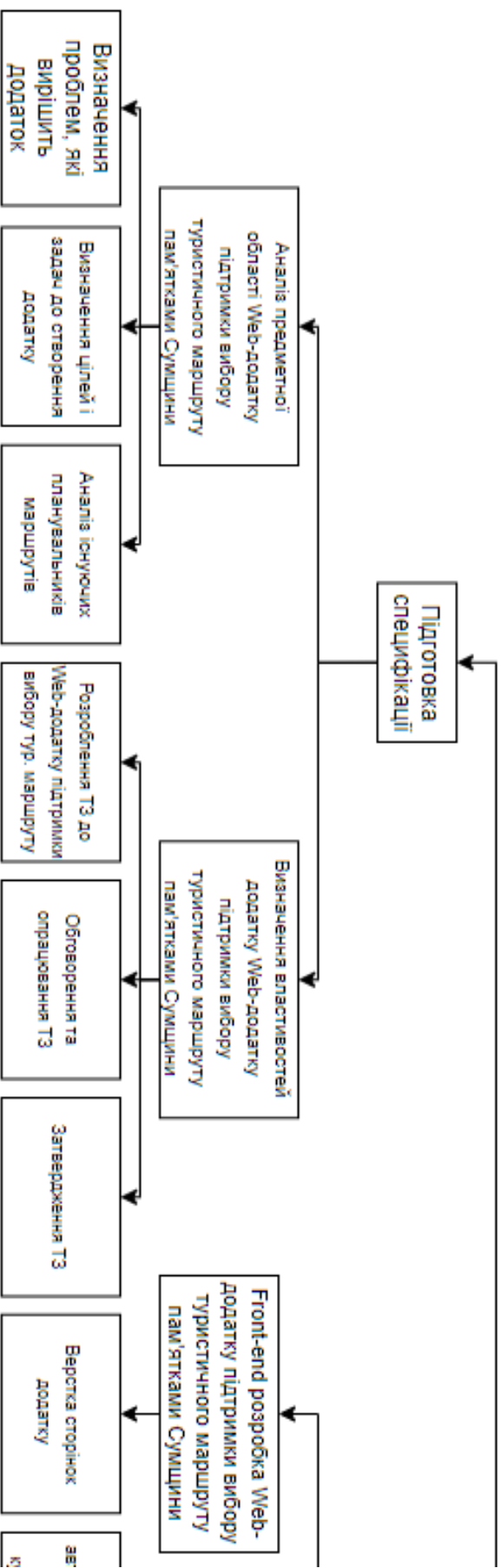


Рисунок Б.1 – Початок WBS-структури Web-додатку

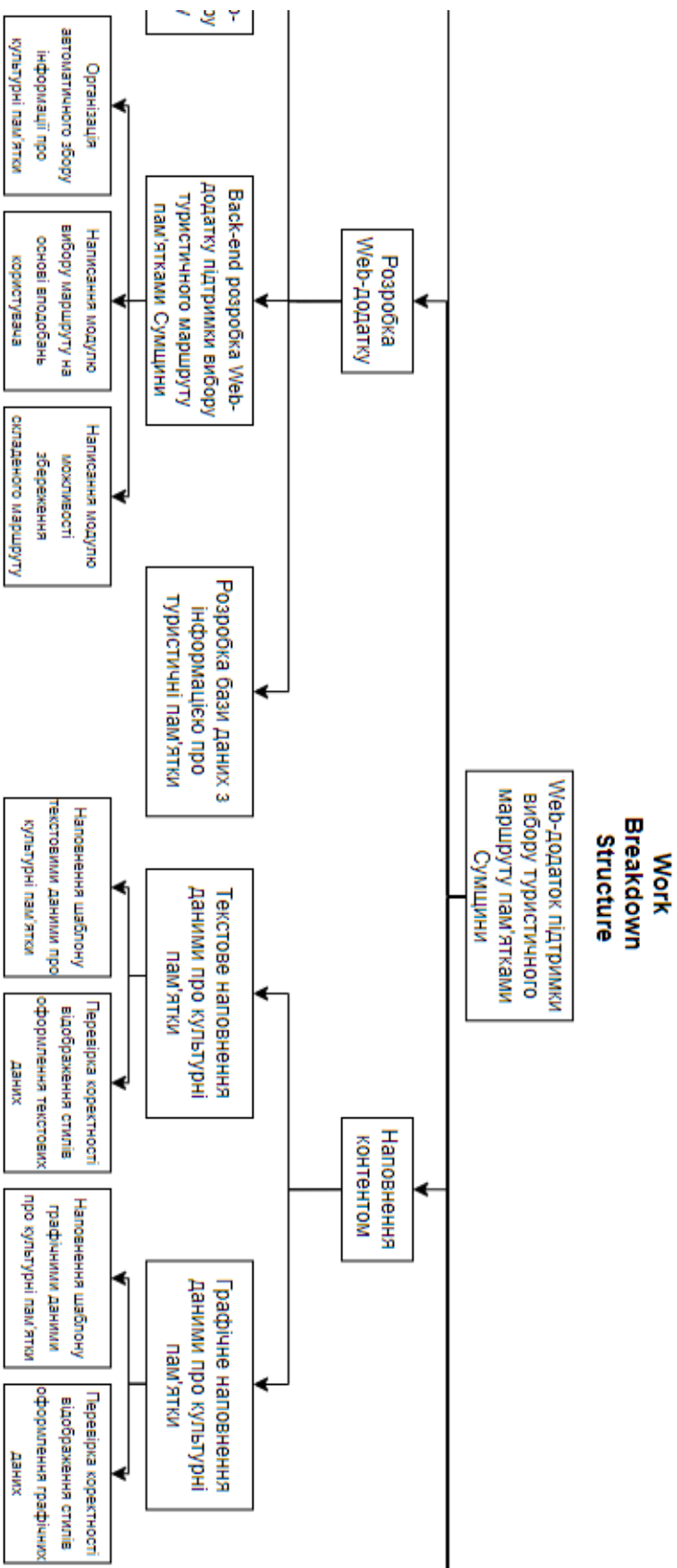


Рисунок Б.2 – Продовження WBS-структури Web-додатку

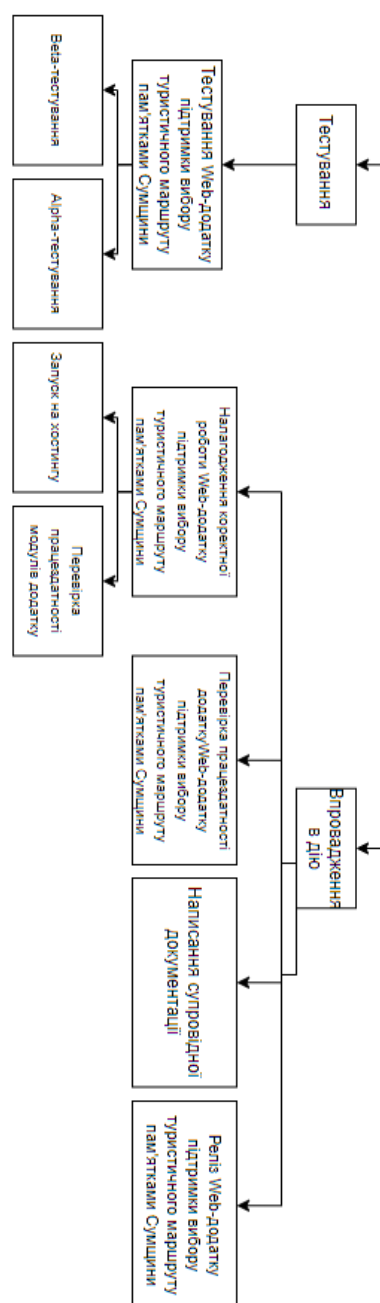


Рисунок Б.3 – Закінчення WBS-структури Web-додатку

Планування структури організації. Після побудови WBS розробляють організаційну структуру виконавців. OBS-структура проекту – організаційна структура виконавців (організацій) проекту. Визначається за переліком пакетів робіт нижнього рівня кожної гілки WBS-структури.

Організаційна структура представляє собою графічне відображення учасників проекту та їх відповідальних осіб, які задіяні в реалізації проекту.

На рисунках Б.4-Б.5 наведено організаційну структуру планування проекту.

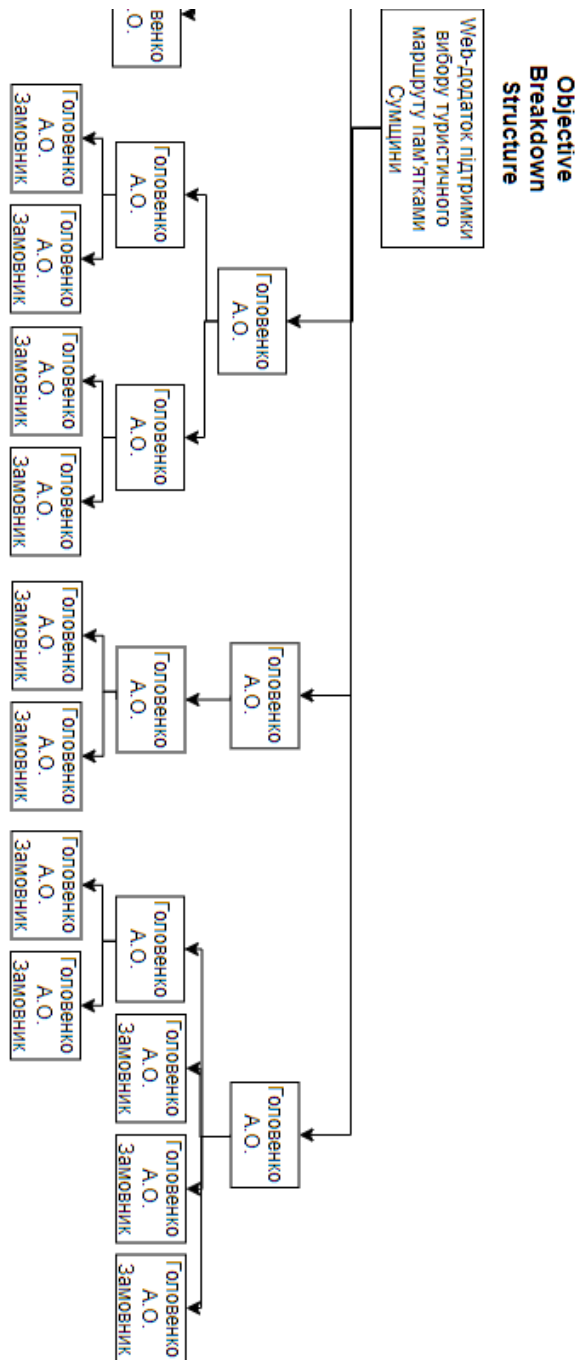


Рисунок Б.5 – Кінець OBS-структури Web-додатку

Побудова матриці відповідальності. На підставі OBS та WBS структур будують матрицю відповідальності проекту. Матриця відповідальності закріплює за кожною елементарною роботою виконавця. Вона забезпечує опис і узгодження структури відповідальності за виконання пакетів робіт, а також являє собою форму опису розподілу відповідальності за реалізацію робіт по проекту, із зазначенням ролі кожного з підрозділів в їх виконанні.

Для кожного з виконавців була визначена його роль:

– відповідальний (В) – повністю відповідає за виконання задачі та має право приймати рішення щодо способу її реалізації;

– консультант (К) – наглядає за ходом виконання завдання і висловлює свої міркування стосовно способу та якості реалізації. Несе відповідальність, якщо не помітить явного недоліку.

– спостерігач (С) – те ж саме що і консультант, але відповідальності не несе.

У таблиці Б.1 показано матрицю відповідальності проекту.

Таблиця Б.1 – Матриця відповідальності проекту

№	WBS\OBS	Головенко А.О.	Кузнецов Е.Г.	Рецензент
1	Аналіз предметної області	В	К	
2	Розроблення ТЗ	В	К	
3	Обговорення та опрацювання ТЗ	В	К	
4	Затвердження ТЗ	В	К	
5	Front-end розробка	В	С	
6	Back-end розробка	В	С	
7	Розробка БД	В	С	
8	Текстове наповнення	В	С	
9	Графічне наповнення	В	С	
10	Тестування розробки	В	С	С
11	Запуск на хостингу	В	С	
12	Перевірка працездатності модулів	В	С	С
13	Перевірка працездатності додатку	В	С	С
14	Написання супровідної документації	В	К	

Розробка PDM мережі. Для кожного пакета робіт розробляють часткові мережеві моделі, в яких встановлюють логічні взаємозв'язки між всіма роботами, які необхідно виконувати для отримання запланованого продукту з пакету робіт. Мережеві моделі дозволяють визначити тривалість виконання пакету робіт. Мережеві моделі рекомендовано будувати з використанням програмного інструментарію (MS Project, Spider Project, Project Expert, Open Plan, Primavera тощо).

WBS- структура містить перелік всіх пакетів робіт, які необхідно виконувати для отримання продукту проекту. Але вона не показує, як ці пакети взаємопов'язані ці пакети у часі, яка тривалість виконання кожного з них та загальна тривалість проекту для отримання його продукту.

Мережеві моделі проекту встановлюють логічний взаємозв'язок між всіма роботами, які необхідно виконати для отримання конкретного продукту PDM-мережі складаються з двох типів елементів: робіт, які розташовані у вузлах, та стрілок, які вказують логічні взаємозв'язки між роботами проекту.

Побудова календарного графіку. Для того щоб мати реальне уявлення про тривалість виконання робіт з урахуванням обмеженості у використанні ресурсів, на підставі часткової мережевої моделі, а також, проекту в цілому з урахуванням вихідних та святкових днів, будують календарний графік робіт. Він є реальним розподілом робіт з пакету за календарними датами, тобто своєрідним розкладом виконання робіт.

Календарний графік з виконання роботи представлений на рисунку Б.6.

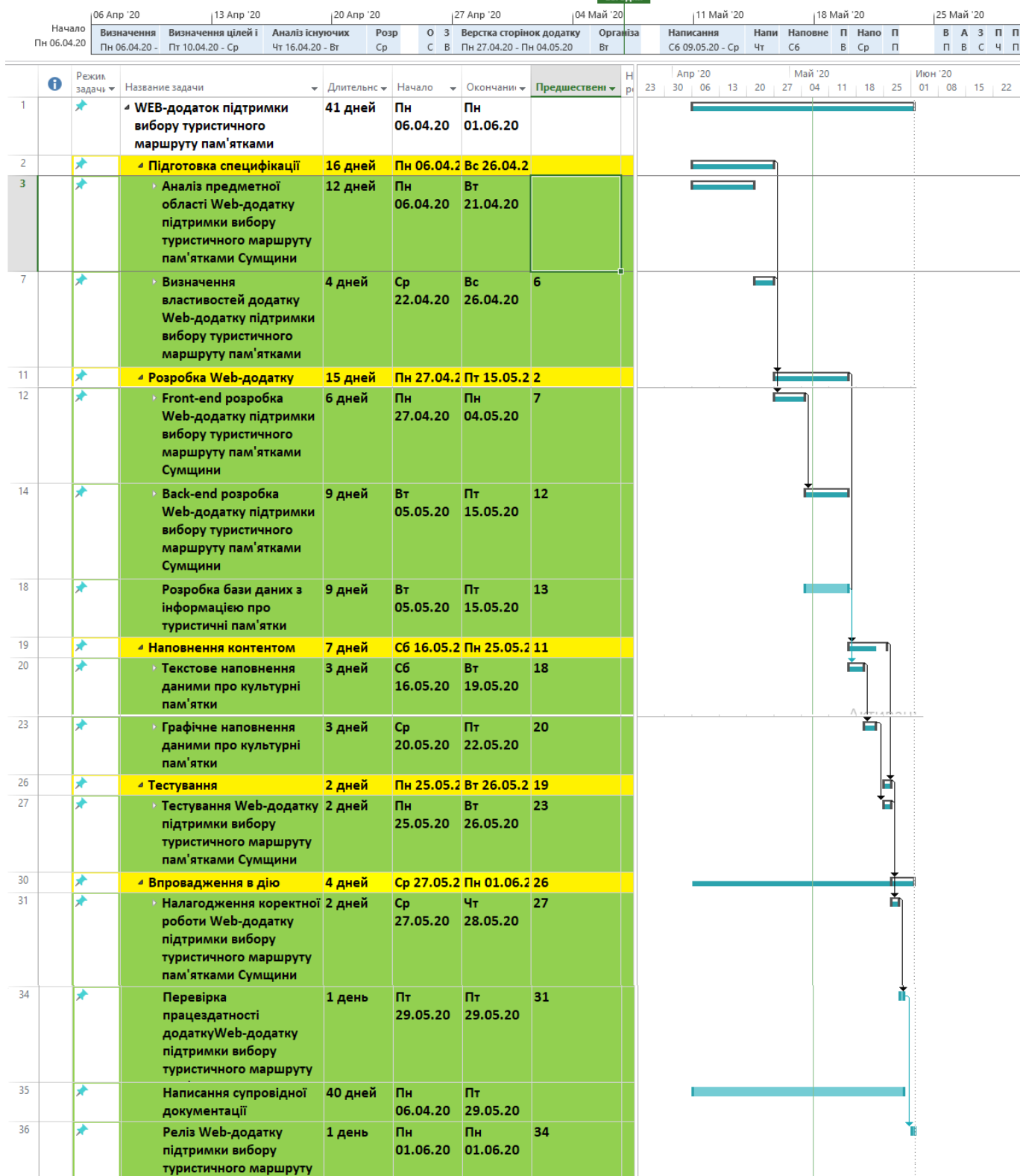


Рисунок Б.6 – Календарний графік

Управління ризиками проекту. Ризик – це імовірна подія, яка у випадку своєї появи негативно або позитивно вплине на проект.

Процес управління ризиками включає наступні етапи:

- ідентифікація;
- процес оцінювання ризиків, який включає в себе якісний, кількісний аналіз;
- заходи реагування на ризики;
- моніторинг заходів і ризиків.

Ризики проекту представлені у таблиці Б.2.

Таблиця Б.2 – Ризики проекту

Ризик	Назва ризику	Опис ризику
1	Низька кваліфікація	Високі вимоги до кваліфікації
2	Зміна вимог	Зміна об'ємів, вартості та терміну робіт замовником
3	Хостинг	Проблеми з роботою хостингу
4	Фінансування	Недотримання термінів і обсягів фінансування
5	Підтримка проекту	Недостатня підтримка проекту зі сторони замовника

Таблиця Б.3 – Матриця ймовірності виникнення ризиків

Ризик	Вірогідність виникнення	Ступінь впливу	Значення ризику	Боротьба з ризиком (час/гроші/час & гроші)
1	2	3	6	час
2	3	5	15	час & гроші
3	2	4	8	час & гроші
4	2	5	10	час & гроші
5	2	2	4	час

Матриця впливу ризиків відображає значимість події на одній осі та ймовірність її виникнення на іншій. Проміжним підсумком цієї роботи має стати план реагування [22]. У таблиці виділяють зеленим кольором прийнятні ризики, жовтим кольором – виправданні ризики, а червоним – недопустимі ризики. Матриця впливу представлена у таблиці Б.4.

Таблиця Б.4 – Матриця впливу

Вірогідність виникнення	Матриця впливу				
5					
4					
3					2
2		5	1	3	4
1					
Ступінь впливу	1	2	3	4	5

Реагування на ризики представлені у таблиці Б.5.

Таблиця Б.5 – Реагування на ризики проекту

Ризик	Назва ризику	Реакція на ризик
1	Низька кваліфікація	Консультації зі спеціалістами предметної області. Проходження курсів підвищення кваліфікації
2	Зміна вимог	Коригування часу на виконання проекту
3	Хостинг	Збільшення часу, зміна хостингу
4	Фінансування	Припинення розробки до відновлення повного фінансування
5	Підтримка проекту	Виділення відповідальної особи з боку замовника, контролюючого терміни і якість робіт за проектом

ДОДАТОК В

ЛІСТИНГ ЗБЕРЕЖЕННЯ ПАМ'ЯТОК ДО БАЗИ ДАНИХ

```

public function store(Request $request){
    $array = [];
    foreach ($request->input('images') as $key => $value) {
        $array[$key] = (explode(",", $value));}
    $title = collect($request->input('title'));
    $content = $request->input('content');
    $img = $request->input('img');
    $location = $request->input('location');
    $category = $request->input('category');
    $places = $title->zip($content, $img, $location, $array,$category);
    foreach ($places as $key => $t) {
        if(!Place::where('title', $t)->exists()) {
            $img_thumb = \Image::make(preg_replace('/370x246/', '800x600', $t[2]))->fit(220, 220);
            $place = Place::create(['title'=> $t[0],'slug' => str2url($t[0]),'content' => $t[1],
            'post_img' => 'img-'.str2url($t[0]).'-full.jpg','post_thumbnail' => 'thumb-'.str2url($t[0]).'-
220x220.jpg',
            'location' => $t[3],'cat_id' => $t[5]]);
            if($t[4][0] !== "") {
                foreach ($t[4] as $k => $v) {
                    $image = Image::create([
                        'img' => 'image-'.str2url($t[0]).'-'. $k.'-full.jpg',
                        'gallery_thumb' => 'thumb-image-'.str2url($t[0]).'-'. $k.'-220x220.jpg',
                        'place_id' => $place->id]);
                    $img_gallery_thumb = \Image::make($v)->fit(220, 220);
                    $post_gallery_img = \Image::make(preg_replace('/370x246/', '800x600', $v))-
>save(public_path().'/img/gallery/image-'.str2url($t[0]).'-'. $k.'-full.jpg', 100);
                    if(isset($img_gallery_thumb) { $img_gallery_thumb-
>save(public_path().'/img/gallery/thumb-image-'.str2url($t[0]).'-'. $k.'-220x220.jpg', 80);
                    }}}
            $post_img = \Image::make(preg_replace('/370x246/', '800x600', $t[2]))-
>save(public_path().'/img/img-'.str2url($t[0]).'-full.jpg', 100);
            $img_thumb->save(public_path().'/img/thumb-'.str2url($t[0]).'-220x220.jpg', 80);}
            return redirect('/admin'); }

```

ДОДАТОК І

ЛІСТИНГ ШАБЛОНУ СТОРІНКИ ДЕТАЛЬНОЇ ІНФОРМАЦІЇ ПРО ПАМ'ЯТКУ

```

@extends('front-end.layout')
@section('content')
  <div id="main_part_inner-single">
    <div id="main_part_inner_in-single">
      <div class="container">
        <div class="col-md-8">
          <h2>{{ $place->title }}</h2>
        </div>
        <div class="col-md-4">
          <div class="button_main">
            @if(!$user)
              <form action="/login" method="get" class="formit">
                <input type="submit" class="button_submit" value="Додати до обраних">
              </form>
            @else
              @if(!$exists)
                <form action="/account/routes/create" method="post" class="formit">
                  {{ csrf_field() }}
                  <input type="hidden" name="place_id" value="{{ $place->id }}">
                  <input type="submit" class="button_submit" value="Додати до обраних">
                </form>
              @else
                <form action="/account/routes/create/delete" method="post" class="formit">
                  {{ csrf_field() }}
                  <input type="hidden" name="place_id" value="{{ $place->id }}">
                  <input type="submit" class="button_submit" value="Видалити з обраних">
                </form>
              @endif
            @endif
          </div>
        </div>
      </div>
    </div>
  </div>
  <div id="content_inner" class="text-center">
    
    <div class="cara"></div>
    <h3>Інформація</h3>
    <div class="contactinfo">
      {!! $place->content !!}
    </div>
    <div class="demo-gallery text-center">
      <ul id="lightgallery" class="list-unstyled row">
        @foreach($place->images as $image)

```



```

    <li class="col-xs-6 col-sm-4 col-md-3" data-src="\img\gallery\{{$image->img}}">
        <a href="">
            
        </a>
    </li>
@endforeach
</ul>
</div>
</div>
<hr class="cleanit">
<div class="cara"></div>
<h3>Ha kapri</h3>
<div class="mapit way_points">
    <p id="location">{{$place->location}}</p>
<div id="map-layer"></div>
<script>
var map;
var service;
var infowindow;
function initMap() {
    var sydney = new google.maps.LatLng(50.906977, 34.7992);
    infowindow = new google.maps.InfoWindow();
    map = new google.maps.Map(
        document.getElementById('map-layer'), {center: sydney, zoom: 17});
    var request = {
        query: $('#location').html(),
        fields: ['name', 'geometry'],
    };
    service = new google.maps.places.PlacesService(map);
    service.findPlaceFromQuery(request, function(results, status) {
        if (status === google.maps.places.PlacesServiceStatus.OK) {
            for (var i = 0; i < results.length; i++) {
                createMarker(results[i]);
            }
            map.setCenter(results[0].geometry.location);
        }
    });
    function createMarker(place) {
        var marker = new google.maps.Marker({
            map: map,
            position: place.geometry.location});
        google.maps.event.addListener(marker, 'click', function() {
            infowindow.setContent(place.name);
            infowindow.open(map, this);
        });
    }
</script>
<script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDIzmXaMoW9gWqIsT3RMSDsJyixudAl4bI
&libraries=places&callback=initMap" async defer></script>
</iframe>
</div>
</div>
@endsection

```

ДОДАТОК Д

ЛІСТИНГ ВІДОБРАЖЕННЯ СТВОРЕННЯ МАРШРУТУ

```

var map;
var waypoints
function initMap() {
    var mapLayer = document.getElementById("map-layer");
    var centerCoordinates = new google.maps.LatLng(50.906977, 34.7992);
    var defaultOptions = { center: centerCoordinates, zoom: 10 }
    map = new google.maps.Map(mapLayer, defaultOptions);
    var directionsService = new google.maps.DirectionsService;
    var directionsDisplay = new google.maps.DirectionsRenderer;
    directionsDisplay.setMap(map);

    $("#drawPath").on("click",function() {
        waypoints = Array();
        $('.way_points:checked').each(function() {
            waypoints.push({
                location: $(this).next('.location').val(),
                stopover: true
            });
        });
        var locationCount = waypoints.length;
        if(locationCount > 0) {
            var start = waypoints[0].location;
            var end = waypoints[locationCount-1].location;
            drawPath(directionsService, directionsDisplay,start,end);
        }
    });
}

function drawPath(directionsService, directionsDisplay,start,end) {
    directionsService.route({
        origin: start,
        destination: end,
        waypoints: waypoints,
        optimizeWaypoints: true,
        travelMode: "WALKING"
    }, function(response, status) {
        if (status === 'OK') {
            directionsDisplay.setDirections(response);
        } else {
            window.alert('Problem in showing direction due to ' + status);
        }
    });
}

```

ДОДАТОК Е

ЛІСТИНГ ВІДОБРАЖЕННЯ ЗБЕРЕЖЕНОГО МАРШРУТУ

```

var map;
var waypoints
function initMap() {
    var mapLayer = document.getElementById("map-layer");
    var centerCoordinates = new google.maps.LatLng(50.906977, 34.7992);
    var defaultOptions = { center: centerCoordinates, zoom: 10 }
    map = new google.maps.Map(mapLayer, defaultOptions);
    var directionsService = new google.maps.DirectionsService;
    var directionsDisplay = new google.maps.DirectionsRenderer;
    directionsDisplay.setMap(map);
    waypoints = Array();
    $('input.place').each(function() {
        waypoints.push({
            location: $(this).val(),
            stopover: true
        });
    });
    var locationCount = waypoints.length;
    if(locationCount > 0) {
        var start = waypoints[0].location;
        var end = waypoints[locationCount-1].location;
        drawPath(directionsService, directionsDisplay,start,end);
    }
}
function drawPath(directionsService, directionsDisplay,start,end) {
    directionsService.route({
        origin: start,
        destination: end,
        waypoints: waypoints,
        optimizeWaypoints: true,
        travelMode: "WALKING"
    }, function(response, status) {
        if (status === 'OK') {
            directionsDisplay.setDirections(response);
        } else {
            window.alert('Problem in showing direction due to ' + status);
        }
    });
}

```

ДОДАТОК Ж

ОПРИЛЮДНЕННЯ РЕЗУЛЬТАТІВ РОБОТИ

*СЕКЦІЯ 2: Інформаційні
технології проектування*

ІМА :: 2020

Web-Application to support the selection of a tourist route to the attractions of Sumshchina

Golovenko A.O., *Student*; Kuznetsov E.G., *Senior Lecturer*
Sumy State University, Sumy, Ukraine

Information technologies are firmly established in human life and make it easier every day, because it is possible to learn, work, receive different consultations of specialists, etc., without leaving home. This saves time considerably because there is a need for more services to be available through electronic means as much as possible. This also applies to travel and tourism.

The purpose of the work is to develop a web-application to support the selection of a tourist route according to the honourable places of Sumshchina. To achieve this goal, the following objectives have been achieved:

- review of existing analogues and identification of their main shortcomings;
- the peculiarities of the organization of tourist routes are investigated;
- the structure of the IT organization of tourist travel has been developed and its software implementation in the form of a web-application has been implemented.

A review of existing software products showed that applications lack the functionality to plan a route based on user tastes. Also, some applications do not provide for saving a compiled route for use without an Internet connection. Therefore, it was decided to create a web application that has an intuitive interface and allows the user from any device to obtain information about the city and to make a travel route on his own depending on his tastes.

In the development of the web application, a handwritten solution was chosen by HTML, CSS, PHP5 and Javascript, which allowed to create a unique application both in terms of design and functionality.

As a result, a web-oriented application was developed, which allows the user from any device to get information about the city: its outstanding attractions, places of rest, catering facilities. In addition, this application allows the user to make a travel route on his own depending on his tastes and then store it in the device memory.