

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
Навчально-науковий інститут
«Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці**

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»
галузь знань 12 «Інформаційні технології»
спеціальність 122 «Комп'ютерні науки»

Форма навчання: денна

КВАЛІФІКАЦІЙНА БАКАЛАВРСЬКА РОБОТА

на тему: «Проектування та розроблення інформаційної системи «Подорож містами України»»

здобувача Герасименка Максима Ігоровича

(ПІБ)

_____ (Підпис)

Науковий керівник:

к.е.н., професор

_____ Сендзюк М.А.

**Робота допущена до захисту перед
екзаменаційною комісією з атестації
здобувачів вищої освіти**

завідувач кафедри:

к.е.н., доцент

_____ Тішков Б.О.

Київ 2025

Міністерство освіти і науки України
Київський національний економічний університет імені Вадима Гетьмана
Навчально-науковий інститут «Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»

галузь знань 12 «Інформаційні технології»

спеціальність 122 «Комп'ютерні науки»

ПОГОДЖЕНО:

Керівник проектної групи(гарант)
освітньо-професійної програми

_____Помазун О.М.

“ _____ ” _____ 2025 р.

ЗАТВЕРДЖУЮ:

Завідувач кафедри

_____Тішков Б.О.

“ _____ ” _____ 2025 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

здобувачу вищої освіти *Герасименку Максиму Ігоровичу*

очної (денної) форми навчання

на підготовку кваліфікаційної бакалаврської роботи
на тему: «Проектування та розроблення інформаційної системи «Подорож
містами України»»

Тему затверджено наказом ректора Університету від « 7 » березня 2025 р.
№ 466-ст.

Кваліфікаційна бакалаврська робота виконується на матеріалах
отриманих під час навчання, навчальних посібників, матеріалів з мережі Інтернет,
а також зібраного фактичного матеріалу з обраної теми дослідження під час
виконання курсових робіт

План кваліфікаційної бакалаврської роботи

Розділ I Характеристика та аналіз предметної галузі

Розділ II Розробка вимог і моделювання інформаційної системи

Розділ III Проектування та реалізація компонентів системи

Об'єкт дослідження Це процес або явище, що обрані для дослідження, зокрема
методика і технологія організації подорожування містами України

Предмет дослідження Інформаційні процеси та комп'ютерні технології з
адміністрування подорожами містами України

Мета кваліфікаційної бакалаврської роботи Провести дослідження і виконати проектні рішення з розроблення інформаційної системи адміністрування подорожами містами України

Конкретні завдання, які здобувач повинен виконати для досягнення поставленої мети:

У розділі I необхідно дати характеристику галузі управління екскурсіями, визначити структуру об'єкта управління та функції її складових, особливу увагу слід приділити факторам, що впливають на організацію інформаційної системи, доцільно навести схему організаційної структури. Аналіз літературних джерел проводиться з метою виявлення прогресивних методик автоматизації адміністрування подорожами містами України.

У розділі II провести аналіз і специфікацію вимог до інформаційної системи «Подорож містами України». Дотримуючись вимог стандартів виконати постановку та розробити алгоритм розв'язання задач з автоматизації функцій адміністрування подорожами містами України. Розробити інформаційну модель підтримки управління подорожами містами України у відповідності до вимог, викладених в методичних вказівках кафедри. Діаграми повинні бути читабельними, більшим шрифтом.

У розділі III виконати проектування таких компонентів системи: інформаційного, технічного та програмного забезпечення у відповідності з методичними рекомендаціями кафедри, вказати можливості і доцільності впровадження.

**Завдання підготував
науковий керівник**

_____ **Сендзюк Мирон Ананійович**
«_10_» березня 2025 р.

**Завдання одержав
здобувач**

_____ **Герасименко Максим Ігорович**
«_10_» березня 2025 р..

Відгук
про кваліфікаційну бакалаврську роботу
здобувача навчально-наукового інституту
«Інститут інформаційних технологій в економіці»
освітньо-професійної програми
«Комп'ютерні науки»

Герасименка Максима Ігоровича

на тему
«Проектування та розроблення інформаційної системи «Подорож містами України»»

1. Актуальність теми пояснюється необхідністю надання сучасного та інтуїтивно зрозумілого інструменту для планування подорожей громадян України, ознайомлення з культурними, історичними та природними скарбами. У час, коли внутрішній туризм є одним із безпечніших та доступніших напрямків відпочинку, підтримка таких ініціатив через цифрові рішення сприяє як культурному, так і економічному розвитку країни.
2. Позитивні риси кваліфікаційної бакалаврської роботи: детальний аналіз предметної галузі, обґрунтований вибір структури інформаційної системи, сформовані бізнес-вимоги для інформатизації., визначені функціональні та нефункціональні вимоги
3. Наявність самостійних розробок автора автор самостійно розробив концепцію інформатизації оригінальної, нетрадиційної предметної області, обґрунтував структуру інформаційної системи, розробив інформаційне забезпечення та виконав програмну реалізацію автоматизованих функцій .
4. Цінність теоретичних висновків полягає в узагальненні сучасних підходів до інформатизації галузі туризму та розробленні конкретних рекомендацій для цифровізації даної галузі. Практична цінність роботи підтверджується наведеними скріншотами основних функцій.
5. Наявність недоліків: недоліків у роботі не виявлено.
6. Загальна оцінка кваліфікаційної бакалаврської роботи та її допущення до захисту перед ЕК Бакалаврська робота Герасименка М.І. є актуальною, виконана на високому рівні, відповідає вимогам до кваліфікаційних бакалаврських робіт за спеціальністю «Комп'ютерні науки» та може бути допущена до захисту перед ДЕК з попередньою високою оцінкою.

Науковий керівник

(підпис)

професор, к.е.н. Сендзюк М.А.

“15” червня 2025 р.

Рецензія

на кваліфікаційну бакалаврську роботу
здобувача вищої освіти

Герасименка Максима Ігоровича

Тема: Проектування та розроблення інформаційної системи «Подорож містами України»

Актуальність теми кваліфікаційної бакалаврської роботи і доцільність її розроблення Обрана тема є надзвичайно актуальною в умовах зростаючого попиту на внутрішній туризм та діджиталізацію туристичних послуг. Розроблення інформаційної системи для підтримки подорожей містами України відповідає сучасним запитам користувачів щодо швидкого доступу до достовірної інформації. Такий підхід сприяє розвитку туристичної галузі, особливо в післякризовий період, коли популяризація внутрішнього туризму набуває особливого значення.

Якість проведеного дослідження Кваліфікаційна робота демонструє системність та глибину дослідження: проведено всебічний аналіз предметної області, сформульовано чіткі функціональні та нефункціональні вимоги до системи, обґрунтовано вибір технологій та інструментів розробки. Значну увагу приділено побудові інформаційної моделі, технічному і програмному забезпеченню, а також візуалізації структури системи за допомогою UML-діаграм.

Позитивні риси кваліфікаційної бакалаврської роботи Серед переваг слід відзначити комплексність підходу до розробки системи, продуману архітектуру, інтуїтивно зрозумілий інтерфейс, використання сучасного технологічного стеку. Особливої уваги заслуговує реалізація моделі бази даних з урахуванням категорій, а також сценаріїв взаємодії з системою. Робота містить приклади реалізації запитів до бази даних і взаємодії між клієнтською частиною та сервером.

Зауваження До незначних зауважень можна віднести обмежену кількість сценаріїв перевірки функціональності в реальному середовищі, а також відсутність повноцінного аналізу користувацьких потреб через опитування чи інтерв'ю. Також, враховуючи перспективність теми, варто окреслити план інтеграції з зовнішніми API туристичних платформ.

Практична значимість висновків і рекомендацій Запропонована система має чіткий прикладний характер і здатна полегшити організацію подорожей містами України, сприяючи розвитку внутрішнього туризму. Висновки та рекомендації є логічними, структурованими та придатними для впровадження в умовах реальної туристичної платформи. Розроблений функціонал може стати основою для повноцінного сервісу, а запропоновані моделі - базою для подальшого масштабування і доповнення новими сервісами.

ФОП Жуковський Дмитро Русланович

(підпис, ПІБ)

АНОТАЦІЯ

кваліфікаційної бакалаврської роботи
здобувача першого (бакалаврського) рівня вищої освіти, 4 курсу,
Герасименка Максима Ігоровича
виконаної на тему: «Проектування та розроблення інформаційної системи
«Подорож містами України»»
Київ: кафедра інформаційних систем в економіці, 2025 р.

Кваліфікаційний бакалаврський проєкт присвячений проєктуванню та розробці інформаційної системи «Подорож містами України» для підтримки туристичної спільноти з використанням сучасних інформаційних технологій.

Робота складається з трьох розділів, логічно пов'язаних між собою.

В першому теоретико-методичному розділі дана характеристика та аналіз предметної області, виділено фактори що впливають на організацію інформаційної системи, визначено сучасний стан проблеми, її актуальність та проаналізовано різноманітні варіації використання вже існуючих застосунків на обрану тематику.

В другому аналітико-дослідницькому розділі проведено аналіз і специфікацію вимог до інформаційної систем. Було виконано постановку задачі, сформовано вимоги до системи, розроблено інформаційну модель підтримки управління.

В третьому конструктивному розділі демонструються різні види забезпечень, які необхідні для роботи інформаційної системи: інформаційне, технічне та програмне.

У висновках міститься узагальнення результатів роботи у вигляді обґрунтованих підсумків.

РЕФЕРАТ

Кваліфікаційна бакалаврська робота містить 74 сторінки, 8 таблиць, 29 рисунків, список використаних джерел з 32 найменувань, 4 додатків.

ПРОЄКТУВАННЯ ТА РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ «ПОДОРОЖ МІСТАМИ УКРАЇНИ»

Перелік ключових слів: інформаційна система, веб-застосунок, проєктування, розробка, туристичні локації.

Об'єкт дослідження: процес формування та функціонування міського туризму в умовах сучасних реалій.

Предмет дослідження: інформаційні процеси та комп'ютерні технології з адміністрування подорожами містами України.

Мета і завдання дослідження: розробка інформаційної системи «Подорож містами України», яка посприяє зручному пошуку туристичних локацій та плануванню туристичних подорожей.

Відповідно до поставленої мети визначені такі *завдання*:

- проаналізувати потреби туристів для визначення ключових функцій інформаційної системи;
- спроектувати зручний інтерфейс інформаційної системи, спрямований на максимальне задоволення потреб користувачів;
- вибрати методології проєктування інформаційної системи;
- визначити забезпечення інформаційної системи та її опис;
- виконати частину програмної реалізації інформаційної системи.

Апаратні засоби, що використовувались при проєктуванні: персональний комп'ютер на базі Ryzen 5 5600.

Одержані результати можуть бути використані у сфері туристичних подорожей та їх планувань. Розроблено інформаційну систему яка дозволяє здійснювати зручний пошук туристичних локацій та має можливість збереження локацій в особистому кабінеті.

Рік виконання кваліфікаційної бакалаврської роботи – 2025.

Рік захисту роботи – 2025.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	3
ВСТУП	4
РОЗДІЛ 1	7
ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ	7
1.1 Характеристика галузі управління подорожами містами України	7
1.2 Сучасний стан проблеми та аналіз літературних джерел	14
РОЗДІЛ 2	19
РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ..	19
2.1 Аналіз і специфікація вимог до інформаційної системи	19
2.2 Постановка задачі та алгоритм розв’язання	23
2.2.1 Характеристика задачі	23
2.2.2 Вхідна інформація	25
2.2.3 Вихідна інформація	27
2.2.4 Алгоритм розв’язання задачі	28
2.3 Інформаційна модель підтримки управління подорожами	36
РОЗДІЛ 3	53
ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ	53
3.1 Інформаційне забезпечення	53
3.2 Технічне забезпечення	60
3.3 Програмне забезпечення	62
3.4 Можливості та доцільність впровадження	66
ВИСНОВКИ	70
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	72
ДОДАТКИ	75

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ІС – інформаційна система

БД – база даних

ПК – персональний комп'ютер

JS – JavaScript

HTML – HyperText Markup Language

CSS – Cascading Style Sheets

СУБД – Система управління базами даних

UML – Unified Modeling Language

SQL – Structured Query Language

UI – User Interface

SysML – Systems Modeling Language

ВСТУП

У сучасних умовах цифровізації суспільства особливої актуальності набувають інформаційні системи, що спрямовані на підтримку внутрішнього туризму, популяризацію національної культурної спадщини та підвищення туристичної привабливості міст України. З урахуванням сучасних викликів, спричинених військовою агресією та кризовими явищами в економіці, питання розвитку внутрішнього туризму стає надзвичайно важливим як для підтримки національної економіки, так і для збереження культурної ідентичності. Створення інноваційного цифрового продукту, що забезпечує зручну взаємодію між мандрівниками, туристичними об'єктами, сервісами та адміністраторами туристичних маршрутів, є нагальною потребою.

Актуальність теми пояснюється необхідністю надання користувачам сучасного та інтуїтивно зрозумілого інструменту для планування подорожей, ознайомлення з культурними, історичними та природними локаціями, а також організації зручної взаємодії з туристичними об'єктами на території України. У час, коли внутрішній туризм є одним із безпечніших та доступніших напрямків відпочинку, підтримка таких ініціатив через цифрові рішення сприяє як культурному, так і економічному розвитку країни.

Огляд наукових праць і практичних проєктів свідчить про значний інтерес до створення інформаційних платформ для туристичної галузі. Зокрема, у дослідженнях Г. І. Гапоненка, М. З. Згуровського, О. В. Євтушенка розглядаються питання цифровізації туризму та побудови геоінформаційних систем. Проте більшість з розроблених рішень мають загальний характер або зосереджуються на окремих туристичних напрямках, не охоплюючи потребу у комплексному представленні українських міст з інтегрованими маршрутами, подієвими анонсами, рекомендаціями та зворотним зв'язком від користувачів. Таким чином, розробка інформаційної системи «Подорож містами України» дозволяє заповнити цю нішу та забезпечити зручну платформу для організації подорожей у межах держави.

Метою цього курсового проєкту є проєктування та розроблення інформаційної системи, яка дозволяє користувачам здійснювати планування подорожі містами України, переглядати детальну інформацію про туристичні локації, маршрути, а також мати змогу створювати особисті маршрути, зберігати улюблені місця та взаємодіяти з контентом.

Завдання дослідження включають:

- аналіз аналогічних систем та визначення їх переваг і недоліків;
- формування функціональних і нефункціональних вимог до інформаційної системи;
- побудова моделі системи з використанням нотацій UML (діаграми прецедентів, класів, послідовності, компонентів тощо);
- розроблення структури бази даних для зберігання туристичної інформації;
- проєктування користувацького інтерфейсу з урахуванням принципів UX/UI;
- розробка окремих функціональних модулів та демонстрація роботи інформаційної системи.

Об'єктом дослідження є процес туристичного планування та ознайомлення з визначними місцями міст України засобами інформаційних технологій.

Предметом дослідження є програмно-інформаційна система для планування подорожей містами України, яка забезпечує інтерактивну взаємодію з туристичним контентом.

Методи дослідження, використані у роботі, включають системний аналіз, структурне проєктування, методи побудови інформаційних моделей, а також засоби програмування з використанням сучасних фреймворків та мов програмування.

Теоретична значущість роботи полягає у застосуванні принципів системного підходу та методів UML-моделювання для розв'язання задачі створення туристичної ІС.

Методична значущість полягає у побудові узагальненої архітектури туристичної системи, яка може бути адаптована до інших подібних проєктів у сфері веброзробки.

Практична значущість дослідження полягає у розробці готової до розгортання платформи, яка здатна забезпечити користувачів інформаційною підтримкою у плануванні подорожей Україною, а також у перспективі - слугувати інструментом для просування регіонального туризму.

Інформаційна база дослідження включає:

- результати досліджень з відкритих наукових джерел з питань веброзробки, системного аналізу, UX/UI;
- аналітичні матеріали Державного агентства розвитку туризму України;
- офіційні сайти туристичних управлінь українських міст;
- досвід використання платформ Google Maps, TripAdvisor, Zruchno.Travel та інших аналогів.

Робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків.

РОЗДІЛ 1

ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Характеристика галузі управління подорожами містами України

Предметною галуззю даного дослідження є **внутрішній туризм України**, що охоплює процеси організації, планування та адміністрування подорожей містами країни із застосуванням сучасних інформаційних технологій. Об'єктом дослідження виступає процес прийняття рішень щодо формування туристичних маршрутів, бронювання послуг та взаємодії користувачів із цифровими платформами у сфері подорожей Україною.

Галузь внутрішнього туризму в Україні демонструє динамічний розвиток, попри складні соціально-економічні умови. За даними Державного агентства розвитку туризму України (ДАРТ), у 2023 році внутрішній туризм став одним із ключових напрямів підтримки місцевих економік, оскільки закордонні подорожі були суттєво обмежені [1]. Лише за перше півріччя 2023 року туристичний збір сягнув понад 85 млн грн, що на 14 % більше ніж за аналогічний період попереднього року [2]. Зростання популярності подорожей українськими містами пов'язане не тільки з потребою у безпечному відпочинку, а й із підвищенням рівня патріотизму, зацікавленістю у відкритті національної спадщини та прагненням підтримати локальний бізнес.

Об'єкти галузі включають широкий спектр установ і бізнесів: державні органи туризму, зокрема ДАРТ та регіональні туристичні відділи, туристичні підприємства - туроператори, готелі, транспортні компанії, музеї, екскурсійні агенції, а також інформаційні платформи й сервіси бронювання. Значну роль відіграють цифрові платформи, які виступають інтерфейсом взаємодії між туристами й постачальниками туристичних послуг. Саме через такі системи

формується переважна більшість бронювань та інформаційна підтримка подорожей.

Прийняття рішень у даній предметній галузі відбувається на декількох рівнях. На рівні користувача рішення полягає у виборі міста чи маршруту для подорожі, визначенні оптимального періоду відпочинку, виборі засобів транспорту та місць проживання. При цьому рішення можуть ухвалюватися як індивідуально, так і групами - наприклад, сім'ями або туристичними колективами, що вимагає врахування колективних уподобань і забезпечення гнучкості системи бронювання. Адміністратори цифрових платформ приймають рішення щодо наповнення контенту, актуалізації даних про туристичні об'єкти, формування рекомендаційних систем. Туристичні оператори приймають рішення про включення нових продуктів до своїх пропозицій на основі аналітики попиту та поведінки користувачів. Таким чином, існує багаторівнева система взаємопов'язаних рішень, що потребує комплексної інформаційної підтримки.

Фактори, що впливають на середовище прийняття рішень, охоплюють сезонність туристичного попиту - за даними досліджень, близько 48 % усіх внутрішніх поїздок українці здійснюють у літній період, тоді як зимовий сезон демонструє значно нижчі показники активності [3]. Значний вплив має також безпекова ситуація: у 2023 році 45 % українців зазначили, що попри військові дії здійснювали внутрішні подорожі, розглядаючи їх як внесок у підтримку економіки та соціальне згуртування [4]. Іншим вагомим фактором є розвиток інтернет-інфраструктури: у регіонах із недостатньо розвиненими цифровими каналами обмежується можливість доступу до сучасних онлайн-сервісів, що створює додаткові виклики для цифровізації галузі. Нарешті, законодавчі та економічні умови, зокрема питання податкового регулювання та державної підтримки внутрішнього туризму, також істотно впливають на загальне середовище функціонування туристичної галузі [5].

Взаємозв'язки рішень у предметній галузі є складними та багаторівневими. Наприклад, рішення одного користувача про бронювання

певного маршруту або готелю може вплинути на доступність ресурсів для інших користувачів. Крім того, системи рекомендацій будують свої пропозиції на основі історичних даних про попередні вибори користувачів, створюючи ефект «соціального доказу», що також впливає на поведінку нових відвідувачів платформи. Рішення адміністраторів системи щодо введення акційних пропозицій або нових маршрутів можуть оперативно змінювати попит на окремі міста чи туристичні об'єкти, що потребує адаптивної архітектури інформаційної системи.

Інформаційна база для прийняття рішень включає як внутрішні, так і зовнішні джерела даних. Внутрішня інформація складається з баз даних про туристичні об'єкти, маршрути, бронювання, рейтинги та відгуки користувачів. Зовнішня інформація охоплює офіційні статистичні дані ДАРТ та Держстату України, прогнози погоди, дані про безпекову ситуацію, інформацію з API картографічних сервісів, а також контент соціальних мереж. Методи збору та оброблення включають автоматизований збір через інтеграцію API, використання алгоритмів машинного навчання для аналізу відгуків, евристичні методи для побудови рекомендаційних систем і аналітичні підходи для формування стратегічної інформації для адміністраторів системи.

В межах даного дослідження була детально розроблена **організаційна структура інформаційної системи "Подорож містами України"**, яка відіграє ключову роль у забезпеченні ефективної взаємодії між її основними суб'єктами - кінцевими користувачами та адміністраторами контенту. Головною метою системи є підвищення якості обслуговування користувачів, зацікавлених у внутрішньому туризмі, а також створення платформи, що сприятиме розвитку туристичної інфраструктури шляхом зручного представлення інформації та підтримки інтерактивної взаємодії.

На **системній діаграмі верхнього рівня** (рис. 1.1), яка відображає загальну архітектуру ІС, центральним елементом виступає сама система "Подорож містами України", що забезпечує ключовий функціонал для роботи користувачів - туристів та адміністраторів. Основними ролями в системі є

звичайні користувачі, які шукають інформацію, формують маршрути та зберігають улюблені локації, та адміністратори, які додають нові туристичні об’єкти, події та супровідні матеріали.

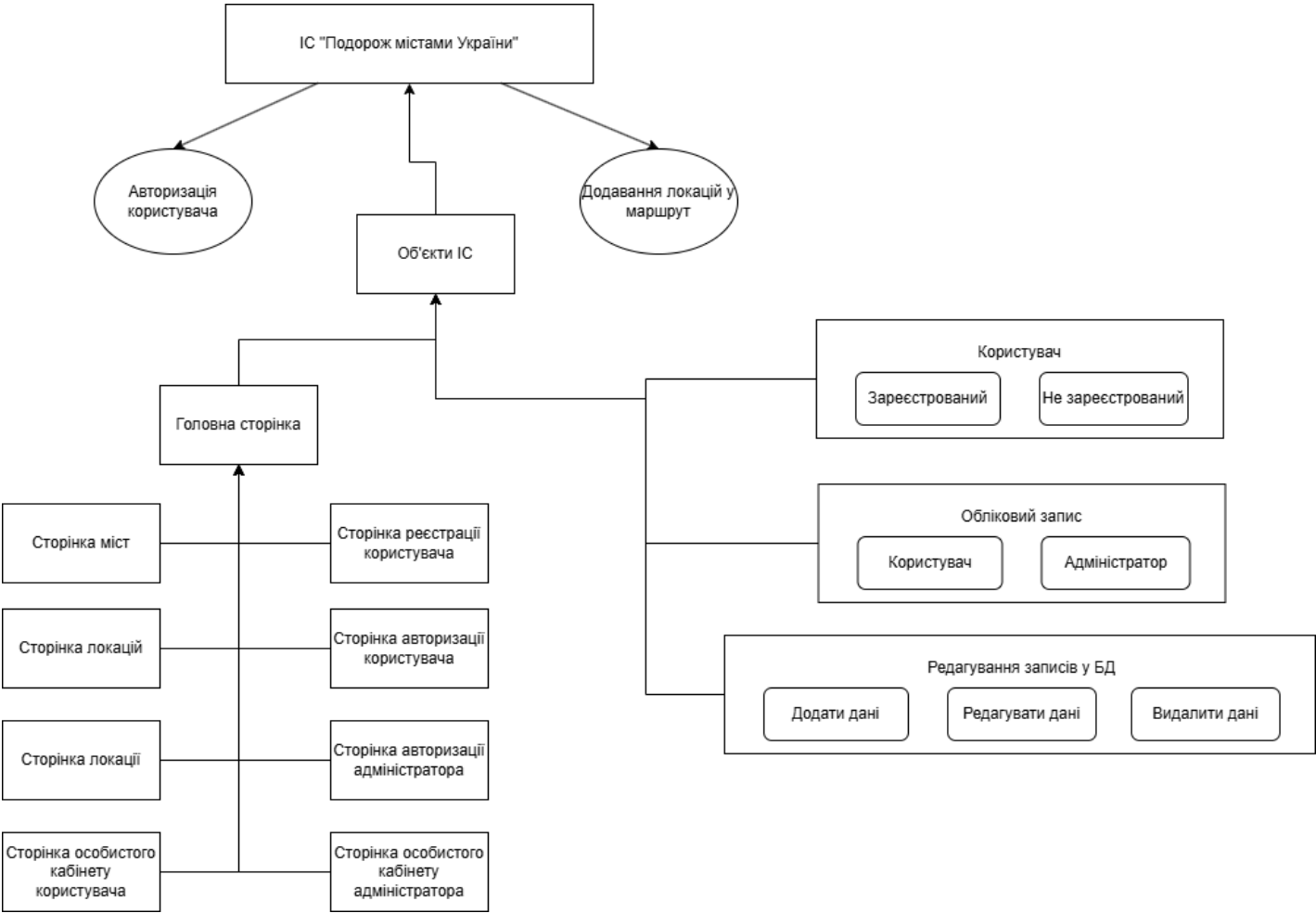


Рисунок 1.1 – Головна діаграма

Джерело: Створено автором

Для глибшого розкриття логіки роботи системи було побудовано низку **UML-діаграм нижчого рівня** (рис 1.2-1.4), які демонструють детальну структуру й поведінкові сценарії платформи:



Рисунок 1.2 – Діаграма "Додавання локацій у маршрут", що відображає процедуру пошуку та додавання локацій до персонального маршруту;
Джерело: Створено автором

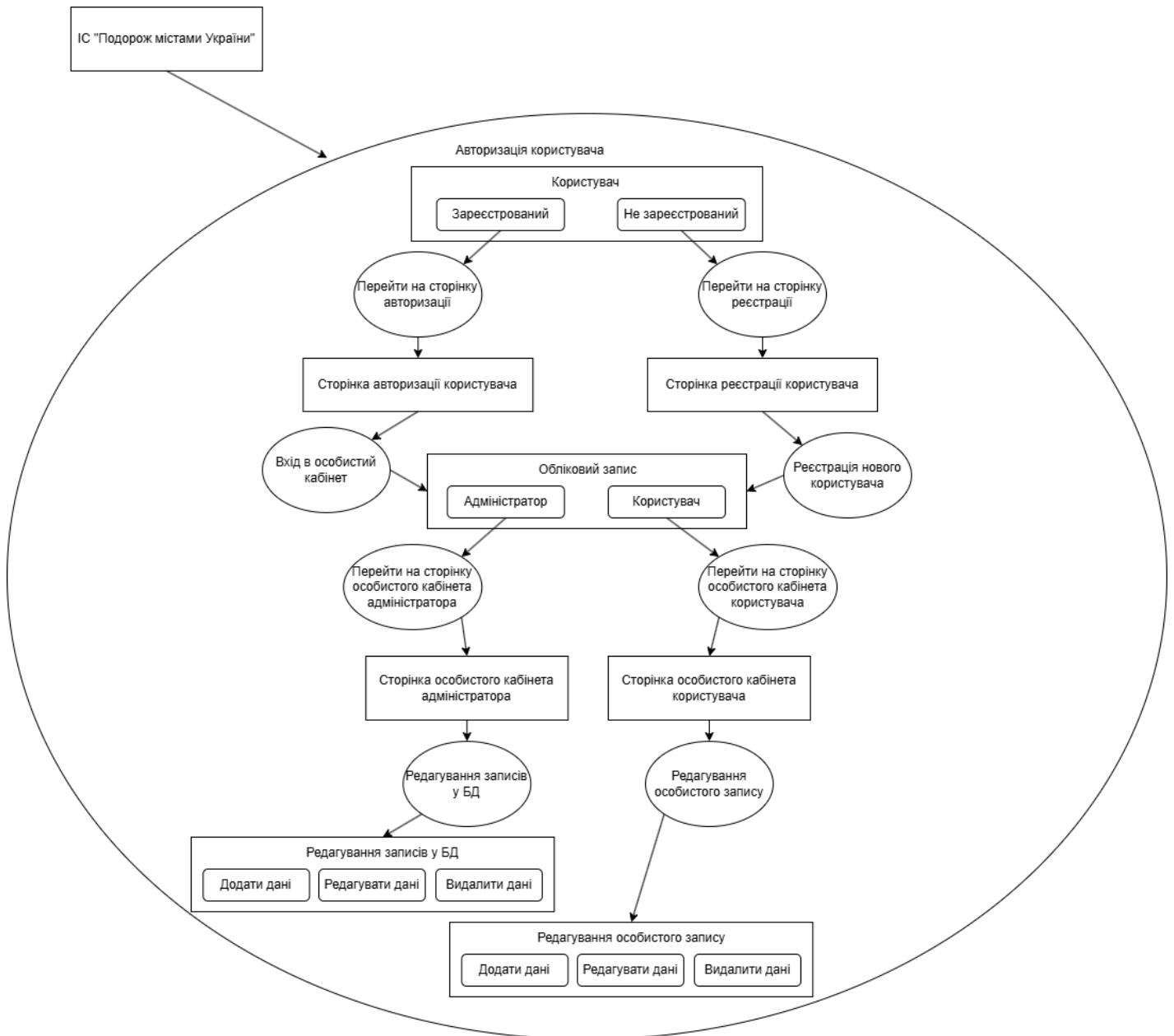


Рисунок 1.3 – Діаграма "Авторизація користувача", яка демонструє процес аутентифікації користувачів і адміністраторів із подальшим доступом до персоналізованого функціоналу;

Джерело: Створено автором

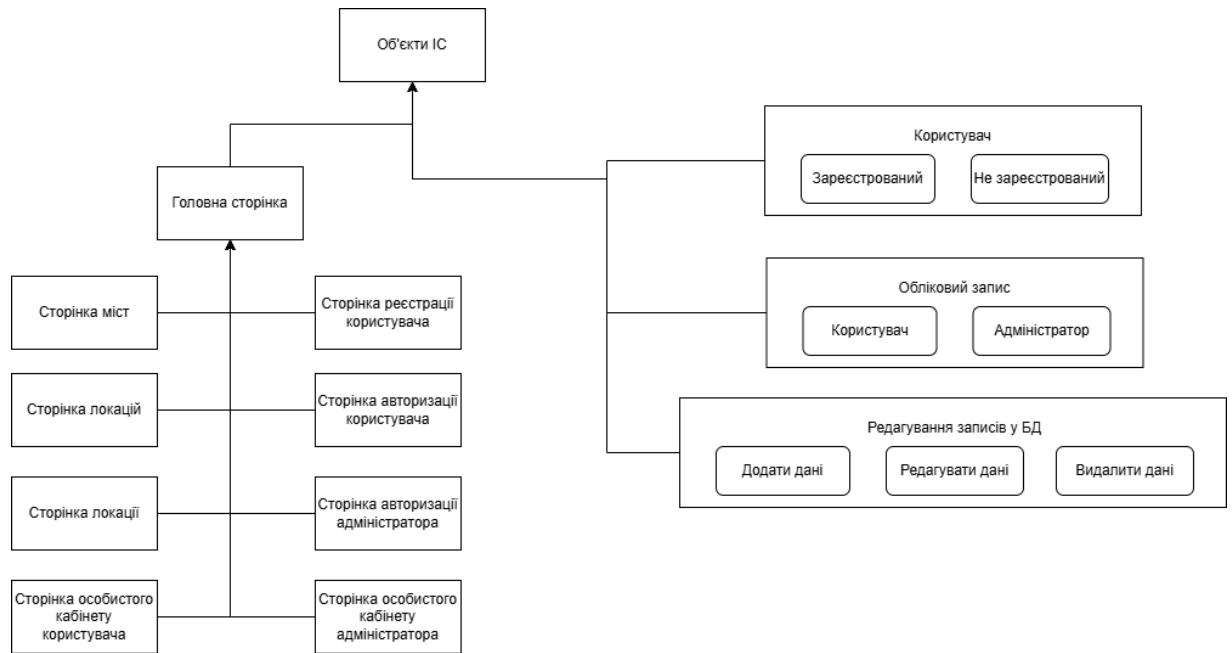


Рисунок 1.4 – Механізм розгортання, який наочно представляє взаємодію всіх ключових об'єктів системи та їхніх компонентів.

Джерело: Створено автором

Розроблені діаграми стали основою для подальшого формування функціональних вимог до системи, а також лягли в основу технічного проєктування і реалізації її архітектури. Їх побудова забезпечує повне розуміння як структурної організації, так і внутрішньої логіки роботи інформаційної системи "Подорож містами України".

Таким чином, організаційна структура ІС є інтегрованою складовою сучасної туристичної галузі, дозволяючи ефективно підтримувати процеси управління подорожами та забезпечувати якісний користувацький досвід.

1.2 Сучасний стан проблеми та аналіз літературних джерел

Аналіз трьох провідних українських платформ показує значний потенціал, але й виокремлює прогалини в уніфікації та розвитку функціональності, що підкреслює актуальність створення Цілісної інформаційної системи "Подорож містами України".

Платформа **Inside UA** (рис. 1.5) [6] надає зручну інтерфейсну картографічну мапу, що дозволяє користувачам переглядати базову інформацію про міста, їх пам'ятки та актуальні події. Інтерфейс досить інтуїтивний - зображення міст, контакти, фото - проте відсутня реєстрація, можливість додавати власні місця, бронювання або відгуки користувачів. Система виконує якісне представлення даних, але без інтерактивності, а отже користувачам бракує механізмів персоналізації.

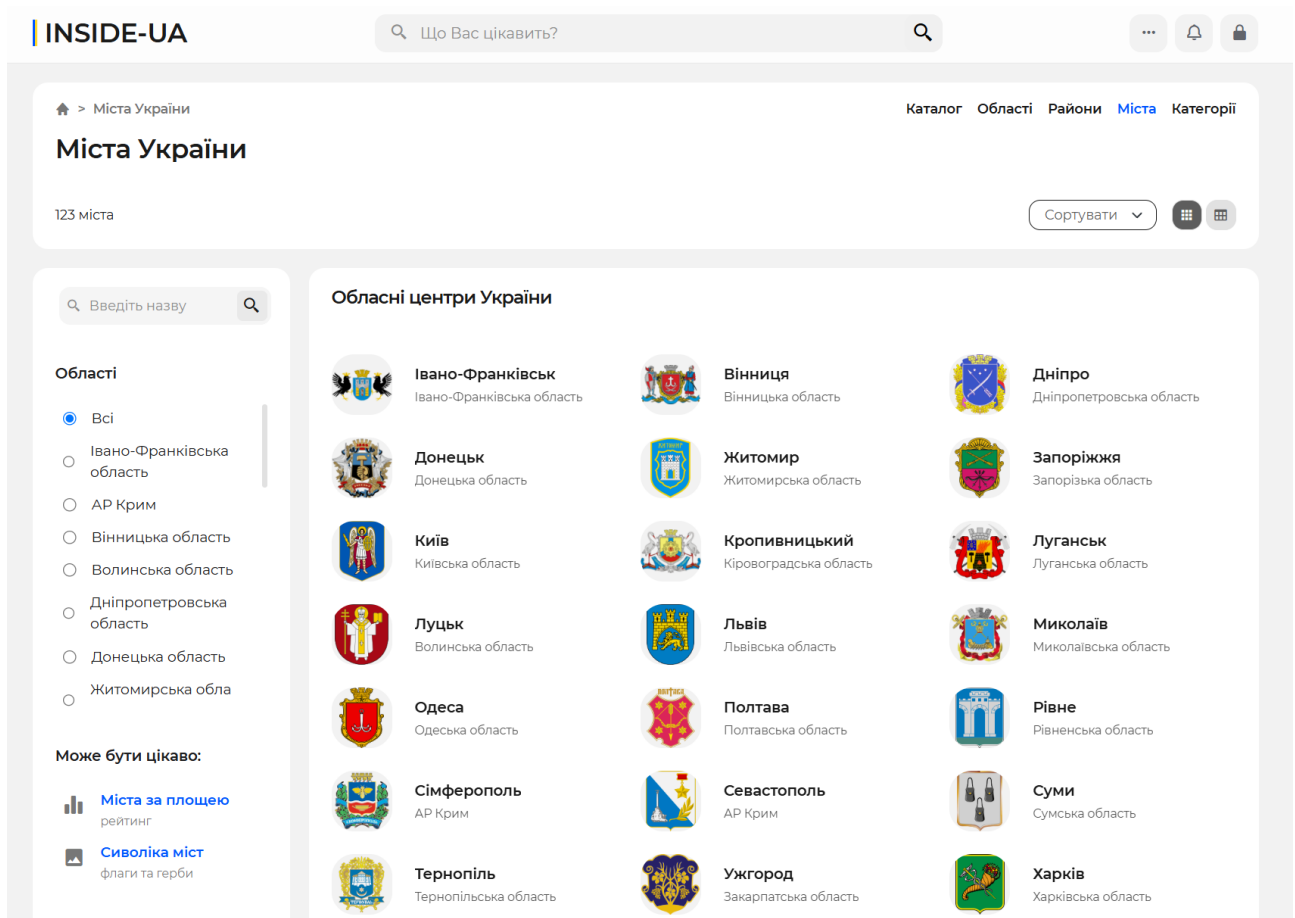


Рисунок 1.5 – Головна сторінка Inside UA

Джерело: Знімок екрану створено автором

Discover UA (рис. 1.6) [7] вже пропонує можливість створення маршрутів із розрахунком часу та дистанції, інтеграцію з Google Maps, 3D-турами та реєстрацію користувачів, що дозволяє залишати відгуки та ділитися локаціями. Його інтерфейсна зручність на досить високому рівні, доступний мобільний адаптив, але відсутня можливість бронювання, аналітичного модуля та роботи з подіями і безпековими повідомленнями.

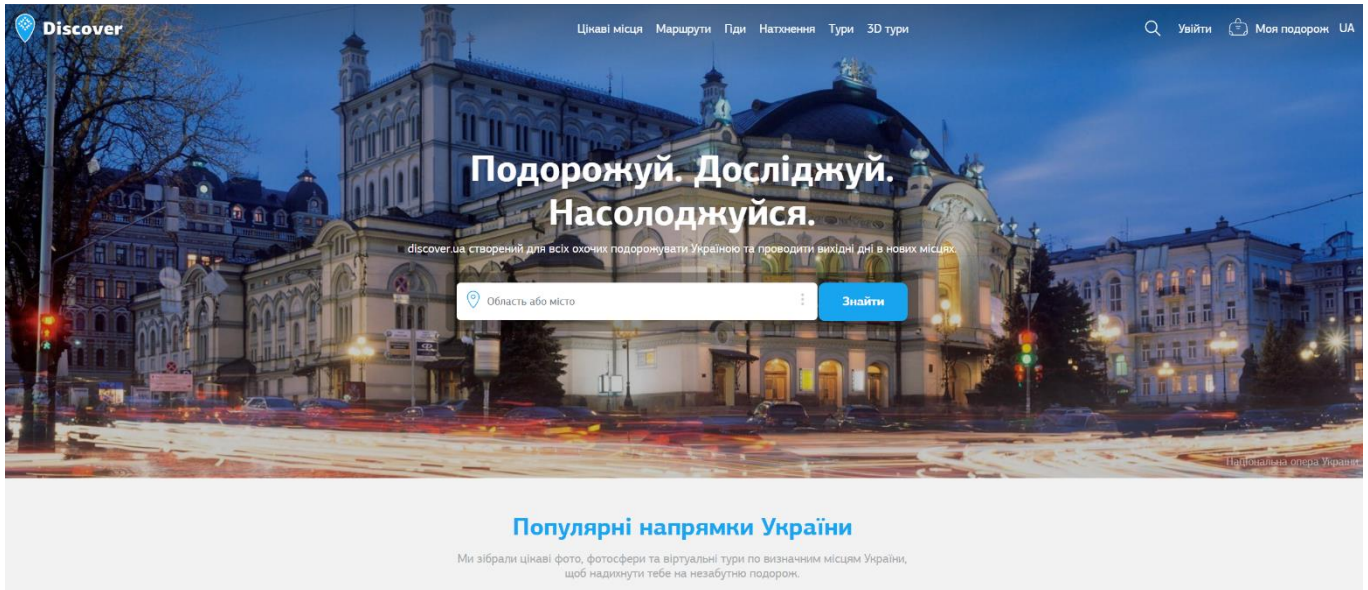


Рисунок 1.6 – Головна сторінка Discover UA

Джерело: Знімок екрану створено автором

MistaUA (рис. 1.7) [8] - це каталог міст і подій з інтуїтивно простим доступом до інформації, актуальних заходів, історичних фактів. Зручний для перегляду, однак не підтримує персоналізовані маршрути, авторизацію, бронювання чи аналітику. Це надає можливість формалізувати пропозиції, але лишає функціонал незакінченим.

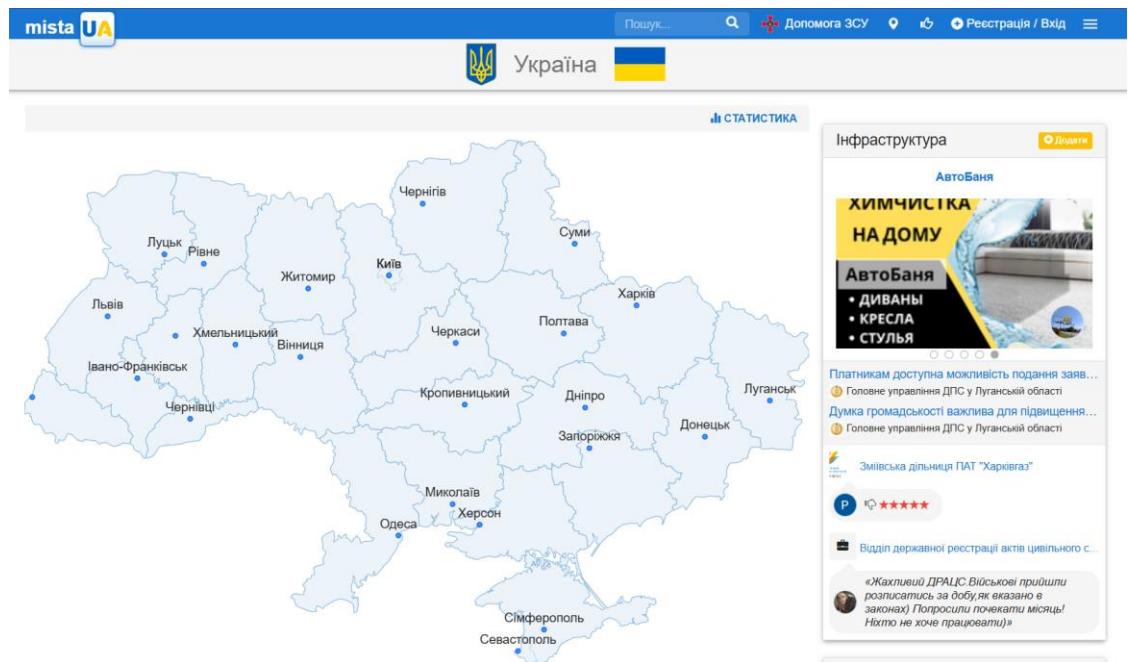


Рисунок 1.7 – Головна сторінка MistaUA

Джерело: Знімок екрану створено автором

Наукові дослідження вказують на високий потенціал цифрових рішень для туризму, але їхнє впровадження часто обмежується вузькими нішами. У зарубіжних джерелах наголошується, що цілісні платформи, подібні до Booking[9], TripAdvisor[10] або Expedia[11], забезпечують значно ширший спектр функцій: маршрути, бронювання, аналітику, рекомендації, соціальний зворотний зв'язок.

Також наведена порівняльна таблиця (табл. 1.1), що включає дизайн, UX[12], технічні можливості та функціональні особливості:

Таблиця 1.1 – Порівняльна характеристика існуючих українських платформ організації туризму

Назва	Inside UA	Discover UA	MistaUA
UX інтуїтивність	Висока	Висока	Середня
Реєстрація та авторизація	Присутня	Присутня	Присутня

Продовження таблиці 1.1

Назва	Inside UA	Discover UA	MistaUA
Перегляд туристичних локацій	Часткова реалізація окремими сторінками	Повна реалізація за містами	Відсутній
Створення туристичних маршрутів	Ні	Так	Ні
Наявність відгуків/рейтингів	Ні	Так	Ні
Аналітика турів	Відсутня	Відсутня	Відсутня
Додавання власних локацій	Відсутнє	Відсутнє	Відсутнє
Наявність нав'язливої реклами	Ні	Ні	Так
Зручний пошук	Ні	Так	Ні
Доступність інформаційного продукту	Безкоштовний	Безкоштовний	Безкоштовний
Використання API сервісів	Так	Так	Так

Джерело: сформовано автором на основі виконаного дослідження

Згідно з даними, наведеними в таблиці 1.1, можна виокремити деякі особливості, які можуть бути корисними при розробці інформаційної системи «Подорож містами України»:

- **інтуїтивний інтерфейс** перегляду локацій і маршрутів, ефективний та зрозумілий для користувачів різного рівня підготовки;
- **можливість створення маршрутів** і персоналізованого планування подорожей;
- **реалізація системи відгуків і рейтингів**, що дозволяє користувачам ділитися досвідом і допомагає у виборі локацій;
- **зручний пошук** по містах, локаціях, подіях;
- **адаптивність під мобільні пристрої**, що відповідає сучасним тенденціям використання туристичних сервісів.

Отже, розглянувши таблицю, можна побачити, що найбільш функціональною системою на даному етапі є Discover UA. Веб-застосунок має найбільш повний функціонал з точки зору можливостей перегляду

туристичних локацій, створення маршрутів, наявності відгуків та рейтингів. Платформа має інтуїтивно зрозумілий інтерфейс, підтримує зручний пошук і є доступною на мобільних пристроях. Наявність можливості додавання власних локацій наразі відсутня, як і аналітичний модуль, що створює можливості для покращення.

Друге місце займає Inside UA. Платформа вирізняється зручністю подачі інформації про міста та частковою реалізацією перегляду локацій, проте функціонально поступається Discover UA через відсутність підтримки створення маршрутів, системи рейтингів та відгуків, а також менш розвинену пошукову систему. Разом із тим Inside UA є якісним прикладом візуально привабливого та зручного оглядового ресурсу.

Третє місце займає Mista UA. Хоча платформа дозволяє переглядати загальну інформацію про міста та події, вона має найбільш обмежений функціонал. Відсутні можливості створення маршрутів, рейтингів, системи персоналізації, а також зручний пошук. Наявність нав'язливої реклами дещо знижує зручність використання. Разом із тим простота використання ресурсу та доступність інформації роблять його зручним для швидкого ознайомлення з основною інформацією.

В цілому, проведений аналіз показує, що на ринку України вже існують приклади якісних рішень для підтримки внутрішнього туризму, однак немає платформи, яка б уніфікувала повний спектр необхідного функціоналу: створення персоналізованих маршрутів, бронювання, аналітики, відгуків, роботи з API[13] безпекових сервісів та інтеграції подій. Це створює значні перспективи для реалізації інформаційної системи «Подорож містами України», яка може зайняти свою нішу, інтегруючи кращі практики існуючих рішень та усуваючи їхні обмеження.

РОЗДІЛ 2

РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Аналіз і специфікація вимог до інформаційної системи

Інформаційна система «Подорож містами України» має стати ефективним інструментом для підтримки та розвитку внутрішнього туризму в Україні. Для забезпечення успішної розробки та подальшого впровадження системи було проведено детальний аналіз і специфікацію вимог, які визначають її функціональність, якість та обмеження. Якісно сформульовані вимоги виступають запорукою створення системи, яка повною мірою відповідатиме потребам користувачів та цілям бізнесу.

Бізнес-вимоги задають стратегічний вектор розробки системи, орієнтуючи її на досягнення конкретної цінності як для кінцевих користувачів, так і для організації, яка забезпечуватиме її експлуатацію. Основними бізнес-вимогами виступають підвищення ефективності бізнес-процесів компанії, автоматизація основних процесів, забезпечення масштабованості та розвитку системи, а також створення стабільного, якісного програмного продукту з можливістю технічної підтримки та подальшого адміністрування. Окрему увагу приділено створенню зручного сервісу для користувачів - туристів, які зможуть планувати власні маршрути, економити час на організацію подорожей та отримувати безпечний доступ до персональних даних. Важливою вимогою також є відповідність національним та міжнародним стандартам захисту інформації. Візуалізація бізнес-вимог наведена на діаграмі бізнес-вимог (рис. 2.1).

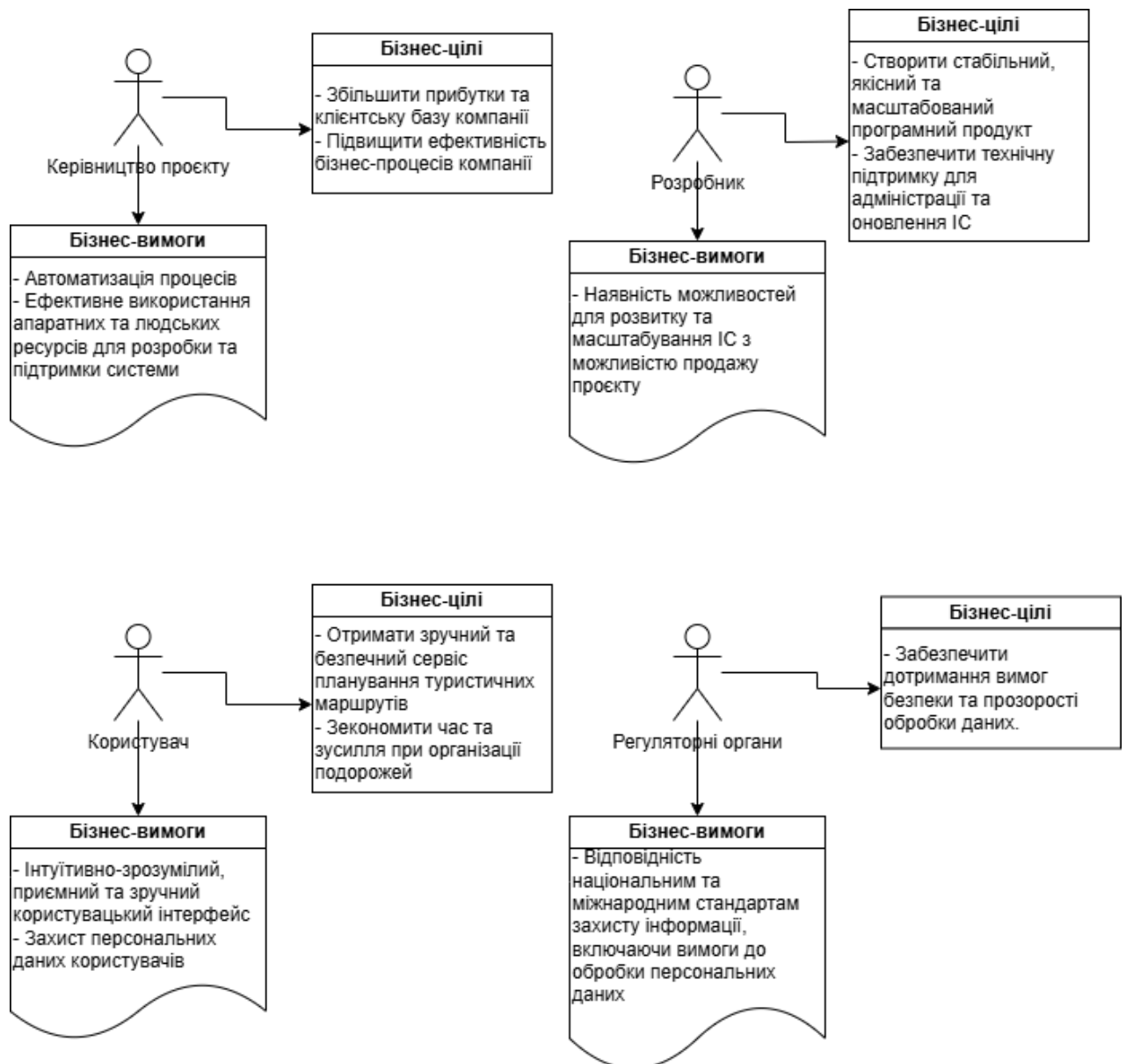


Рисунок 2.1 – Діаграма бізнес-вимог до ІС «Подорож містами України»

Джерело: Створено автором

Функціональні вимоги визначають перелік функцій, які повинна підтримувати система для досягнення зазначених бізнес-цілей. Інформаційна система передбачає наявність системи особистого кабінету користувача, яка дозволить переглядати та редагувати особистий список локацій, що зацікавили користувача. Окрім того, користувачі зможуть переглядати список доступних міст та локацій, здійснювати вибір локацій і додавати нові локації до власних маршрутів.

Система аутентифікації користувачів забезпечуватиме можливість реєстрації, авторизації та доступу до особистого кабінету, а також окремий механізм авторизації для адміністратора. Адміністративна підсистема дозволить отримувати доступ до інформації про локації та користувачів, а також здійснювати додавання та редагування відповідних даних. Структуру функціональних вимог відображено на діаграмі функціональних вимог (рис. 2.2).

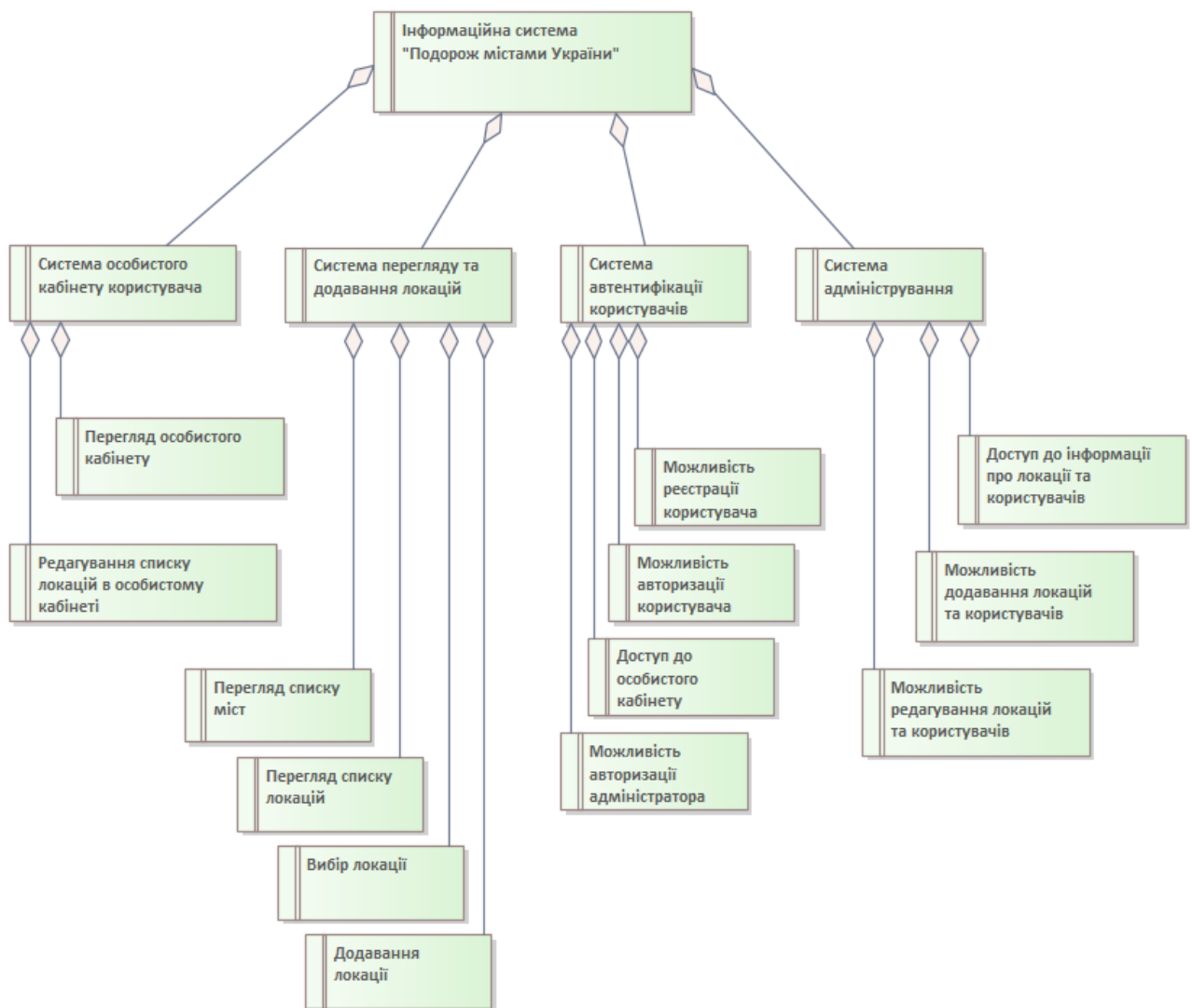


Рисунок 2.2 – Діаграма функціональних вимог до ІС «Подорож містами України»

Джерело: Створено автором

Нефункціональні вимоги формують якісні характеристики системи, які не пов'язані безпосередньо з її функціональністю, однак критично важливі

для забезпечення задоволення користувацьких очікувань. Інформаційна система має гарантувати високу продуктивність, що передбачає швидку обробку запитів і підтримку паралельного доступу користувачів. Зручність використання системи забезпечуватиметься завдяки інтуїтивній навігації, адаптивному інтерфейсу та сумісності з усіма популярними браузерами.

У сфері безпеки особливу увагу приділено шифруванню персональних даних, використанню HTTPS-протоколу[14], авторизації з різними рівнями доступу, а також захисту від несанкціонованого доступу. Система повинна бути надійною - передбачено автоматичне резервне копіювання даних, відновлення після збоїв і забезпечення безперебійної роботи. Масштабованість і гнучкість системи досягаються завдяки підтримці інтеграції з API та реалізації гнучкої архітектури. Діаграму нефункціональних вимог наведено на рис. 2.3.

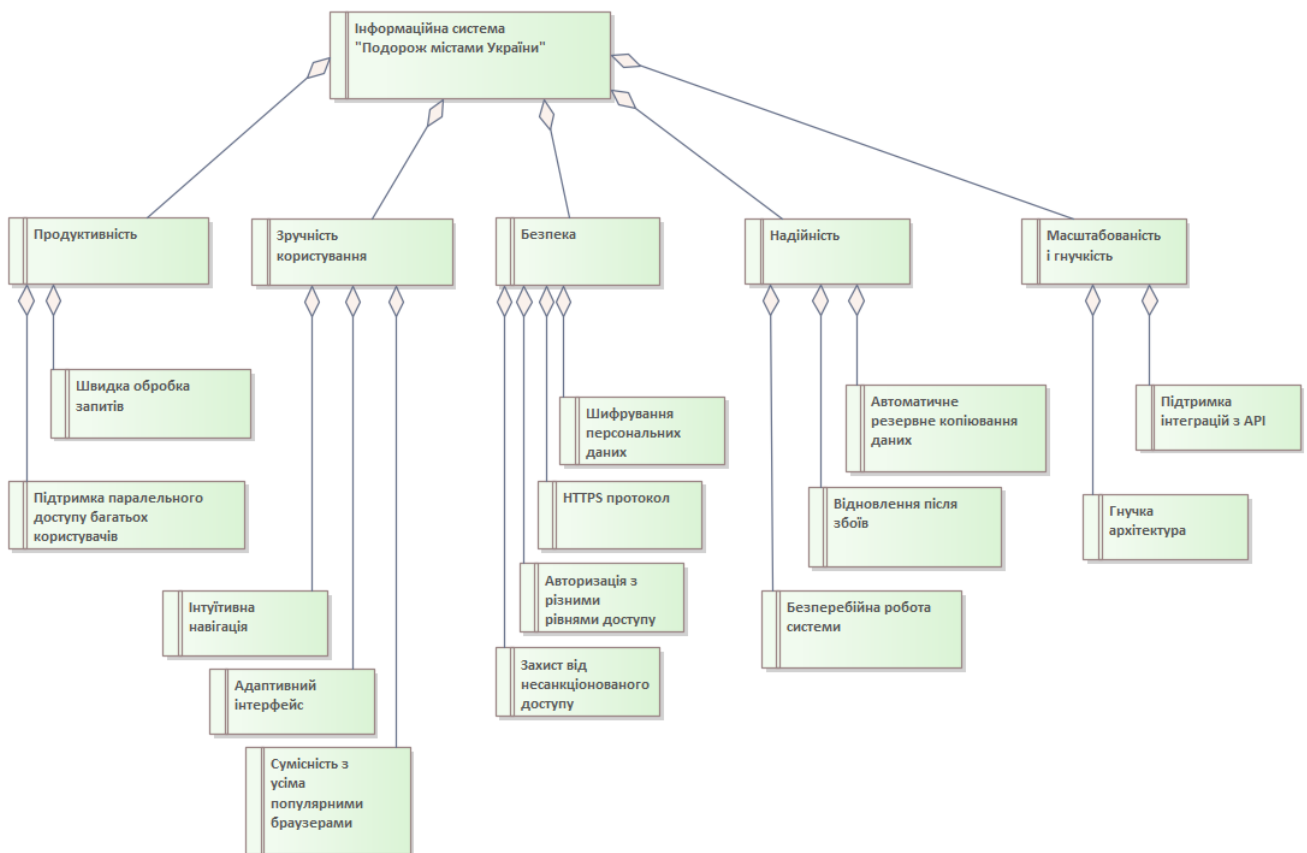


Рисунок 2.3 – Діаграма нефункціональних вимог до ІС «Подорож містами України»

Джерело: Створено автором

Формування вимог до системи базувалося на аналізі потреб користувачів туристичних платформ, функціональних можливостей існуючих аналогів (Inside UA, Discover UA, Mista UA), нормативних вимог щодо захисту персональних даних, а також на очікуваннях адміністраторів та бізнес-цілях розробника. У процесі розроблення вимог було враховано необхідність їх чіткості, актуальності, коректності, повноти та відсутності внутрішніх суперечностей.

Таким чином, сформована модель вимог до інформаційної системи «Подорож містами України» забезпечує цілісне бачення як функціональних можливостей системи, так і якісних характеристик, які мають бути реалізовані під час її розробки. На основі цієї моделі буде здійснено подальше проектування архітектури системи та визначено підходи до її впровадження.

2.2 Постановка задачі та алгоритм розв'язання

2.2.1 Характеристика задачі

Розглядаючи економічний зміст задач оброблення інформації у проєктованій інформаційній системі «Подорож містами України», першочерговим завданням є вивчення потреб користувачів, які планують внутрішні подорожі, цікавляться туристичними локаціями та прагнуть зручно формувати маршрути для поїздок країною. Основною цільовою аудиторією є як звичайні туристи, так і внутрішні мандрівники, а також адміністративний персонал, що працює з наповненням контенту платформи.

Аналіз потреб цієї категорії користувачів дозволяє визначити актуальний функціонал системи: можливість швидкого пошуку міст та локацій, формування персоналізованих маршрутів, збереження улюблених місць, зручна навігація та доступ до детальної інформації про об'єкти туристичної інфраструктури. Система має підтримувати як базові функції для

неавторизованих користувачів (перегляд доступного контенту), так і розширений функціонал у кабінеті авторизованого користувача (редагування маршрутів, персоналізація профілю).

Модель монетизації платформи може передбачати декілька варіантів підписки або розширених можливостей для адміністраторів туристичних агенцій та локальних бізнесів. Наприклад, функції розширеного управління маршрутами, завантаження комерційного контенту (акції, тури), або пріоритетного відображення локацій можуть бути надані у межах окремого пакету для бізнес-користувачів. Вартість такої підписки, орієнтовно, може становити близько 100 гривень на місяць, що є конкурентною ціною серед аналогічних платформ на ринку.

Таким чином, аналіз економічного потенціалу користувачів дозволяє не лише оптимізувати стратегію ціноутворення, а й сформувати актуальний та життєздатний функціонал системи, орієнтований на потреби цільової аудиторії.

В інформаційній моделі задачі[15] на рис. 2.4 відображено вихідні та вхідні повідомлення, а також довідники та масиви системи, що забезпечують повний цикл інформаційного обслуговування користувачів та адміністраторів у процесі планування подорожей містами України.



Рисунок 2.4 – Інформаційна модель задачі

Джерело: Створено автором

2.2.2 Вхідна інформація

У якості вхідної інформації інформаційна система «Подорож містами України» передбачає введення різноманітних даних, пов'язаних із користувачами, локаціями, маршрутами та їх персональними вподобаннями. До вхідної інформації належать текстові дані (імена, описи локацій, маршрути), числові параметри (ідентифікатори, рейтинги), а також файли-довідники, що забезпечують наповнення контентної бази системи.

Особливістю проєктованої системи є те, що дані надходять з кількох джерел. Персональні дані користувача передаються при реєстрації, вході в систему або редагуванні профілю. Інформація про локації, маршрути та міста генерується як з внутрішніх джерел (адміністратори додають або редагують контент), так і з зовнішніх файлів-довідників, які підтримують актуальність бази даних. Запити користувачів - такі як пошук локацій або перегляд міст - надходять у вигляді екранних форм у режимі реального часу.

Критично важливими є вимоги до актуальності та коректності вхідної інформації, оскільки саме від цього залежить якість формованих маршрутів і корисність наданої інформації для користувачів. Зокрема, дані про локації мають бути актуальними та структурованими, а персональні дані користувача - захищеними відповідно до вимог законодавства.

Операції з введення персональних маршрутів, редагування списків локацій або зворотного зв'язку здійснюються миттєво і передбачають інтерактивний режим взаємодії. Файлові довідники (міст, локацій, маршрутів, користувачів) оновлюються відповідно до змін в БД та підтримують цілісність і повноту інформаційної моделі системи.

Перелік і опис вхідних повідомлень наведено у табл. 2.1.

Таблиця 2.1 – Перелік і опис вхідних повідомлень

Назва вхідного повідомлення	Ідентифікатор	Форма подання	Термін і частота надходження	Джерело
1	2	3	4	5
Дані для входу в систему	D_VHID	Екранна форма	За потребою	Авторизований користувач
Дані для реєстрації в системі	D_REJESTR	Екранна форма	За потребою	Неавторизований користувач
Запит на перегляд списку міст	D_CITY	Екранна форма	За потребою	Неавторизований / авторизований користувач
Запит на перегляд локацій міста	D_LOCATIONS	Екранна форма	За потребою	Неавторизований / авторизований користувач
Додавання локації до маршруту	D_ADD_LOC	Екранна форма	За потребою	Авторизований користувач
Редагування маршруту	D_EDIT_ROUTE	Екранна форма	За потребою	Авторизований користувач
Видалення локації з маршруту	D_REMOVE_LOC	Екранна форма	За потребою	Авторизований користувач
Зміна особистих даних користувача	D_USER_EDIT	Екранна форма	За потребою	Авторизований користувач
Заповнення форми зворотного зв'язку	D_FEEDBACK	Екранна форма	За потребою	Неавторизований / авторизований користувач
Адміністративне додавання нової локації	D_ADMIN_ADD_LOC	Екранна форма	За потребою	Адміністратор
Адміністративне редагування локації	D_ADMIN_EDIT_LOC	Екранна форма	За потребою	Адміністратор
Довідник користувачів	USER	Файл	По мірі реєстрації користувачів	Система
Довідник міст	CITY	Файл	По мірі оновлення бази даних	Система
Довідник локацій	LOCATIONS	Файл	По мірі оновлення бази даних	Система
Довідник маршрутів користувачів	ROUTES	Файл	По мірі збереження або редагування маршрутів	Авторизований користувач

Продовження таблиці 2.1

1	2	3	4	5
Довідник форм зворотного зв'язку	FEEDBACK	Файл	По мірі надходження форм	Неавторизований / авторизований користувач

Джерело: сформовано автором на основі виконаного дослідження

2.2.3 Вихідна інформація

Вихідна інформація включає результати взаємодії користувача та адміністратора із системою. Вона охоплює як підтвердження виконаних дій (реєстрація, вхід у систему, додавання локацій, формування маршрутів), так і відображення актуальних даних із бази: довідників користувачів, міст, локацій, маршрутів.

Підтвердження дій, таких як успішна реєстрація або авторизація, формуються миттєво для зручності користувача. Результати пошуку локацій або міст відображаються у відповідь на запити користувача, дозволяючи швидко знайти необхідну інформацію. Окремо важливим є відображення інформації з довідників - ці дані є актуалізованою базою знань системи і слугують основою для подальшої побудови маршрутів.

У системі передбачено підтвердження редагування профілю користувача, додавання маршрутів, редагування списків у кабінеті користувача. Користувач також може отримувати інформацію про виконання адміністративних операцій - наприклад, про оновлення довідника локацій або про результати модерації контенту.

Важливою частиною вихідної інформації є зворотний зв'язок - повідомлення користувачів про роботу системи, їх зауваження або пропозиції. Всі форми зворотного зв'язку зберігаються у відповідному довіднику і можуть бути використані для подальшого аналізу та вдосконалення функціоналу ІС.

Таким чином, вихідна інформація не тільки забезпечує комфорт взаємодії користувача із системою, але й формує основу для подальшого розвитку інформаційної системи та підвищення її якості.

Перелік і опис вихідних повідомлень наведено у табл. 2.2.

Таблиця 2.2 – Перелік і опис вихідних повідомлень

Назва вхідного повідомлення	Ідентифікатор	Форма подання	Термін і частота надходження	Джерело
1	2	3	4	5
Підтвердження входу в систему	RES_VHID	Екранна форма	За потребою	Авторизований користувач
Підтвердження реєстрації	RES_REJESTR	Екранна форма	За потребою	Неавторизований користувач
Відображення списку міст	RES_CITY	Екранна форма	За потребою	Всі користувачі
Відображення списку локацій міста	RES_LOCATIONS	Екранна форма	За потребою	Всі користувачі
Відображення персонального маршруту користувача	RES_ROUTE	Екранна форма	За потребою	Авторизований користувач
Підтвердження додавання локації у маршрут	RES_ADD_LOC	Екранна форма	За потребою	Авторизований користувач
Підтвердження редагування маршруту	RES_EDIT_ROUTE	Екранна форма	За потребою	Авторизований користувач
Підтвердження видалення локації з маршруту	RES_REMOVE_LOC	Екранна форма	За потребою	Авторизований користувач
Інформація про профіль користувача	RES_USER_INFO	Екранна форма	За потребою	Авторизований користувач
Підтвердження надсилання форми зворотного зв'язку	RES_FEEDBACK	Екранна форма	За потребою	Всі користувачі
Оновлена інформація про локацію (адміністративні дії)	RES_ADMIN_LOC	Екранна форма	За потребою	Адміністратор

Джерело: сформовано автором на основі виконаного дослідження

2.2.4 Алгоритм розв'язання задачі

У табл. 2.3 наведено масиви використаної та результатної інформації інформаційної системи «Подорож містами України». Масив форм зворотного зв'язку (FORM) зберігає контактні дані користувачів та їх повідомлення до

адміністратора системи з пропозиціями або зауваженнями щодо функціональності ресурсу. Масив локацій (LOCATION) містить повну довідкову інформацію про туристичні об'єкти, доступні в системі: назву, опис, зображення, координати, тип локації тощо. Масив користувачів (USER) акумулює персональні дані користувачів: ім'я, контактні дані, облікові записи, історію авторизацій та персональні налаштування. Масив маршрутів (ROUTE) зберігає створені користувачами маршрути, які включають послідовність обраних локацій, їх налаштування та дату створення. Масив списків улюблених локацій (LOCLIST) дозволяє користувачам формувати персоналізовані добірки локацій за власними інтересами та швидко до них повертатися під час планування подорожі.

Кожен масив має власний унікальний ідентифікатор та не обмежений у кількості записів, що дозволяє системі масштабуватись відповідно до зростання обсягу даних у процесі її експлуатації.

Таблиця 2.3 – Перелік масивів використаної та результатної інформації

Масив	Ідентифікатор	Максимальна кількість записів
1	2	3
Масив форм зворотнього зв'язку	FORM	Без обмежень
Масив локацій	LOCATION	Без обмежень
Масив користувачів	USER	Без обмежень
Масив маршрутів	ROUTE	Без обмежень
Масив списків улюблених локацій	LOCLIST	Без обмежень

Джерело: сформовано автором на основі виконаного дослідження

Математичне забезпечення інформаційної системи забезпечує автоматизацію ключових обчислювальних задач, що виникають під час взаємодії користувача з системою:

1) Розрахунок середнього рейтингу локації (2.1):

Середнє значення оцінки для певної локації обчислюється як середнє арифметичне всіх виставлених рейтингів:

$$R_l = \frac{1}{n_l} \sum_{j=1}^{n_1} R_j, \quad (2.1)$$

де R_j – рейтинг, виставлений j -тим користувачем; n_l – кількість користувачів, які оцінили локацію

2) Розрахунок кількості локацій у маршруті (2.2):

При додаванні нових локацій у маршрут:

$$N = N_0 + M \quad (2.2)$$

де N_0 – початкова кількість локацій у маршруті, M – кількість доданих локацій.

3) Розрахунок кількості маршрутів користувача (2.3):

При створенні нового маршруту кількість маршрутів для користувача u збільшується:

$$S_u = S_u + 1 \quad (2.3)$$

Побудова алгоритму розв'язання задачі для інформаційної системи «Подорож містами України» є важливою складовою її ефективної реалізації та подальшого функціонування. Розробка чітких і зрозумілих алгоритмів має велике значення з декількох причин. По-перше, ефективний алгоритм дозволяє автоматизувати та оптимізувати виконання ключових операцій і процесів системи, що значно спрощує роботу для користувачів і підвищує загальну продуктивність ІС. По-друге, коректно побудований алгоритм забезпечує точність та надійність виконання задач, що є особливо важливим при роботі з персональними даними користувачів та інформацією про туристичні локації.

Крім того, розробка алгоритму дозволяє підвищити масштабованість системи, що забезпечує її стабільну роботу навіть за умов зростання кількості користувачів, локацій та інформаційних потоків. Це робить інформаційну систему більш гнучкою та адаптивною до змін ринкових вимог і потреб аудиторії.

Слід також зазначити, що прозорість та зрозумілість алгоритмів сприяють полегшенню процесів розробки, тестування, супроводу та подальшого вдосконалення інформаційної системи. Саме тому побудова алгоритму розв'язання задачі є обов'язковим етапом процесу розробки ІС, що сприяє досягненню її відповідності очікуванням користувачів.

На рис. 2.5 відображено основу алгоритму розв'язання задачі для інформаційної системи «Подорож містами України». Від цієї базової структури користувач або адміністратор починає взаємодію з системою. Після виведення головної сторінки користувач обирає шлях: адміністратор переходить до сторінки авторизації адміністратора, а звичайний користувач - до сторінки авторизації або реєстрації (якщо облікового запису ще немає).

Після авторизації користувач потрапляє на головну сторінку, де здійснюється вибір розділу. В залежності від обраного розділу алгоритм направляє користувача до відповідного функціонального модуля системи. Це може бути: перегляд та додавання локацій, перегляд і редагування особистого кабінету користувача, доступ до адміністративної частини для модерації локацій та користувачів (для адміністратора), робота із системою зворотного зв'язку.

На рис. 2.6-2.8 відображено модулі алгоритму які належать до основного алгоритму та є його складовими. Серед них алгоритм модуля «Сторінка міста», «Особистий кабінет користувача», «Особистий кабінет адміністратора».

В алгоритмі також враховується інтеграція з базами даних (масивами інформації), а обробка відповідної вхідної та вихідної інформації здійснюється у відповідності до бізнес-логіки кожного модуля.

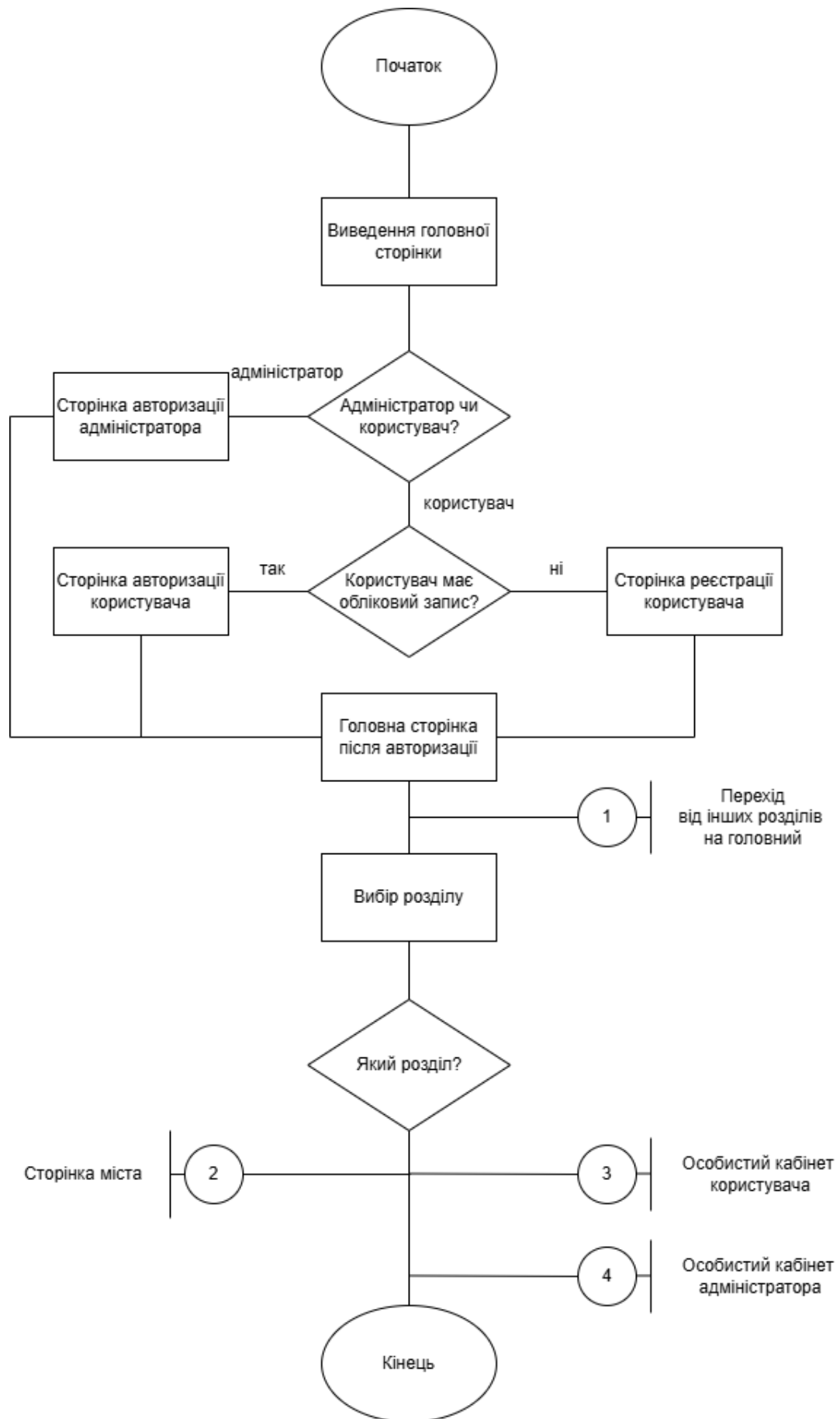


Рисунок 2.5 – Алгоритм – Основа алгоритму

Джерело: Створено автором

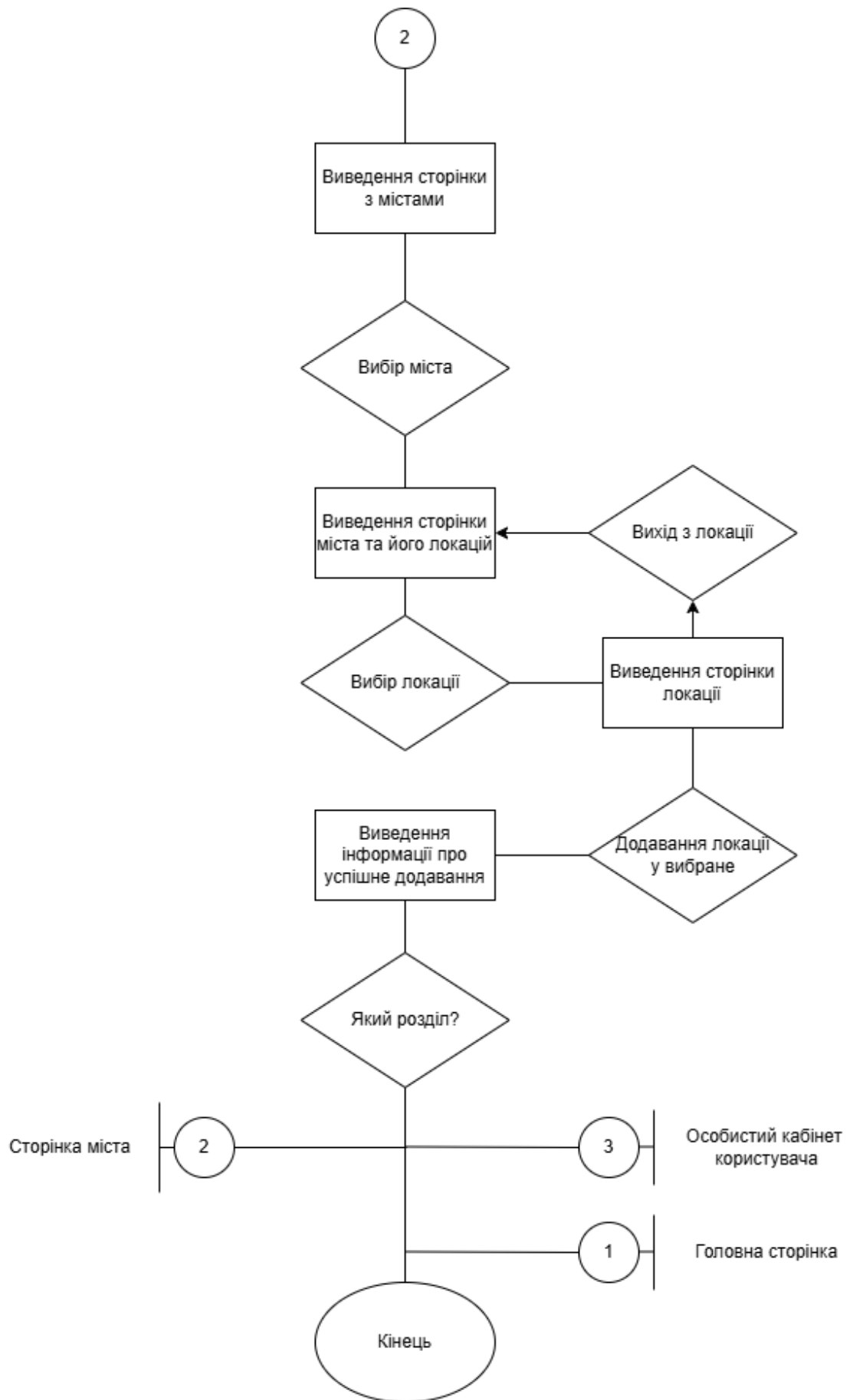


Рисунок 2.6 – Алгоритм – Модуль «Сторінка міста»

Джерело: Створено автором

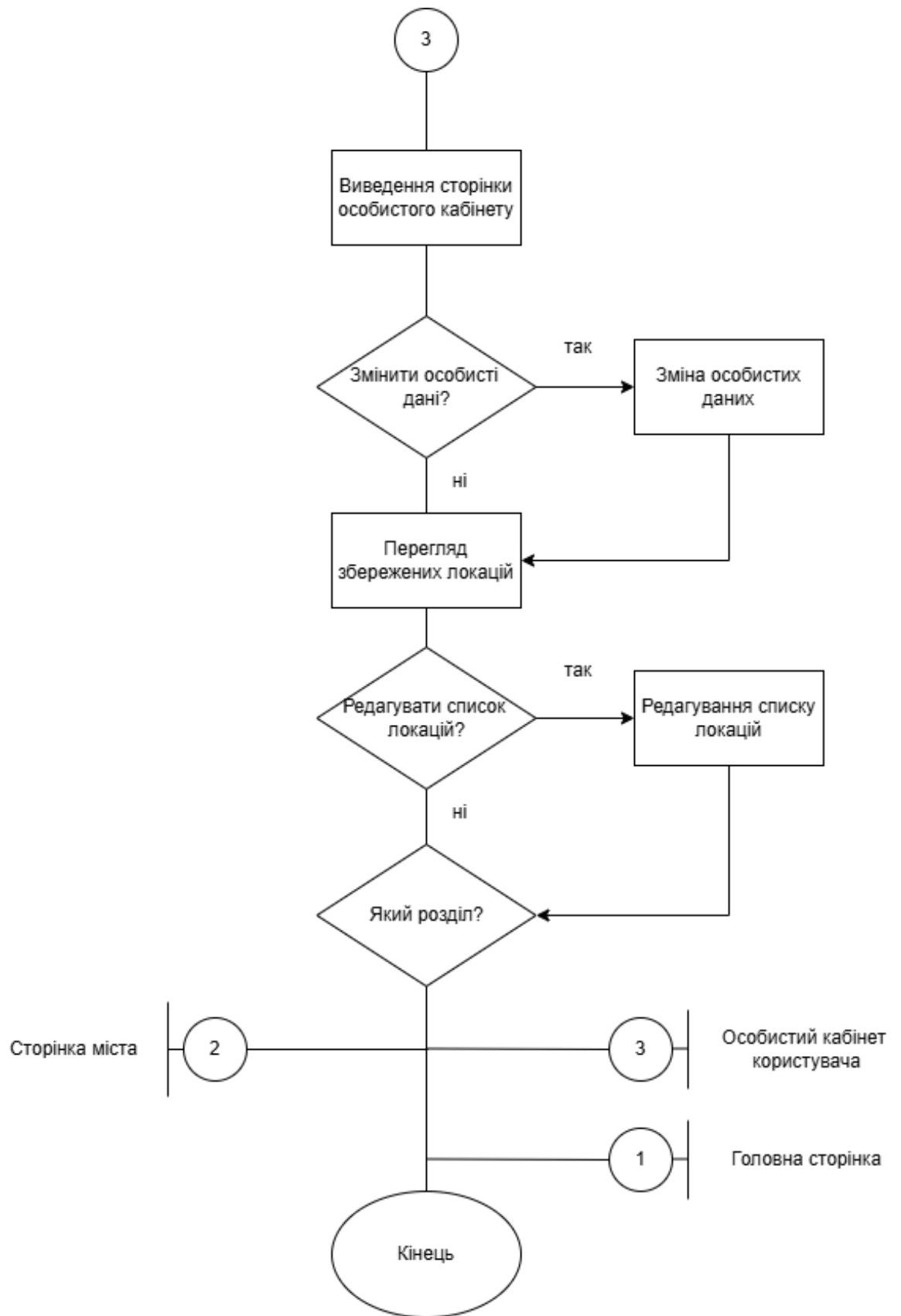


Рисунок 2.7 – Алгоритм – Модуль «Особистий кабінет користувача»

Джерело: Створено автором

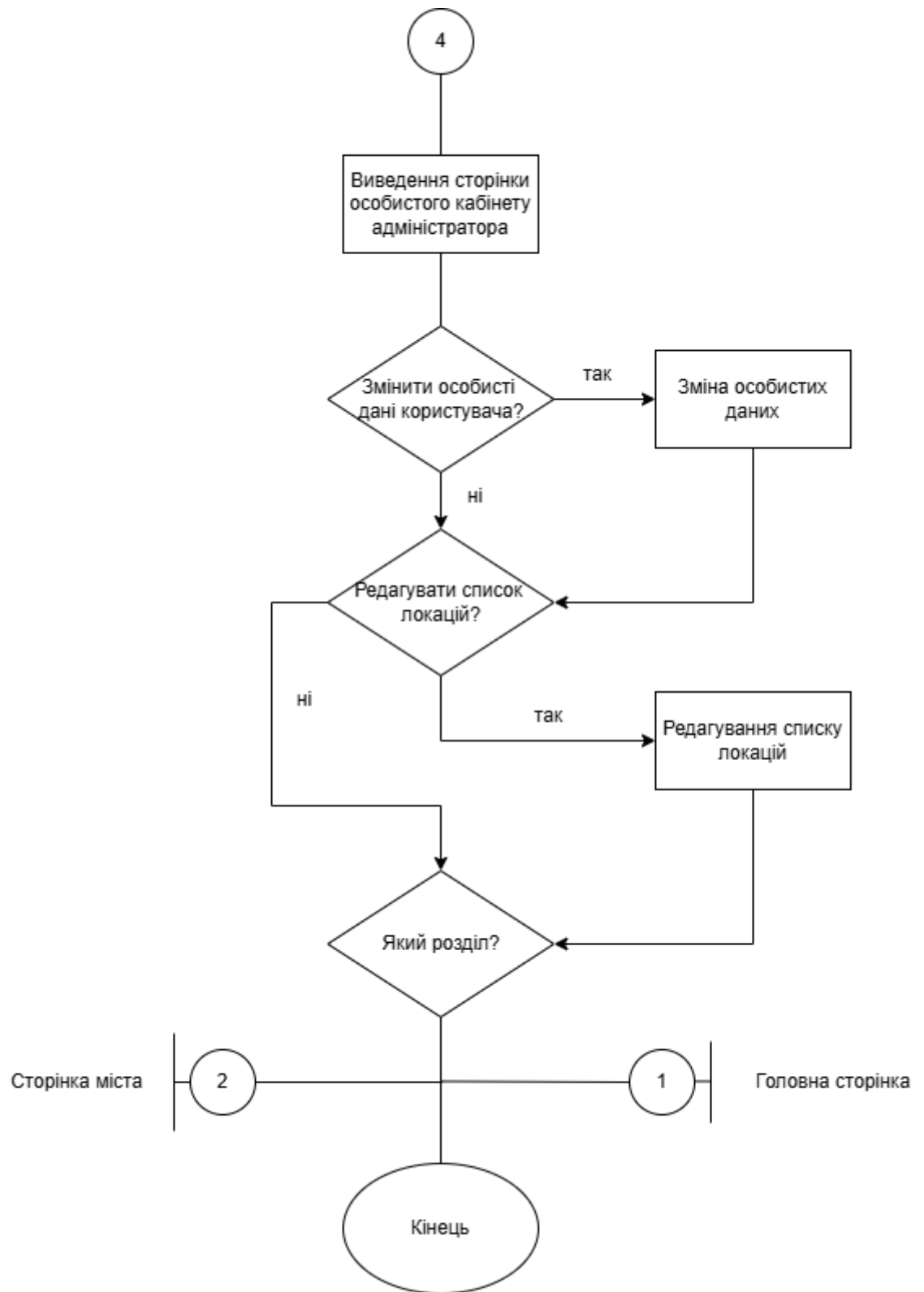


Рисунок 2.8 – Алгоритм – Модуль «Особистий кабінет адміністратора»

Джерело: Створено автором

2.3 Інформаційна модель підтримки управління подорожами

Задля реалізації моделювання інформаційної системи «Подорож містами України» було застосовано об'єктно-орієнтований підхід, що дозволяє ефективно структурувати та зрозуміти складну систему за допомогою абстрагування. При моделюванні використовувалося програмне середовище Enterprise Architect.

Діаграма варіантів використання (Use Case Diagram) є графічним представленням основних сценаріїв взаємодії користувачів із системою. На рис. 2.9 представлено модель поведінки системи підтримки управління подорожами у вигляді діаграми прецедентів.

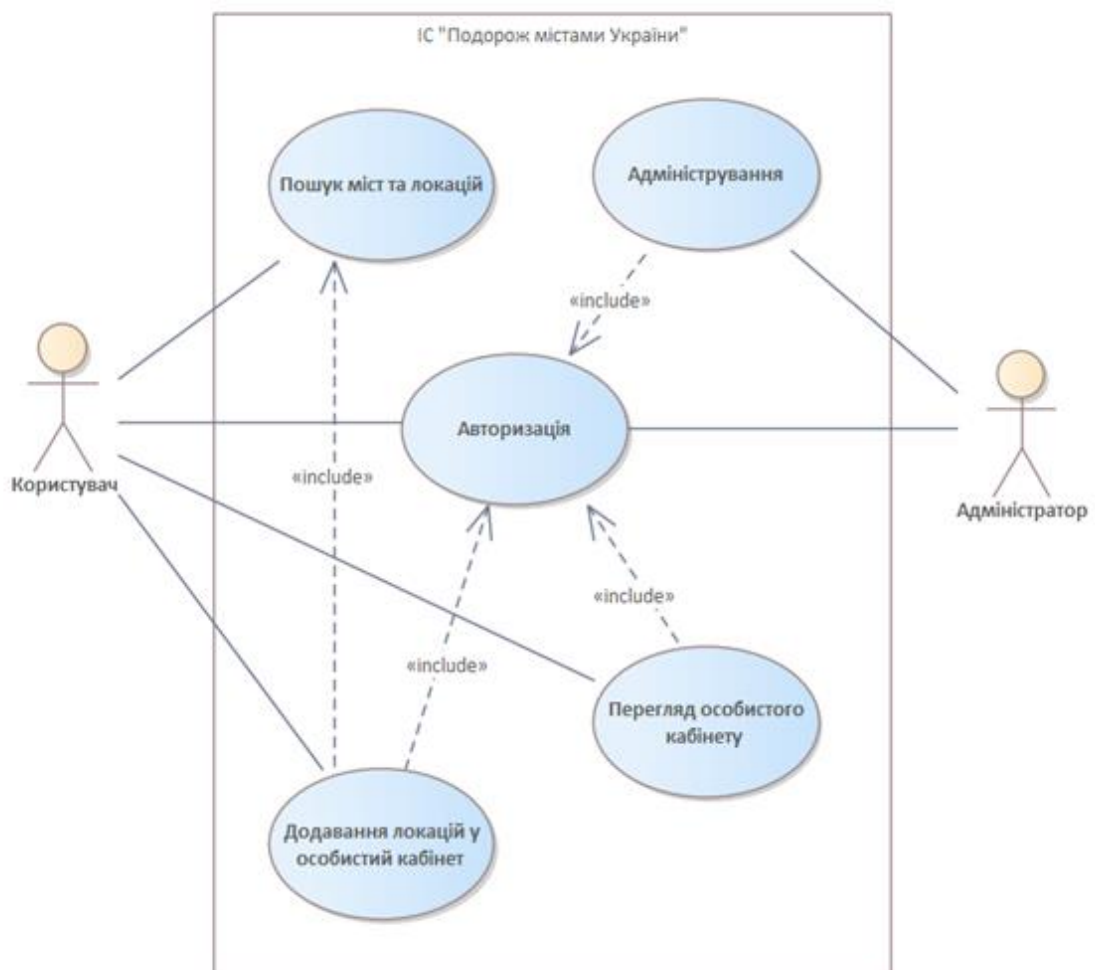


Рисунок 2.9 – Діаграма прецедентів ІС «Подорож містами України»

Джерело: Створено автором

У розробленій інформаційній системі визначено два основних актори - Користувач та Адміністратор. Користувач взаємодіє з системою з метою пошуку міст та локацій, авторизації для доступу до персоналізованого функціоналу, додавання локацій у особистий кабінет, а також перегляду особистого кабінету. Адміністратор, у свою чергу, проходить авторизацію та здійснює адміністративні дії щодо управління базою даних локацій та користувачів.

Отже, ключовим етапом для обох типів користувачів є авторизація. Для користувача це відкриває доступ до персоналізованих сервісів системи, а для адміністратора - до розширених можливостей адміністрування.

Діаграма послідовності (Sequence Diagram) моделює динамічну поведінку системи у відповідь на конкретні сценарії використання. Було створено відповідні діаграми послідовності для основних сценаріїв системи.

На рис. 2.10 подано процес перегляду та додавання локацій у збережене. Користувач відкриває інформаційну систему та отримує список доступних міст. Після вибору певного міста формується список локацій у вибраному місті. Користувач має можливість обрати бажану локацію та додати її у власний профіль. Цей процес супроводжується послідовною взаємодією між UI, контролером, сервісом та базою даних, де інформація про додану локацію зберігається у профілі користувача.

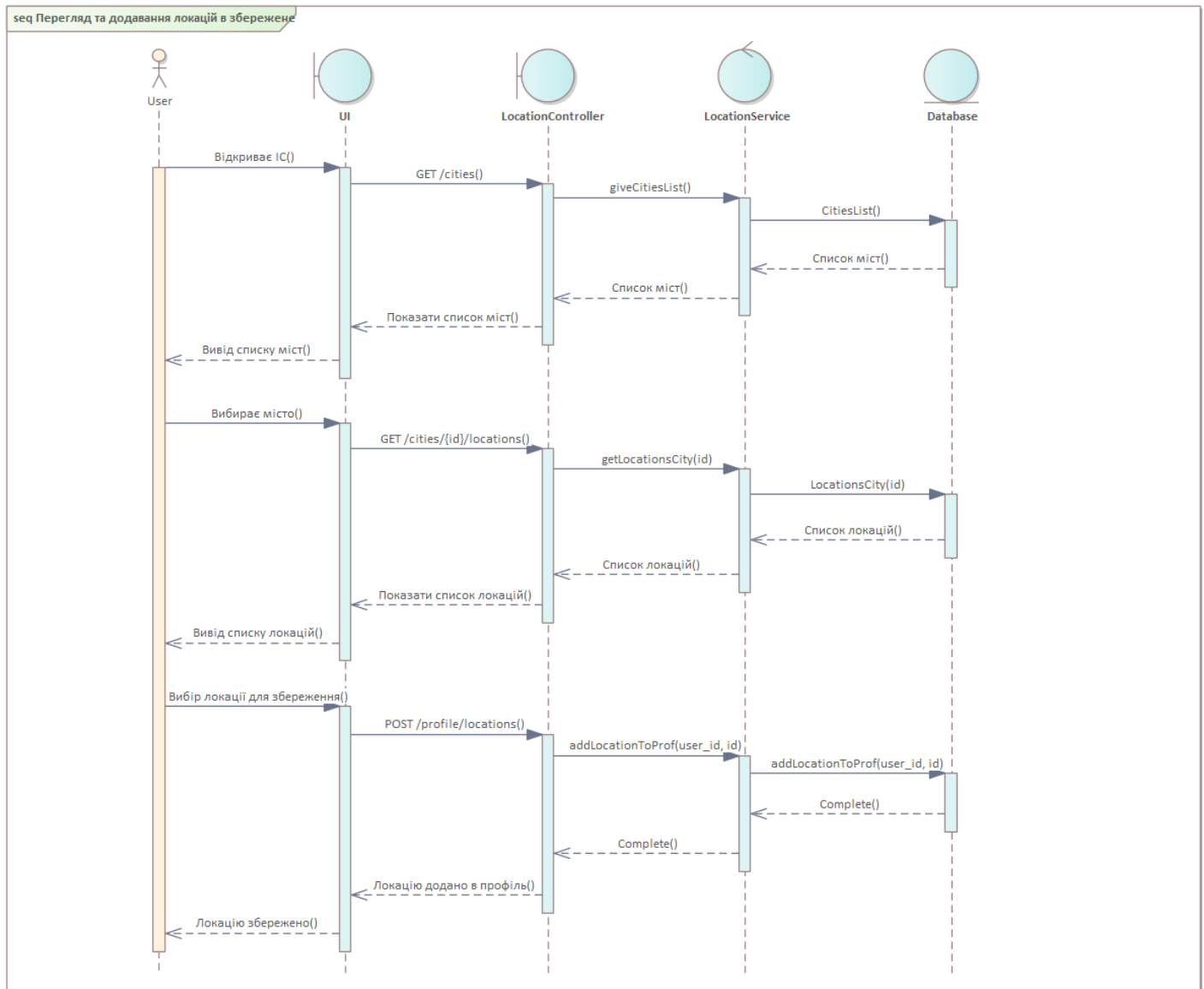


Рисунок 2.10 – Діаграма послідовності – Перегляд та додавання локацій в збережене

Джерело: Створено автором

Процес аутентифікації користувачів та доступу до особистого кабінету подано на рис. 2.11 Користувач може пройти реєстрацію, ввівши відповідні дані, які перевіряються і зберігаються у базі даних. Після цього користувач має змогу здійснити авторизацію, ввівши логін та пароль. Система проводить перевірку введених даних і у разі успіху надає доступ до особистого кабінету. При відкритті особистого кабінету система завантажує персональні дані користувача з бази даних та відображає їх у відповідному інтерфейсі.

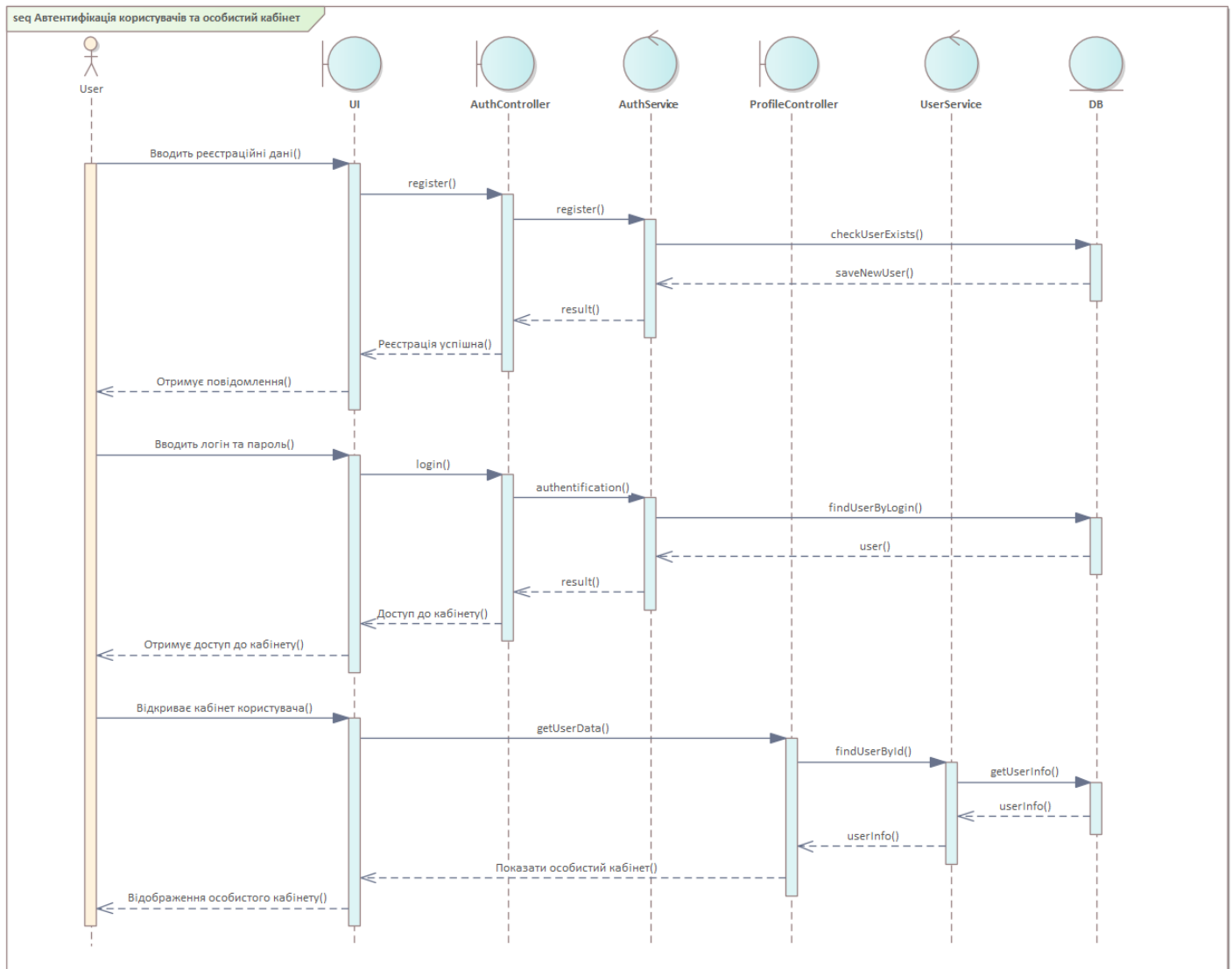


Рисунок 2.11 – Діаграма послідовності – Автентифікація користувачів та особистий кабінет

Джерело: Створено автором

На рис. 2.12 подано процес аутентифікації адміністратора та здійснення адміністративного управління. Адміністратор виконує авторизацію за допомогою логіна та пароля. Після успішної перевірки він отримує доступ до адміністративної панелі. Адміністратор може переглядати та редагувати повні списки локацій, міст та користувачів. Усі зміни, що вносяться адміністратором, фіксуються у базі даних та супроводжуються повідомленням про успішне оновлення.

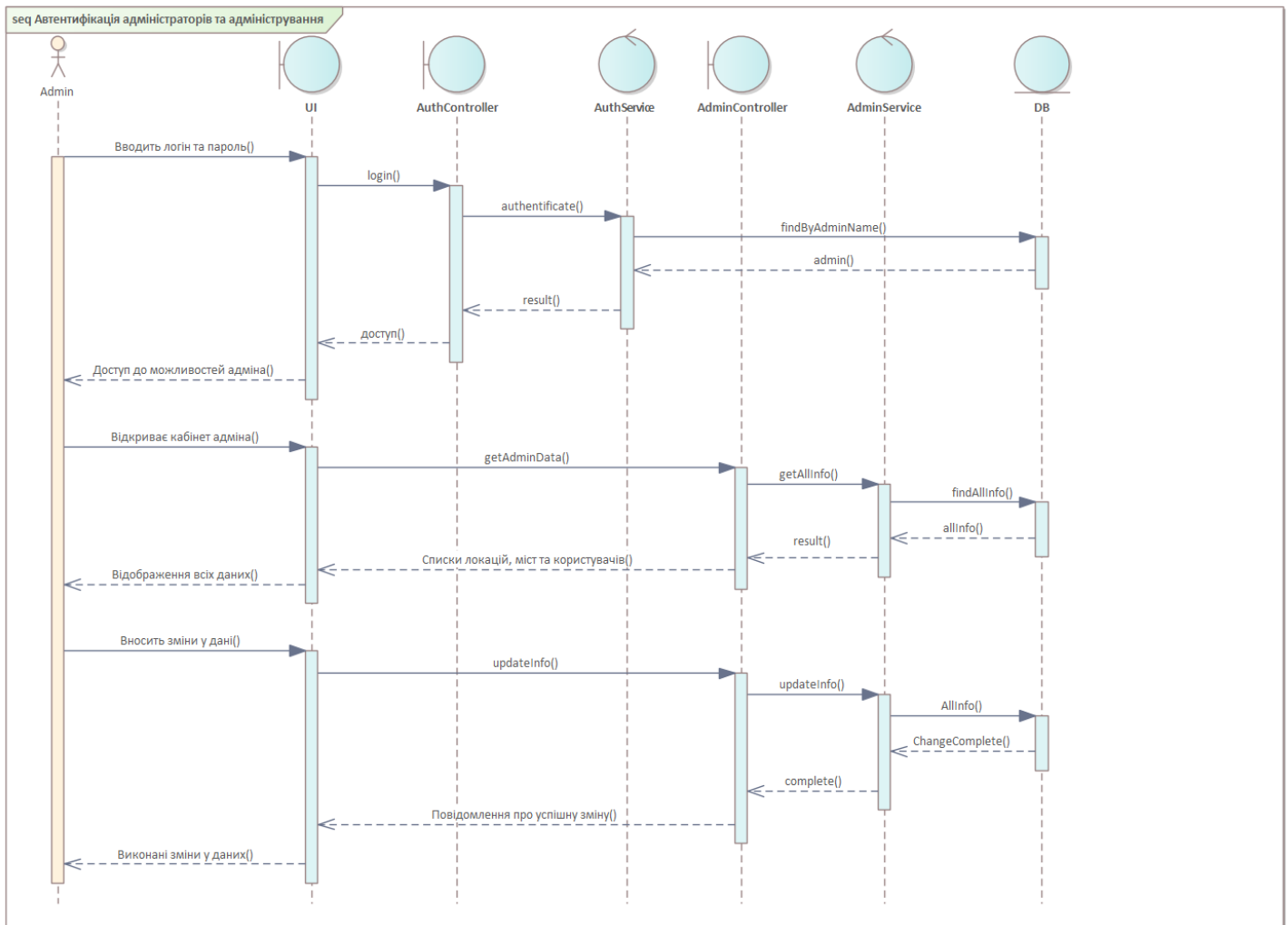


Рисунок 2.12 – Діаграма послідовності – Автентифікація адміністраторів та адміністрування

Джерело: Створено автором

Таким чином, побудовані діаграми прецедентів та послідовності демонструють ключові сценарії використання інформаційної системи «Подорож містами України», забезпечуючи наочне уявлення про взаємодію користувачів і адміністратора з системою та її динамічну поведінку.

У процесі моделювання структури системи було створено *логічну модель бази даних* (рис. 2.13), яка відображає основні сутності інформаційної системи «Подорож містами України», їх атрибути та взаємозв'язки між ними. Модель бази даних побудована з урахуванням принципів реляційної структури, що забезпечує цілісність та узгодженість даних.

До ключових сутностей системи належать: користувачі, туристичні локації, міста та збережені локації, які дозволяють зберігати обрані користувачем місця для подальшого перегляду або формування маршрутів. Усі зв'язки між таблицями реалізовані за допомогою зовнішніх ключів, що дає змогу підтримувати зв'язність даних на рівні схеми.

Такий підхід дозволяє ефективно реалізувати основну функціональність системи - реєстрацію користувачів, організацію довідника міст і локацій, збереження обраних місць у персональному кабінеті та швидкий пошук інформації. Структура бази даних є гнучкою для розширення та дозволяє легко інтегрувати додаткові функції, такі як категоризація локацій, рейтинги або відгуки.

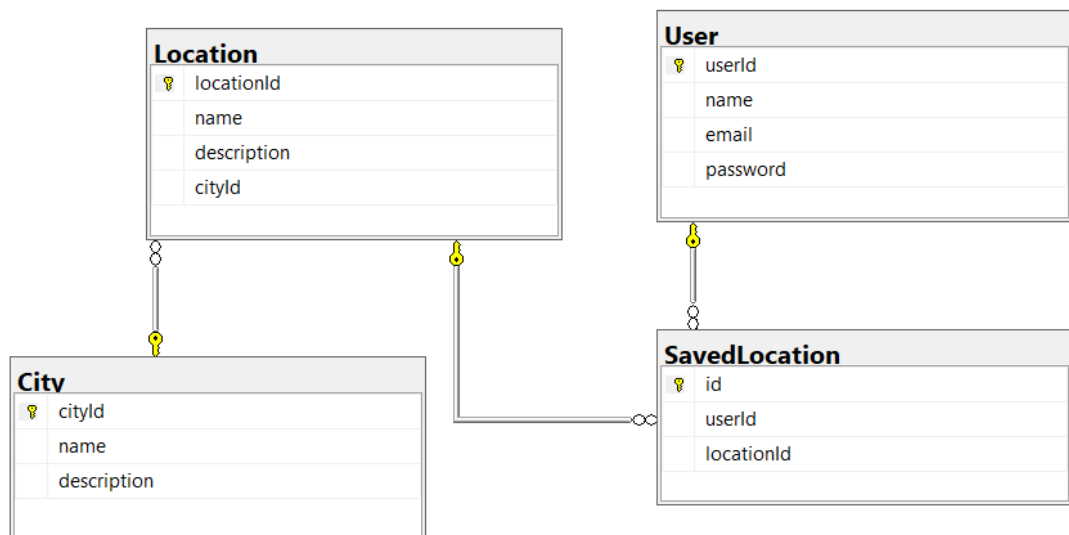


Рисунок 2.13 – Модель бази даних у SQL Server Management Studio

Джерело: Створено автором

На основі розробленої моделі бази даних у середовищі SQL Server Management Studio у Enterprise Architect було створено діаграму класів, яка відображає логічну структуру основних сутностей інформаційної системи «Подорож містами України» (див. рис. 2.14).

Центральне місце в моделі займає клас User, який містить ключову інформацію про користувача: його унікальний ідентифікатор, ім'я, електронну

адресу та зашифрований пароль. Цей клас пов'язаний з проміжною сутністю SavedLocation, що реалізує зв'язок між користувачем та туристичними локаціями, які він додав до свого особистого кабінету.

Клас SavedLocation відіграє роль зв'язувальної таблиці та містить поля для збереження ідентифікатора запису, посилань на відповідного користувача (userId) та локацію (locationId), яку було додано до збережених. Це дозволяє реалізувати механізм індивідуального вибору та збереження цікавих об'єктів для подорожей.

Клас Location містить основні характеристики туристичної локації: ідентифікатор, назву, опис та посилання на місто, до якого вона належить. Зв'язок між локацією та містом забезпечується через зовнішній ключ cityId, що вказує на відповідний запис у класі City.

Клас City описує населені пункти, де розміщені туристичні об'єкти, і включає унікальний ідентифікатор, назву та текстовий опис. Така структура дозволяє ефективно організувати пошук локацій за містами, реалізувати фільтрацію та навігацію, а також розширити функціональність системи в майбутньому - наприклад, за рахунок введення категорій, рейтингу або відгуків.

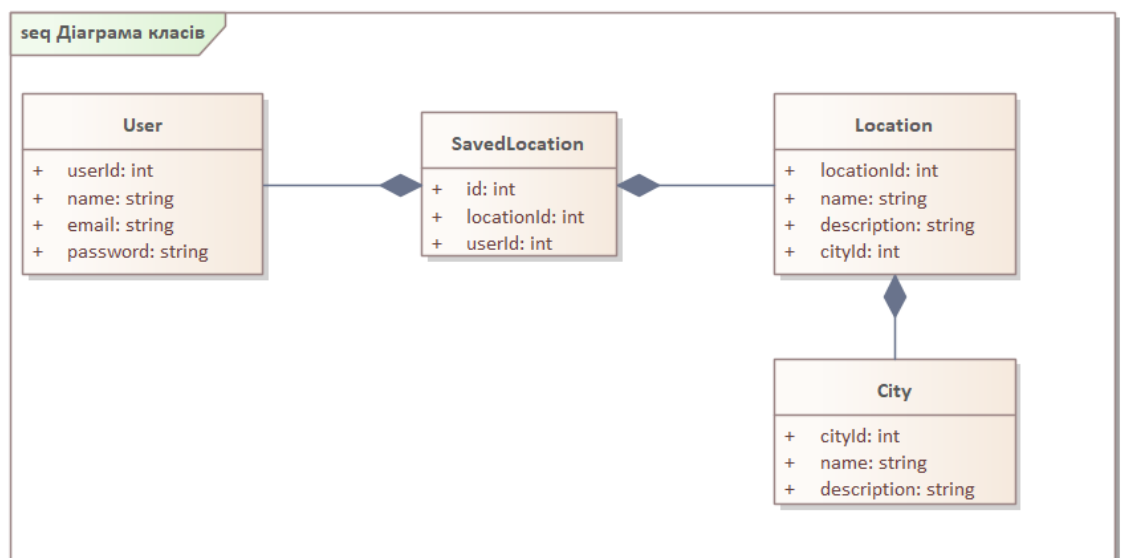


Рисунок 2.14 – Діаграма класів

Джерело: Створено автором

Для представлення внутрішньої архітектури інформаційної системи «Подорож містами України» було побудовано діаграму визначення блоків (Block Definition Diagram, BDD), яка відображає структуру системи у вигляді ієрархії взаємопов'язаних компонентів та дозволяє візуалізувати її логічну та технічну побудову (див. рис. 2.15). Така діаграма дозволяє окреслити склад системи на концептуальному рівні та показати, з яких частин вона складається, як між собою взаємодіють її окремі функціональні елементи та що саме входить до складу кожної з підсистем.

Центральним компонентом структури виступає блок «ІС 'Подорож містами України'», що включає три ключові складові: базу даних, програмне забезпечення та технічне забезпечення. Блок бази даних відповідає за збереження всієї критично важливої інформації системи. У його склад входять окремі частини, які зберігають дані про користувачів, міста, локації, а також зв'язки між користувачами та збереженими локаціями, а також окремий блок даних для адміністраторів. Такий розподіл дає змогу ізолювати різні види інформації та підтримувати структурну впорядкованість у сховищі.

Блок програмного забезпечення охоплює логіку функціонування системи та обробку взаємодій з користувачем. До нього належать сервіси отримання даних, інтерфейс користувача та сервіс особистого кабінету. Ці компоненти забезпечують збирання та передачу даних між користувачем і базою даних, надають доступ до збережених локацій, а також реалізують загальну бізнес-логіку вебзастосунку.

Технічне забезпечення включає сервер бази даних і вебсервер. Ці елементи забезпечують технічну інфраструктуру для роботи системи, включаючи фізичне зберігання даних, доступність інтерфейсу, приймання та обробку клієнтських запитів. Така структура є масштабованою, зручною для супроводу та модернізації, а також дає змогу розмежувати логічні, прикладні й інфраструктурні рівні системи, що відповідає сучасним принципам побудови інформаційних систем.

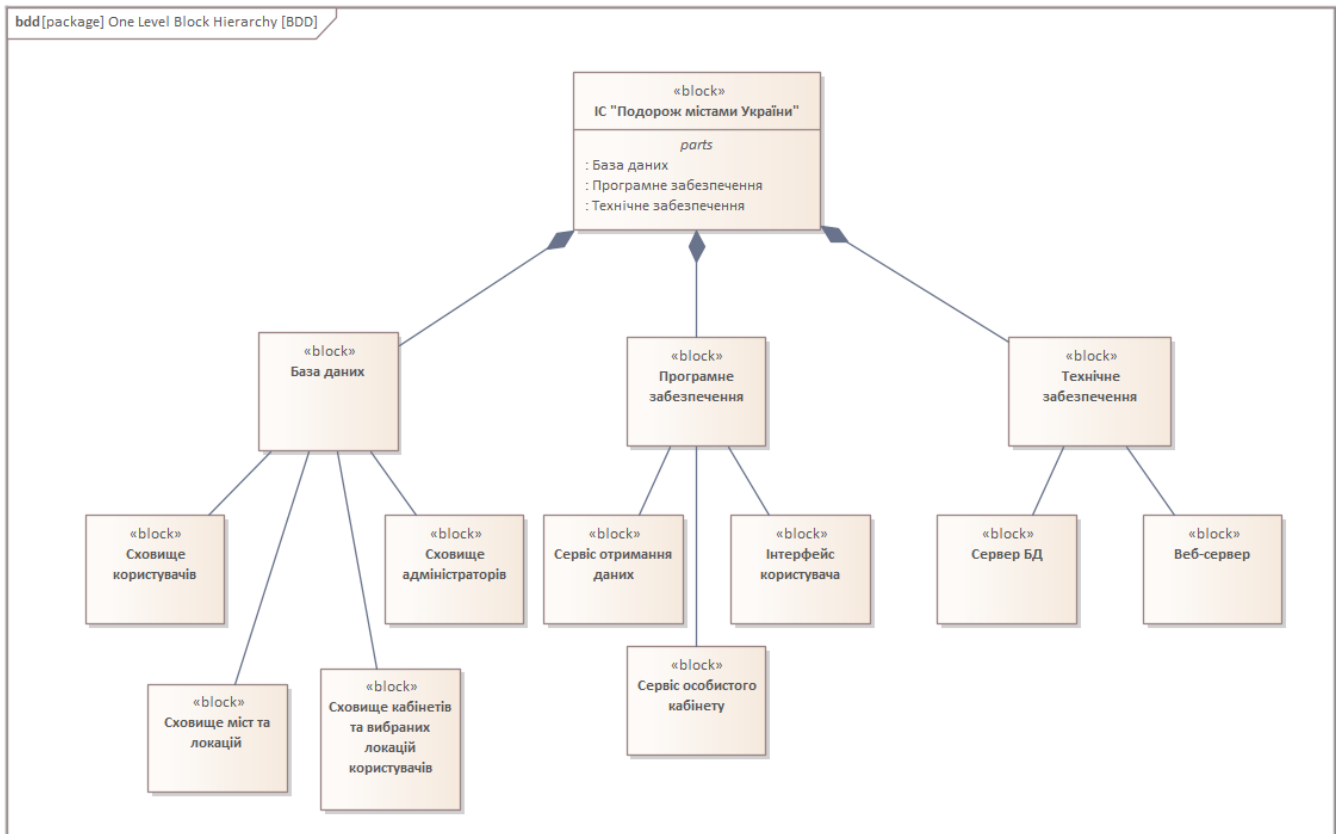


Рисунок 2.15 – Діаграма визначення блоків

Джерело: Створено автором

На рисунку 2.16 представлено діаграму внутрішніх блоків (IBD - Internal Block Diagram), яка відображає структурну взаємодію між основними компонентами інформаційної системи «Подорож містами України». Ця діаграма демонструє архітектуру на рівні взаємозв'язків між блоками та показує, яким чином елементи системи обмінюються даними та реалізують спільну функціональність через інтерфейси, визначені портами.

Центральним елементом виступає блок «ІС «Подорож містами України»», який взаємодіє з трьома ключовими підсистемами: програмним забезпеченням, базою даних та технічним забезпеченням. Кожен із компонентів підключено до головного блоку через специфіковані порти, які ілюструють напрямки зв'язку та логіку обміну даними між частинами системи.

ПЗ взаємодіє з інформаційною системою через порти uAccess та fileAccess, які відповідають за логіку доступу користувача та обмін даними відповідно. Також передбачено порти dbAccess та filePort, що забезпечують підключення до бази даних та передачу інформації. Внутрішній зв'язок між програмним забезпеченням та базою даних здійснюється через порти dataPort і specPort, що відповідають за прямий доступ до структурованих даних.

Блок технічного забезпечення з'єднаний із центральною системою за допомогою порту sensorPort, який комунікує із зовнішнім інтерфейсом sensorConnection. Це забезпечує апаратне середовище для функціонування системи та дозволяє зберігати дані на сервері, підтримувати обробку запитів та реалізувати надійне з'єднання між клієнтом і системними ресурсами.

Уся модель побудована з урахуванням принципу розділення відповідальності між підсистемами, де кожен блок виконує чітко визначену роль, а передача інформації між компонентами організована через стандартизовані порти. Такий підхід дозволяє досягти масштабованості, гнучкості та зручності у подальшій підтримці системи.

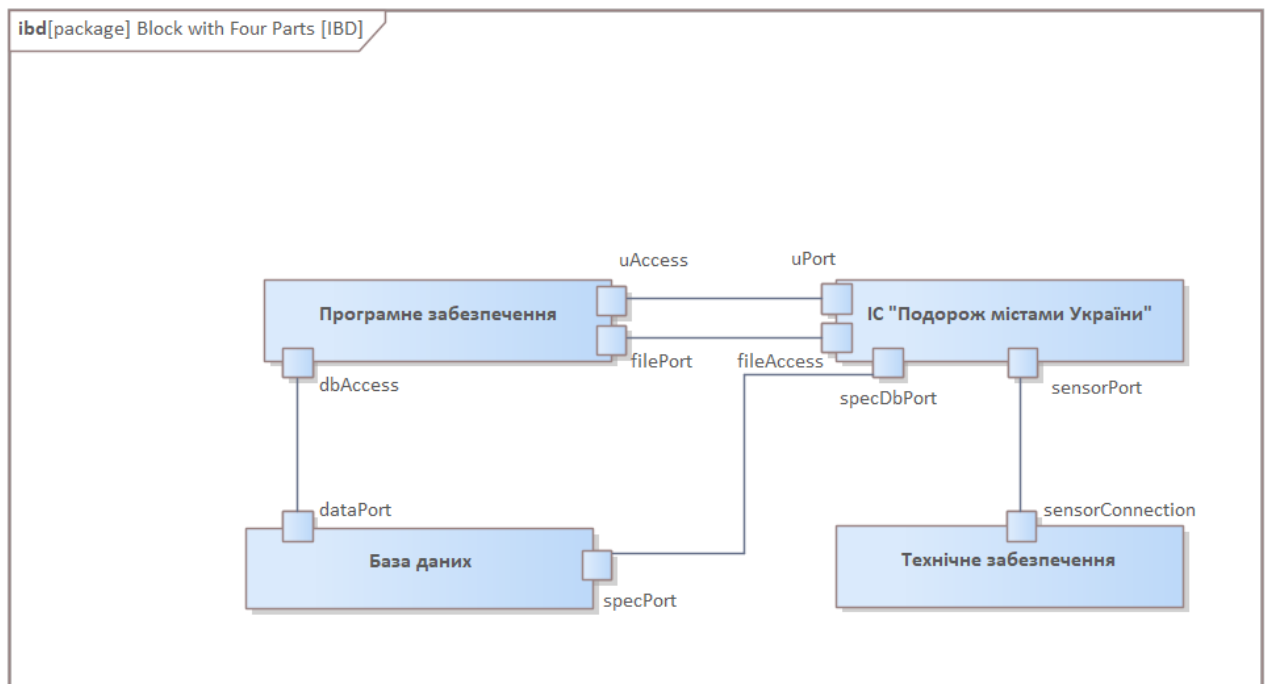


Рисунок 2.16 – Діаграма внутрішніх блоків

Джерело: Створено автором

Трасування вимог є невід’ємною складовою процесу проєктування інформаційних систем, яка забезпечує узгодженість між вимогами замовника, реалізованою функціональністю, структурою системи, користувацькими інтерфейсами та відповідними сценаріями тестування. Основною метою трасування є перевірка того, щоб кожна функціональна вимога була реалізована у конкретному елементі системи та підлягала подальшому тестуванню. Це гарантує прозорість та контроль якості на всіх етапах життєвого циклу програмного забезпечення.

У межах реалізації інформаційної системи «Подорож містами України» було створено дві діаграми трасування, які відображають зв’язки між основними складовими проєкту: вимогами, прецедентами, класами, UI-компонентами та тест-кейсами. На першій діаграмі трасування (рис. А.1 в Додатку А) зосереджено увагу на зв’язках між підсистемами, варіантами використання та класами, що беруть участь у реалізації функціональності. Наприклад, прецеденти «Авторизація», «Перегляд особистого кабінету», «Пошук міст та локацій» чи «Адміністрування» логічно пов’язані з відповідними класами, такими як User, SavedLocation, Location, City. Це дозволяє візуально прослідкувати, які саме об’єкти системи залучені до реалізації кожного use case, і переконатися в повному покритті вимог через об’єктну модель.

На другій діаграмі трасування (рис. А.2 в додатку А) було деталізовано та розширено зв’язки шляхом включення користувацьких інтерфейсів і тест-кейсів. Така структура дозволяє визначити, які саме UI-сторінки реалізують відповідні функціональні сценарії та яким чином ці сценарії перевіряються під час тестування. Наприклад, прецедент «Додавання локації у особистий кабінет» пов’язаний з інтерфейсом «UI Додавання локації» та перевіряється тест-кейсом TestCase2. Аналогічно, сценарій «Авторизація» реалізований через сторінку «UI Авторизація» і покритий тест-кейсом TestCase4. Таким чином, для кожної бізнес-вимоги можна простежити повний шлях: від вимоги

- до сценарію використання, до користувацького інтерфейсу - і до відповідного тесту.

Подібна модель трасування забезпечує логічну цілісність, структурну узгодженість і формалізовану перевірку кожного елементу системи, що особливо важливо при масштабуванні, супроводі та розширенні проєкту.

Моделювання користувацького інтерфейсу:

На рисунках 2.17–2.21 представлено діаграми користувацького інтерфейсу (UI) інформаційної системи «Подорож містами України», розроблені в середовищі Enterprise Architect. Макети демонструють основні екрани платформи, які забезпечують повноцінну взаємодію користувача з функціоналом системи - від ознайомлення з доступними містами та пам'ятками до управління персональним кабінетом або адмініструванням. Інтерфейс побудований у єдиному стилі, має просту структуру навігації, чітке розділення зон та елементів керування, що підвищує зручність користування.

На рисунку 2.17 зображено головну сторінку, яка виконує роль стартового екрана. Користувач бачить назву системи, поле для вибору міста, а також візуальні блоки для трьох міст - Київ, Львів та Одеса. Натискання на відповідну картку або пошук через рядок вибору відкриває доступ до переліку туристичних об'єктів у зазначеному місті. Верхня панель містить стандартні елементи - логотип, іконки входу та налаштувань. Нижня частина оформлена у вигляді гасла «Шануймо рідні міста!», що підкреслює ідею культурної ідентичності та національного туризму.

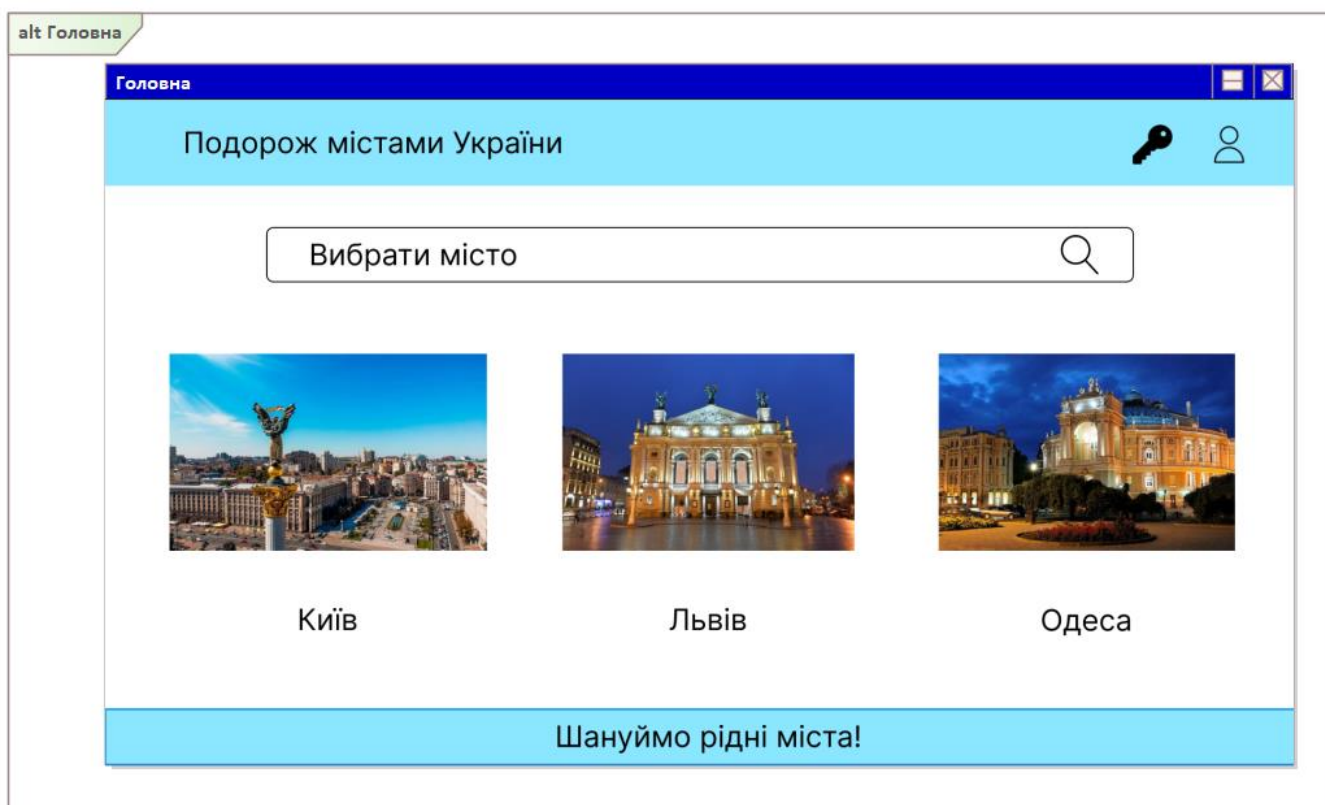


Рисунок 2.17 – Діаграма інтерфейсу головної сторінки

Джерело: Створено автором

Рисунок 2.18 відображає сторінку перегляду конкретної локації. У цьому прикладі - «Золоті ворота». Користувач бачить детальний опис об'єкта, посилання на карту, інформацію про вартість квитків та години роботи. Праворуч розміщено зображення пам'ятки, а під описом - кнопка «Додати в маршрут», яка дозволяє зберегти об'єкт до особистого списку. Інтерфейс побудований з акцентом на зручність ознайомлення з пам'яткою та швидке прийняття рішення щодо її додавання.

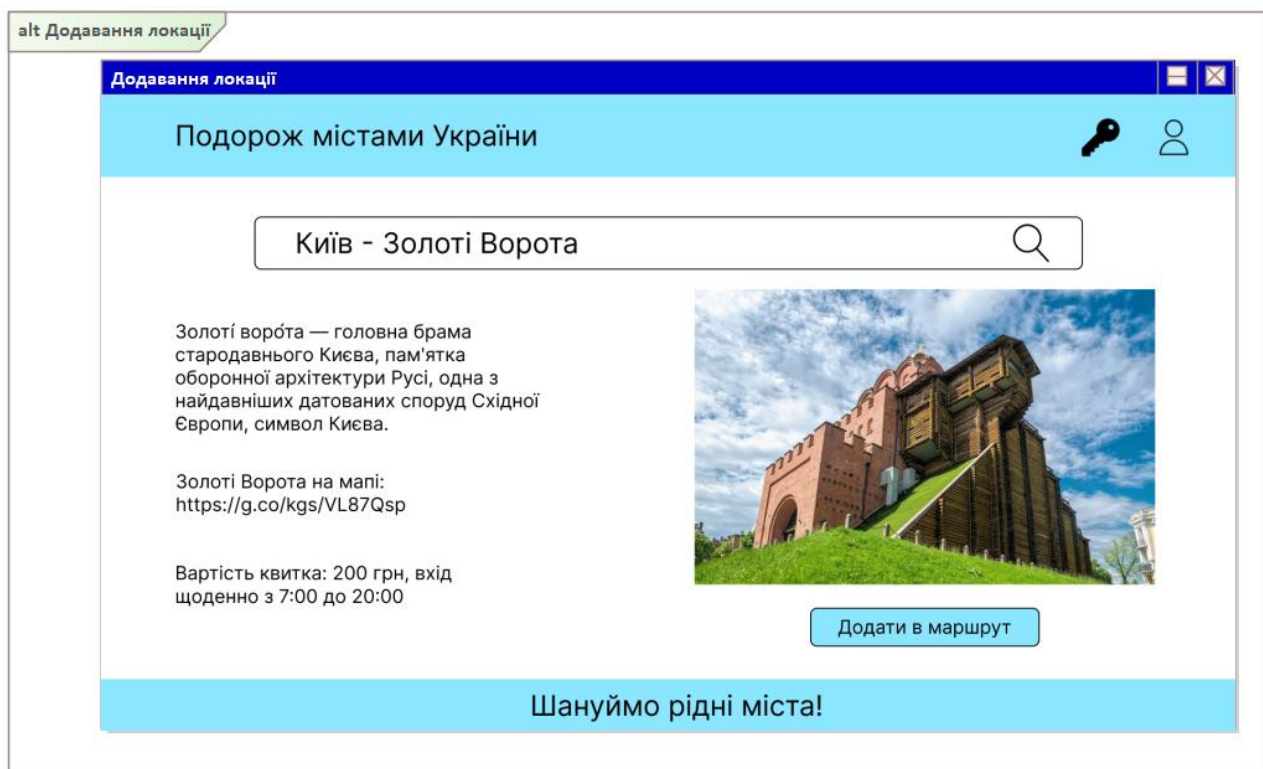


Рисунок 2.18 – Діаграма інтерфейсу сторінки додавання локації

Джерело: Створено автором

На рисунку 2.19 зображено екран авторизації користувача. Він містить два поля для введення електронної пошти та пароля, а також кнопки входу та переходу до реєстрації. Цей інтерфейс є важливою частиною системи, оскільки відкриває доступ до персоналізованого функціоналу, зокрема збереження обраних локацій і перегляду історії. Дизайн мінімалістичний, що дозволяє користувачу зосередитися на ключових діях без зайвих елементів.

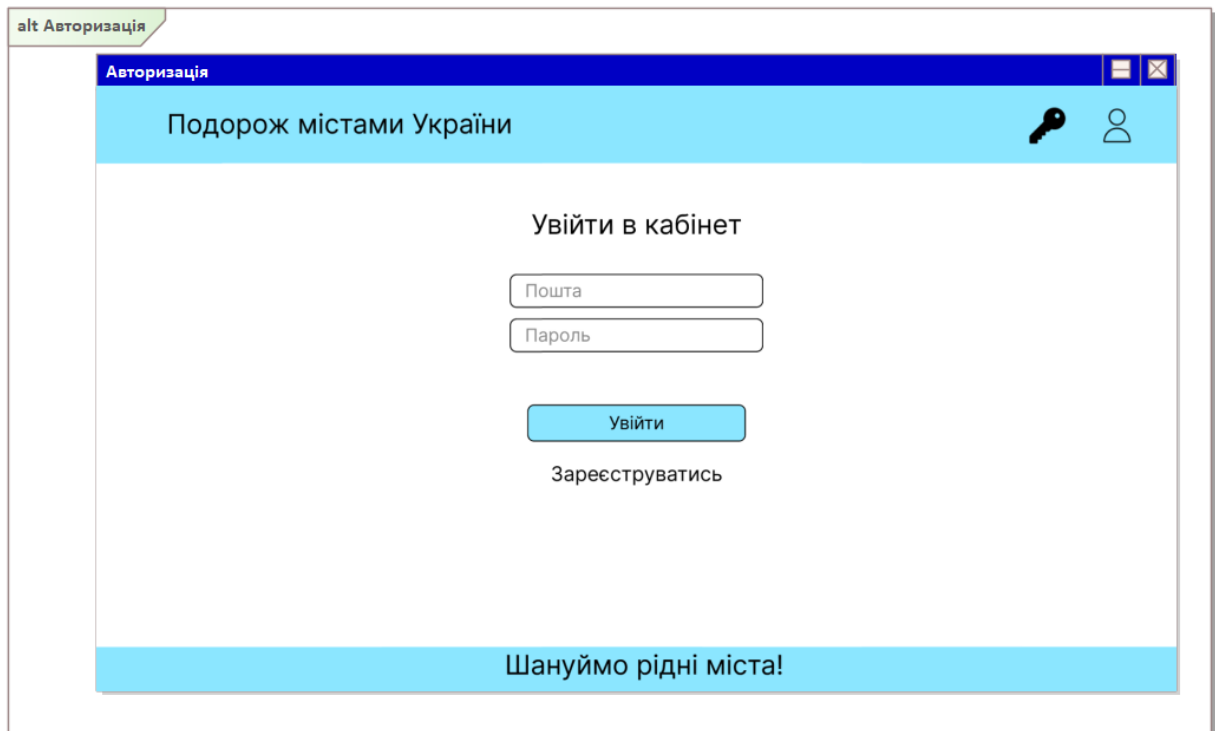


Рисунок 2.19 – Діаграма інтерфейсу сторінки авторизації

Джерело: Створено автором

Інтерфейс особистого кабінету користувача наведено на рисунку 2.20. У верхній частині сторінки зазначено, що це «Кабінет», далі поділено простір на дві частини - історія перегляду та збережені пам'ятки. У лівій частині знаходиться форма з персональними даними користувача (прізвище, ім'я, телефон), яку можна заповнити або редагувати. Праворуч - таблиці зі списками переглянутих і збережених локацій, а також кнопка для очищення історії. Цей інтерфейс дозволяє зручно організувати взаємодію користувача з його персональними даними та маршрутами.

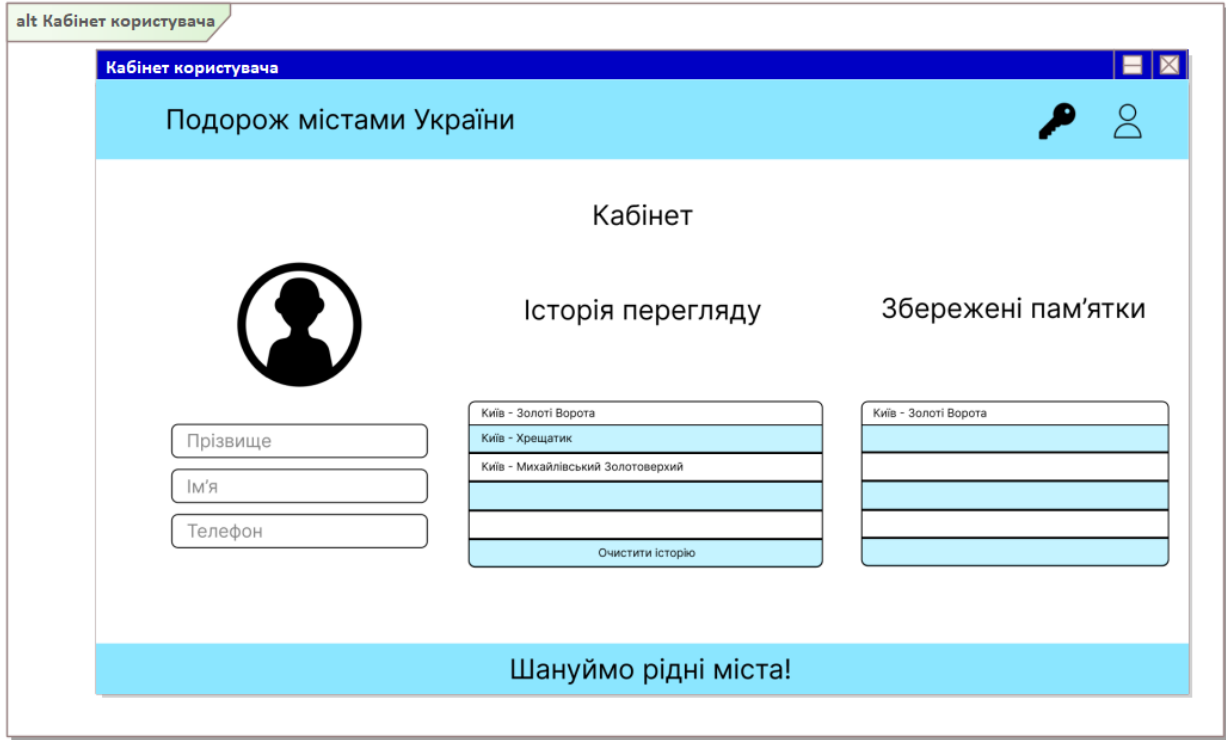


Рисунок 2.20 – Діаграма інтерфейсу сторінки кабінету користувача

Джерело: Створено автором

На рисунку 2.21 представлено сторінку адміністратора. Вона має таблиці, які дають змогу переглядати та редагувати дані про міста та користувачів. Зокрема, адміністратор може змінювати інформацію про наявні міста, а також переглядати дані користувачів і вносити необхідні корективи. У кожному рядку присутні іконки редагування, які дозволяють швидко переходити до редагування відповідного запису. Такий інтерфейс дозволяє оперативно управляти вмістом системи, актуалізувати дані та забезпечувати контроль за користувачами.

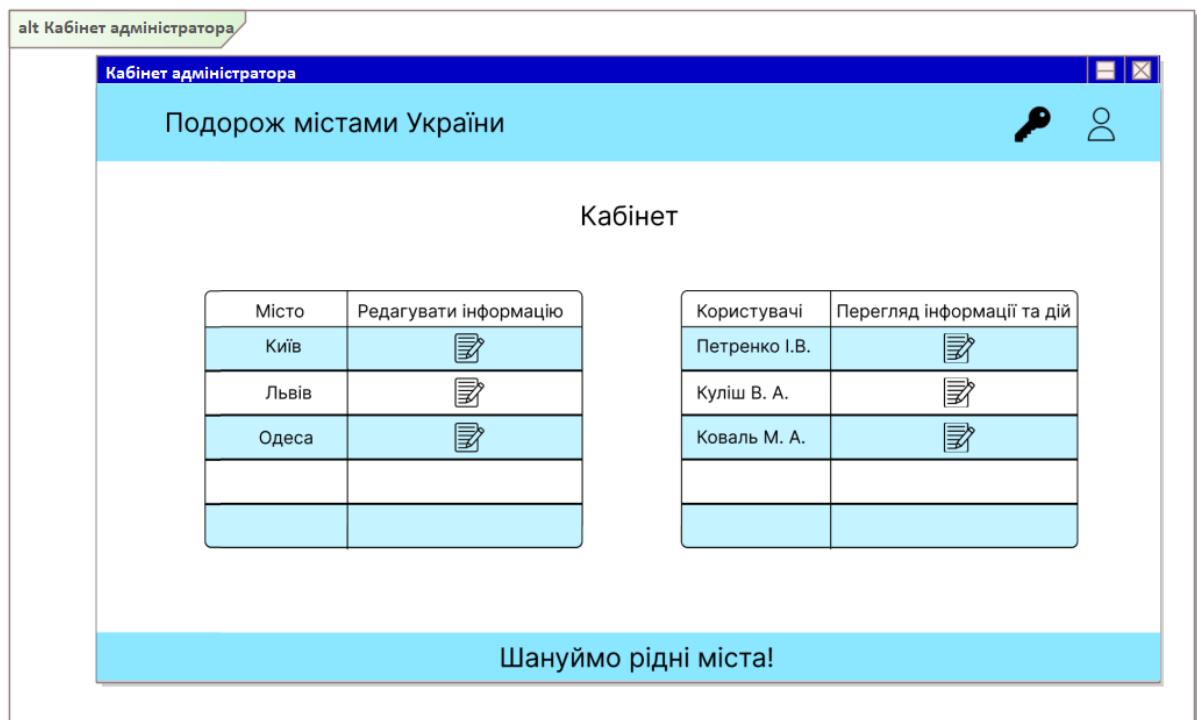


Рисунок 2.21 – Діаграма інтерфейсу сторінки кабінету адміністратора

Джерело: Створено автором

РОЗДІЛ 3

ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ

3.1 Інформаційне забезпечення

У даному підрозділі представлено загальну характеристику інформаційного забезпечення (ІЗ) інформаційної системи «Подорож містами України». Інформаційне забезпечення є основою для оброблення, зберігання та відображення даних у системі, забезпечує взаємодію між користувачами та системою і підтримує виконання бізнес-функцій.

У складі ІЗ системи передбачено інтерфейс користувача для введення та відображення інформації, а також база даних для зберігання інформації;

Структура інформаційного забезпечення включає *базу даних і елементи інтерфейсу користувача*.

База даних:

Для зберігання інформації у системі використовується реляційна база даних MySQL[16]. Структура БД побудована на основі розробленої інформаційної моделі задачі (див. підрозділ 2.2), а також діаграми класів (див. підрозділ 2.3).

Нижче наведено структуру інформаційних масивів, для кожного з них подано таблицю з її описом.

Опис масиву users

1. Найменування масиву: Користувачі
2. Ідентифікатор масиву: users.
3. Найменування носія інформації: сервер MySQL.
4. Максимальний об'єм масиву: без обмежень.
5. Довжина запису: 256 байтів.
6. Метод організації: послідовний.

Ключі упорядкування: userId.

Таблиця 3.1 –Масив користувачів

Найменування	Ідентифікатор у програмі	Умовне позначення	Формат	Первинний/вторинний ключ	Умова значення	Обов'язкове поле	Індексне поле	Логічні чи семантичні зв'язки
1	2	3	4	5	6	7	8	9
Ідентифікатор користувача	userId	Ui	int	PK	> 0	Так	ІНД	-
Ім'я користувача	name	Un	varchar(50)	-	-	Так	-	-
Електронна адреса	email	Ue	varchar(100)	-	NOT NULL	Так	ІНД	-
Пароль	password	Up	varchar(100)	-	NOT NULL	Так	-	-

Джерело: створено автором

Опис масиву cities

1. Найменування масиву: Міста.
2. Ідентифікатор масиву: cities.
3. Найменування носія інформації: сервер MySQL.
4. Максимальний об'єм масиву: без обмежень.
5. Довжина запису: 256 байтів.
6. Метод організації: послідовний.
7. Ключі упорядкування: cityId.

Таблиця 3.2 – Масив міст

Найменування	Ідентифікатор у програмі	Умовне позначення	Формат	Первинний/вторинний ключ	Умова значення	Обов'язкове поле	Індексне поле	Логічні чи семантичні зв'язки
1	2	3	4	5	6	7	8	9
Ідентифікатор міста	cityId	Ci	int	PK	> 0	Так	ІНД	-
Назва міста	cityName	Cn	varchar(100)	-	NOT NULL	Так	ІДД	-
Опис	cityDescription	Cd	text	-	-	Ні	-	-

Джерело: створено автором

Опис масиву locations

1. Найменування масиву: Локації.
2. Ідентифікатор масиву: locations.
3. Найменування носія інформації: сервер MySQL.
4. Максимальний об'єм масиву: без обмежень.
5. Довжина запису: 512 байтів.
6. Метод організації: послідовний.
7. Ключі упорядкування: locationId.

Таблиця 3.3 – Масив локацій

Найменування	Ідентифікатор у програмі	Умове позначення	Формат	Первинний/вторинний ключ	Умова значення	Обов'язкове поле	Індексне поле	Логічні чи семантичні зв'язки
1	2	3	4	5	6	7	8	9
Ідентифікатор локації	locationId	Li	int	PK	> 0	Так	ІНД	-
Назва локації	locationName	Ln	varchar(100)	-	NOT NULL	Так	ІДД	-
Опис	locationDescription	Ld	text	-	-	Ні	-	-
Місто	cityId	Ci	int	FK	> 0	Так	ІДД	cities

Джерело: створено автором

Опис масиву saved_locations

1. Найменування масиву: Збережені локації.
2. Ідентифікатор масиву: saved_locations.
3. Найменування носія інформації: сервер MySQL.
4. Максимальний об'єм масиву: без обмежень.
5. Довжина запису: 256 байтів.
6. Метод організації: послідовний.
7. Ключі упорядкування: id.

Таблиця 3.4 – Масив збережених локацій

Найменування	Ідентифікатор у програмі	Умове позначення	Формат	Первинний/вторинний ключ	Умова значення	Обов'язкове поле	Індексне поле	Логічні чи семантичні зв'язки
1	2	3	4	5	6	7	8	9
Ідентифікатор запису	id	Id	int	PK	> 0	Так	ІНД	-
Користувач	userId	Ui	int	FK	> 0	Так	ІДД	users
Локація	locationId	Li	int	FK	> 0	Так	ІДД	locations

Джерело: створено автором

Інтерфейс користувача:

Інтерфейс користувача представлено веб-інтерфейсом, побудованим за класичною архітектурою MVC (Model-View-Controller). Інтерфейс реалізовано за допомогою HTML[17], CSS[18] та JavaScript[19], серверна логіка - на базі сучасних фреймворків.

Основні елементи інтерфейсу користувача:

1) Сторінки системи:

- головна сторінка;
- сторінка авторизації;
- сторінка реєстрації;
- особистий кабінет користувача;
- сторінка перегляду списку міст;
- сторінка перегляду списку локацій;
- сторінка додавання локацій у профіль;
- сторінка адміністрування (для адміністратора).

2) Форми:

- форма входу;
- форма реєстрації;
- форма додавання локацій у профіль;
- форма редагування інформації у профілі;

- форма авторизації адміністратора.

3) Елементи навігації:

- кнопки переходу між сторінками;
- випадаючі меню для вибору міста / локації;
- модальні вікна для підтвердження дій.

Організація зберігання даних:

Для зберігання даних використовується сервер БД з електронним типом носія інформації (серверні SSD)[20], що дозволяє забезпечити високу швидкість доступу до даних та їхню надійність.

Принципи організації ІЗ:

Головними принципами організації інформаційного забезпечення є:

- Доступність - зручний та швидкий доступ користувачів до інформації у системі.
- Безпека - захист персональних даних користувачів та корпоративної інформації від несанкціонованого доступу.
- Масштабованість - можливість розширення БД та функціоналу без значних змін архітектури системи.
- Надійність - забезпечення стабільної роботи системи із мінімальним ризиком втрати даних.

Перелік джерел та носіїв інформації:

- Внутрішні джерела: адміністрація системи наповнює базу даних інформацією про нові міста та локації через веб-інтерфейс адміністрування.
- Зовнішні джерела: інформація про міста та локації може імпортуватися з відкритих джерел у вигляді XML/JSON-файлів.
- Інтерактивні дії користувачів: користувачі додають локації у свій профіль через інтерфейс системи. Усі ці дії автоматично фіксуються у відповідних таблицях БД.

Загальні вимоги до збору та передачі інформації:

- Точність і актуальність - збір і передача лише актуальної інформації.
- Безперервність оновлення - дані у БД оновлюються у режимі реального часу після відповідних дій користувачів або адміністратора.
- Безпека - використання механізмів авторизації та шифрування даних.
- Контроль якості даних - перевірка коректності введених даних на рівні інтерфейсу та на рівні БД.

Методи контролю інформації:

- Автентифікація користувачів - перевірка введених логіна та пароля для доступу до системи.
- Валідація даних - перевірка введених даних на коректність при кожному введенні у форми системи.
- Логування подій - запис системних подій (вхід/вихід користувачів, додавання локацій, редагування профілю) у лог-файли для забезпечення моніторингу та аудиту.

Кодування та ідентифікація об'єктів:

Для всіх таблиць БД використовується тип ідентифікаторів INT із автоматичним збільшенням (AUTO_INCREMENT). Це дозволяє гарантувати унікальність ідентифікаторів і забезпечує масштабованість при зростанні кількості записів.

Також було побудовано *даталогічну модель* бази даних для інформаційної системи «Подорож містами України» [21]. Модель створено у середовищі проєктування бази даних та відображає основні сутності системи, їх атрибути та взаємозв'язки між ними (рис. 3.1).

До складу моделі входять такі основні таблиці:

- User - зберігає інформацію про користувачів системи (ідентифікатор, ім'я, електронна пошта, пароль).

- City - зберігає довідник міст, включаючи їх ідентифікатор, назву та опис.
- Location - містить інформацію про локації (ідентифікатор, назву, опис, посилання на місто).
- SavedLocation - таблиця-зв'язок, яка фіксує збережені локації у профілях користувачів (ідентифікатор запису, посилання на користувача та локацію).

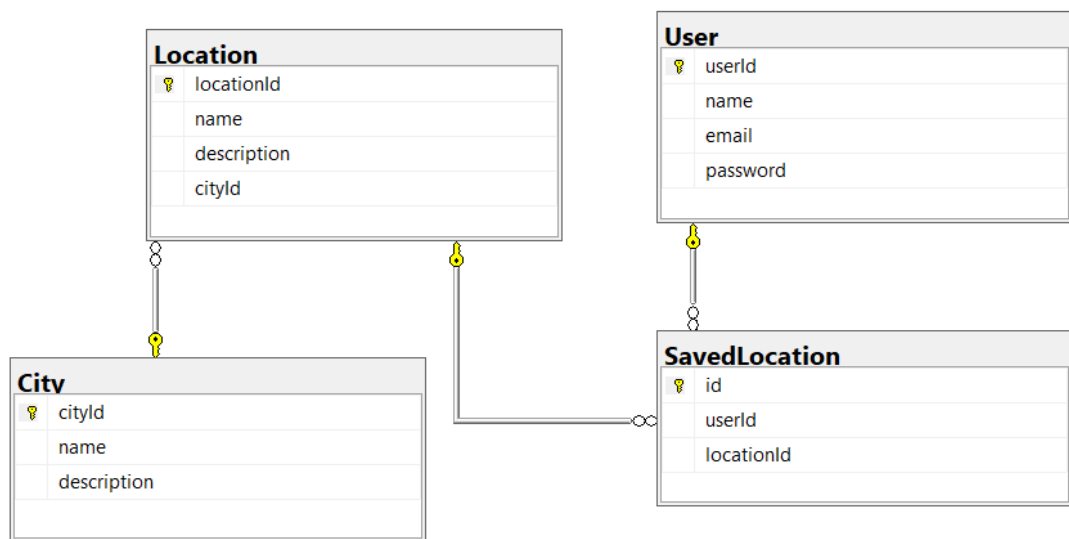


Рисунок 3.1 – Даталогічна модель бази даних

Джерело: Створено автором

Наявність чітко визначених зв'язків між таблицями забезпечує цілісність даних та підтримує основну бізнес-логіку системи: збереження локацій у профілі користувача, відображення списку міст та локацій, персоналізацію профілів користувачів.

У результаті, добре спроектована даталогічна модель бази даних не лише підтримує поточні вимоги застосунку, але й готова до майбутніх змін та доповнень, таких як розширення профілю користувача, додавання нових типів локацій або додаткових властивостей.

3.2 Технічне забезпечення

Для забезпечення ефективного функціонування інформаційної системи «Подорож містами України» важливо ретельно обґрунтувати та організувати технічне забезпечення системи. Технічне забезпечення включає комплекс технічних засобів (КТЗ), що складається з серверного обладнання, робочих станцій користувачів, мережевої інфраструктури та периферійних пристроїв. Головна мета технічного забезпечення - забезпечення стабільної, безпечної та високопродуктивної роботи веб-застосунку в умовах багатокористувацького режиму.

Інформаційна система «Подорож містами України» функціонуватиме на віддаленому сервері, що розміщений у захищеному дата-центрі або на орендованих потужностях хмарного провайдера. Серверна частина системи містить ядро веб-застосунку, базу даних та необхідні файли для забезпечення роботи інформаційної системи. Доступ користувачів до системи здійснюється за допомогою веб-інтерфейсу через Інтернет.

Користувачі підключаються до веб-сервера через Інтернет-мережу з використанням стандартного веб-браузера. Веб-сервер обробляє HTTP-запити, звертається до серверу бази даних для отримання необхідної інформації про міста, локації та збережені дані користувача. Структура взаємодії компонентів системи передбачає мінімальну кількість проміжних ланок для забезпечення швидкого доступу та мінімальної затримки при роботі з даними.

На стороні Back-office передбачена робота адміністратора системи, який здійснює управління довідниками міст та локацій, а також виконує перегляд та редагування даних користувачів. Розробники відповідають за підтримку та оновлення системи, технічна підтримка забезпечує обробку звернень користувачів та підтримання працездатності сервісу.

Технічне забезпечення системи передбачає використання серверних потужностей хостинг-провайдера з достатніми характеристиками для розміщення веб-застосунку та бази даних. При виборі платформи для розміщення системи враховувались такі критерії: висока надійність, стабільна робота, захист даних, можливість масштабування та відповідність сучасним вимогам безпеки.

Серверна інфраструктура повинна забезпечувати резервне копіювання даних для відновлення у разі збоїв. Передбачено виконання регулярних резервних копій, що зберігатимуться на окремих серверах або у хмарному сховищі.

Для забезпечення безпеки обладнання застосовується комплексний підхід, що включає фізичний захист серверів у дата-центрах, обмеження доступу до обладнання за допомогою систем ідентифікації персоналу, використання захищених каналів зв'язку для передавання конфіденційної інформації.

Для забезпечення потреб розробки та адміністрування системи використовується мінімальна кількість автоматизованих робочих місць (АРМ). Для кожного члена команди передбачається одне АРМ із наступними технічними характеристиками:

- Процесор: Intel Core i5 або еквівалент.
- Оперативна пам'ять: не менше 8 ГБ.
- Накопичувач: SSD обсягом не менше 256 ГБ.
- Мережеве підключення: Ethernet для стабільного з'єднання.

На АРМ встановлюється набір програмного забезпечення для розробки та адміністрування (системи керування базами даних, середовища розробки, інструменти моніторингу та тестування).

Для забезпечення ефективної роботи користувачів на стороні клієнта передбачено використання стандартних персональних комп'ютерів або мобільних пристроїв з доступом до Інтернету та сучасним веб-браузером (Google Chrome, Mozilla Firefox, Microsoft Edge тощо).

Кількість необхідних АРМ визначається виходячи з поточних потреб команди розробки та супроводу системи. Для невеликої системи «Подорож містами України» достатньо організувати 3-4 робочих місця для розробників, адміністратора та фахівця технічної підтримки.

Вимоги до периферійних пристроїв включають використання:

- Моніторів з роздільною здатністю не нижче Full HD.
- Ергономічної клавіатури та миші.
- Гарнітури для забезпечення голосового зв'язку при необхідності.

Обрані технічні засоби повністю відповідають вимогам системи та забезпечують зручність і ефективність роботи як на етапі розробки, так і під час експлуатації інформаційної системи «Подорож містами України».

3.3 Програмне забезпечення

Описано структуру програмного забезпечення (ПЗ), що використовується для розробки, тестування та забезпечення функціонування інформаційної системи "Подорож містами України".

У якості базового програмного забезпечення використано операційну систему Windows 10 [22], яка забезпечує управління апаратними ресурсами комп'ютера та створює середовище для виконання програм. Системне програмне забезпечення також включає драйвери та системні утиліти, які забезпечують коректну взаємодію апаратного та програмного забезпечення.

Для розробки, моделювання та управління даними використано наступне спеціалізоване програмне забезпечення:

- Draw.io [23] - для створення діаграм та схем, що ілюструють логіку системи та її структуру;

- Enterprise Architect [24] - для побудови UML-діаграм (діаграми прецедентів, послідовностей, класів), моделювання структури та бізнес-процесів системи;
- MySQL Server Management Studio [25] - для створення, управління та адміністрування бази даних, а також для виконання SQL-запитів;
- Visual Studio Code [26] - як основне середовище розробки для написання HTML, CSS та JavaScript коду;
- Figma [27] - для створення макетів інтерфейсу користувача (UI/UX дизайн), прототипування веб-сторінок.

Прикладне програмне забезпечення передбачає створення текстових сценаріїв (use case scenarios) для кожного ключового варіанта використання системи. Підготовка сценаріїв здійснювалася у середовищі Enterprise Architect та викладена у Додатку Б.

Нижче наведено приклади реалізованих сценаріїв для тестування інформаційної системи:

1) Пошук та вибір локації

Нормальний сценарій:

- А) Користувач відкриває головну сторінку ІС.
- Б) Користувач обирає місто зі списку доступних.
- В) Користувач обирає локацію у вибраному місті.

Очікуваний результат:

Система відображає відповідний список локацій для обраного міста. При виборі локації - відображає детальну інформацію про неї.

2) Додавання локації в профіль

Нормальний сценарій:

- А) Користувач натискає кнопку "Додати в маршрут".
- Б) Система перевіряє, чи користувач авторизований.

В) У разі авторизації - додає вибрану локацію до профілю користувача.

Очікуваний результат:

Система додає вибрану локацію в профіль користувача, повідомляючи про успішне збереження.

3) Перегляд кабінету користувача

Нормальний сценарій:

А) Користувач входить у свій обліковий запис.

Б) Переглядає свої збережені локації у кабінеті користувача.

Очікуваний результат:

Система перевіряє авторизацію та у разі успіху відображає персональні дані користувача та список збережених локацій.

4) Авторизація користувача

Нормальний сценарій:

А) Користувач відкриває сторінку авторизації.

Б) Вводить пошту та пароль.

В) Натискає кнопку "Увійти".

Очікуваний результат:

Система перевіряє наявність облікового запису та у разі успішної авторизації - відкриває особистий кабінет.

5) Авторизація адміністратора

Нормальний сценарій:

А) Адміністратор відкриває сторінку авторизації.

Б) Вводить пошту та пароль.

В) Натискає кнопку "Увійти".

Очікуваний результат:

Система підтверджує авторизацію адміністратора та відкриває інтерфейс керування.

6) Редагування локації

Нормальний сценарій:

- А) Адміністратор авторизується у системі.
- Б) Відкриває персональний кабінет.
- В) Переходить до списку міст та редагує потрібну локацію.

Очікуваний результат:

Система приймає зміни та оновлює інформацію у базі даних. Адміністратор отримує підтвердження про успішне оновлення.

Для забезпечення роботи комп'ютерної техніки та мережі було обрано наступне *системне програмне забезпечення*:

- Операційна система: Windows 10 - для забезпечення стабільної роботи серверів та робочих місць розробників.
- Інструменти моделювання: Draw.io, Enterprise Architect.
- Інструменти для роботи з базами даних: MySQL Server Management Studio.
- Технології для розробки веб-застосунку: HTML, CSS, JavaScript.
- Середовище розробки: Visual Studio Code.
- Інструменти для створення UI/UX дизайну: Figma.

Прикладне програмне забезпечення також включає:

- ПЗ загального призначення: Microsoft Office - для створення документації, звітів та супровідних матеріалів.
- Браузери: Google Chrome, Mozilla Firefox - для тестування веб-застосунку у різних середовищах.

Програмна документація включає такі відомості:

- Назва: Веб-застосунок "Подорож містами України".
- Розробник: Герасименко Максим Ігорович.
- Версія: 1.0.

- Склад ПЗ: Windows, Draw.io, Enterprise Architect, MySQL Server Management Studio, HTML, CSS, JavaScript, Figma, Visual Studio Code.
- Експлуатація: Регулярні оновлення, постійна підтримка розробника.

Призначення програмного забезпечення:

- Реалізація функціональності пошуку міст та локацій.
- Збереження обраних локацій у профілі користувача.
- Надання можливості адміністративного керування локаціями.
- Забезпечення швидкого доступу та зручного перегляду особистого кабінету.

3.4 Можливості та доцільність впровадження

У ході виконання даного кваліфікаційного бакалаврського проєкту було розроблено інформаційну систему "Подорож містами України", яка реалізує ключові функції для підтримки організації та планування індивідуальних подорожей користувачів по містах України. Оригінальними пропозиціями, що закладено в основу розробленої системи, стали:

- використання спрощеного інтерфейсу для пошуку міст та локацій з можливістю збереження маршрутів у персональному профілі;
- персоналізована робота з маршрутом (додавання/видалення локацій у профіль);
- автоматизоване відображення динамічної інформації за обраними критеріями (місто → список локацій);

- розмежування прав доступу між користувачем та адміністратором (редагування контенту можливе тільки адміністраторами через відповідний інтерфейс).

Виконано пілотну апробацію основного функціоналу ІС за допомогою тестових сценаріїв (див. підрозділ 3.3 та Додаток Б), що підтверджує працездатність розробленої системи на контрольному прикладі.

Практична цінність і потенційна ефективність:

На сучасному ринку представлено низку платформ для організації подорожей (Google Maps, TripAdvisor, Rome2Rio тощо), однак вони або не надають можливості створення персоналізованого маршруту з прив'язкою до профілю, або надмірно складні для користувачів, що шукають базові функції для планування подорожі містами України.

Порівняно з альтернативами, розроблена ІС має такі переваги:

- спеціалізована на українському туристичному контенті;
- забезпечує просту взаємодію без перевантаженого інтерфейсу;
- підтримує гнучке керування особистими маршрутами;
- дозволяє адмініструвати контент у реальному часі без потреби технічного втручання у БД.

Відзначено також високу повноту та достовірність отриманих результатів при тестуванні, оскільки система у всіх тест-кейсах коректно обробляла запити користувачів та змінювала відповідний стан БД. Потенційна похибка полягає у відсутності масштабного тестування під високим навантаженням, однак для цілей пілотного впровадження це не є критичним обмеженням.

Упроваджена модель є адекватною та обґрунтованою за прийнятими припущеннями:

- передбачено чітке розділення логіки між клієнтською частиною та БД;
- забезпечено базовий рівень безпеки облікових записів;

- система зручно масштабується - додавання нових міст та локацій не вимагає змін у коді застосунку.

Готовність до реального використання та перспективи розвитку:

На основі проведеної апробації можна зробити висновок, що розроблена ІС готова до обмеженого реального використання, зокрема:

- у рамках регіональних туристичних порталів;
- для туристичних агенцій або міських інформаційних центрів;
- у складі сайтів, що популяризують внутрішній туризм.

Перспективи розвитку ІС "Подорож містами України" включають:

- Інтеграція API сторонніх сервісів - зокрема, для автоматичного завантаження актуальної туристичної інформації, фото локацій та відгуків.
- Розширення функціоналу персоналізації - додавання рейтингів користувачів, рекомендацій маршрутів за інтересами, інтеграція з календарями.
- Підтримка багатомовності - для приваблення туристів з інших країн.
- Мобільна версія застосунку - для кращого доступу в процесі подорожі (у перспективі розробка PWA або нативного мобільного клієнта).
- Аналітичні модулі для адміністраторів - автоматизований збір статистики за популярністю локацій, аналіз активності користувачів тощо.

Рекомендації щодо подальших досліджень

Для підвищення якості системи у майбутніх роботах доцільно застосувати:

- методи оптимізації БД (індексація, кешування запитів);
- алгоритми персоналізованих рекомендацій (наприклад, за допомогою collaborative filtering);

- інструменти CI/CD для автоматизації тестування та деплою системи;
- інструменти автоматизованого UI-тестування для перевірки кросбраузерної сумісності.

Таким чином, пілотне впровадження показало високу доцільність використання ІС "Подорож містами України" у реальних умовах, а також значний потенціал її подальшого розвитку у напрямку створення сучасного інструменту підтримки та просування внутрішнього туризму України.

ВИСНОВКИ

У кваліфікаційному бакалаврському проєкті «Проектування та розроблення інформаційної системи «Подорож містами України»» було досліджено актуальність автоматизації процесів планування та адміністрування подорожей українськими містами. Така система відповідає сучасним потребам внутрішнього туризму, зростаючому попиту на цифрові сервіси для мандрівників, а також загальній тенденції переходу туристичних послуг у онлайн-формат.

У першому розділі було здійснено аналіз галузі управління екскурсіями та подорожами, визначено структуру об'єкта управління, його функції та ключові елементи, які мають бути підтримані інформаційною системою. Було проаналізовано сучасний стан ринку цифрових рішень для туризму, а також окреслено фактори, що впливають на організацію ІС для підтримки планування подорожей. Зроблено висновок щодо доцільності розробки системи, яка надає користувачам інтуїтивно зрозумілий інтерфейс для планування маршрутів, вибору локацій та формування персоналізованого досвіду подорожей.

У другому розділі проведено аналіз і специфікацію вимог до інформаційної системи «Подорож містами України». Були сформульовані бізнес-вимоги, функціональні та нефункціональні вимоги, виконано постановку задач автоматизації функцій адміністрування подорожами. Розроблено алгоритм функціонування системи та побудовано інформаційну модель підтримки управління подорожами містами України. Для моделювання поведінки системи створено діаграму варіантів використання, діаграми послідовності для основних сценаріїв роботи користувача та адміністратора, а також діаграму класів та даталогічну модель бази даних. Побудовані моделі дозволили формалізувати процеси, що реалізуються у

системі, забезпечити цілісність структури даних та узгодженість взаємодії компонентів системи.

У третьому розділі було спроектовано інформаційне, технічне та програмне забезпечення системи. Інформаційне забезпечення включає структуру бази даних та опис інформаційних потоків. Для технічного забезпечення було визначено необхідну конфігурацію серверного та клієнтського обладнання, що гарантує стабільну та ефективну роботу системи. У складі програмного забезпечення використано сучасні інструменти розробки. Було створено тестові сценарії, які охопили основні процеси системи. Результати тестування підтвердили працездатність розробленого функціоналу системи та відповідність вимогам до її стабільності та якості.

У процесі виконання проєкту було здійснено апробацію розробленого програмного забезпечення на тестовому прикладі з використанням реальних сценаріїв користувацької взаємодії. Проведений аналіз показав, що розроблена система демонструє високу гнучкість та ефективність для використання у сфері підтримки туристичних подорожей. Практична цінність запропонованої ІС полягає у спрощенні процесів планування подорожей, забезпеченні персоналізованого досвіду для мандрівників та автоматизації ключових функцій адміністрування даних про локації та маршрути.

На основі проведених досліджень та отриманих результатів можна стверджувати, що створена інформаційна система «Подорож містами України» відповідає вимогам предметної області, є готовою до подальшого розгортання у реальному середовищі та має потенціал до подальшого розвитку. У майбутньому перспективними напрямками удосконалення системи можуть стати інтеграція рекомендаційних механізмів, розширення підтримки мультимедійного контенту, оптимізація мобільної версії системи, а також підключення зовнішніх сервісів для автоматичного оновлення даних про туристичні об'єкти.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Державне агентство розвитку туризму України - офіційні плани та звіти, статистика внутрішнього туризму за 2023–2024 рр. URL: <https://www.tourism.gov.ua/plans-and-report> (дата звернення: 24.05.2025)
2. Join UP! - аналітика туристичних зборів за I півріччя 2023 року, середній чек і особливості бронювання турів - стаття “Туризм у 2023 році - головні підсумки...”. URL: <https://thepage.ua/ua/news/turizm-u-2023-roci-golovni-pidsumki-vid-turoperatora-join-up> (дата звернення: 24.05.2025)
3. Аналітика сезонності туристичних потоків. URL: <https://www.tourism.gov.ua/statistic> (дата звернення: 24.05.2025)
4. ДАРТ–UNICEF: опитування щодо подорожей під час війни. URL: <https://suspilne.media/culture/570361-podorozi-pid-cas-vijni-ponad-50-ukrainciv-ne-izdili-v-insi-regioni-z-turisticnou-metou-opituvanna> (дата звернення: 24.05.2025)
5. Ільницька-Гикавчук Г. Я. «Роль цифрових платформ в зростанні туристичного бізнесу» URL: <https://journals.khnu.km.ua/vestnik/wp-content/uploads/2022/01/en-2021-6t1-38.pdf> (дата звернення: 24.05.2025)
6. Inside UA. URL: <https://inside-ua.com/> (дата звернення: 24.05.2025)
7. Discover UA. URL: <https://discover.ua/> (дата звернення: 24.05.2025)
8. MistaUA. URL: <https://mista.ua/> (дата звернення: 24.05.2025)
9. Booking.com. URL: <https://www.booking.com/> (дата звернення: 24.05.2025)
10. Tripadvisor. URL: <https://www.tripadvisor.com/> (дата звернення: 24.05.2025)
11. Expedia. URL: <https://euro.expedia.net/> (дата звернення: 24.05.2025)

- 12.UX Design for Beginners - Interaction Design Foundation. URL: <https://www.interaction-design.org/literature/topics/ux-design> (дата звернення: 24.05.2025)
- 13.API. URL: <https://en.wikipedia.org/wiki/API> (дата звернення: 24.05.2025)
- 14.HTTPS-Protocol. URL: <https://en.wikipedia.org/wiki/HTTPS> (дата звернення: 24.05.2025)
- 15.Information model. URL: https://en.wikipedia.org/wiki/Information_model (дата звернення: 24.05.2025)
- 16.MySQL - MySQL 8.0 Reference Manual. URL: <https://dev.mysql.com/doc/refman/8.0/en/> (дата звернення: 25.05.2025)
- 17.HTML Living Standard. WHATWG. URL: <https://html.spec.whatwg.org/> (дата звернення: 25.05.2025)
- 18.Cascading Style Sheets (CSS). W3C. URL: <https://www.w3.org/Style/CSS/Overview.en.html> (дата звернення: 25.05.2025)
- 19.JavaScript. Mozilla Developer Network. URL: <https://developer.mozilla.org/en-US/docs/Web/JavaScript> (дата звернення: 25.05.2025)
- 20.Solid-state drive. URL: https://en.wikipedia.org/wiki/Solid-state_drive (дата звернення: 25.05.2025)
- 21.Data modeling. URL: https://en.wikipedia.org/wiki/Data_modeling (дата звернення: 25.05.2025)
- 22.Microsoft Windows 10. Офіційний сайт. URL: <https://www.microsoft.com/uk-ua/software-download/windows10> (дата звернення: 25.05.2025)
- 23.Draw.io - Diagrams.net. URL: <https://www.diagrams.net/> (дата звернення: 26.05.2025)

24. Enterprise Architect User Guide. Sparx Systems. URL: https://sparxsystems.com/enterprise_architect_user_guide/ (дата звернення: 26.05.2025)
25. Microsoft SQL Server Management Studio (SSMS). URL: <https://learn.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms> (дата звернення: 26.05.2025)
26. Visual Studio Code. URL: <https://code.visualstudio.com/> (дата звернення: 26.05.2025)
27. Figma - інструмент для дизайну інтерфейсів. URL: <https://www.figma.com/> (дата звернення: 26.05.2025)
28. UML 2.5.1 Specification. Object Management Group. URL: <https://www.omg.org/spec/UML/2.5.1> (дата звернення: 27.05.2025)
29. JSON - JavaScript Object Notation. ECMA International. URL: <https://www.json.org/json-en.html> (дата звернення: 27.05.2025)
30. Bootstrap - The most popular HTML, CSS, and JS library. URL: <https://getbootstrap.com/> (дата звернення: 27.05.2025)
31. Responsive Web Design Basics. Google Developers. URL: <https://developers.google.com/web/fundamentals/design-and-ux/responsive> (дата звернення: 27.05.2025)
32. GitHub Docs. URL: <https://docs.github.com/en> (дата звернення: 27.05.2025)

ДОДАТКИ

Додаток А

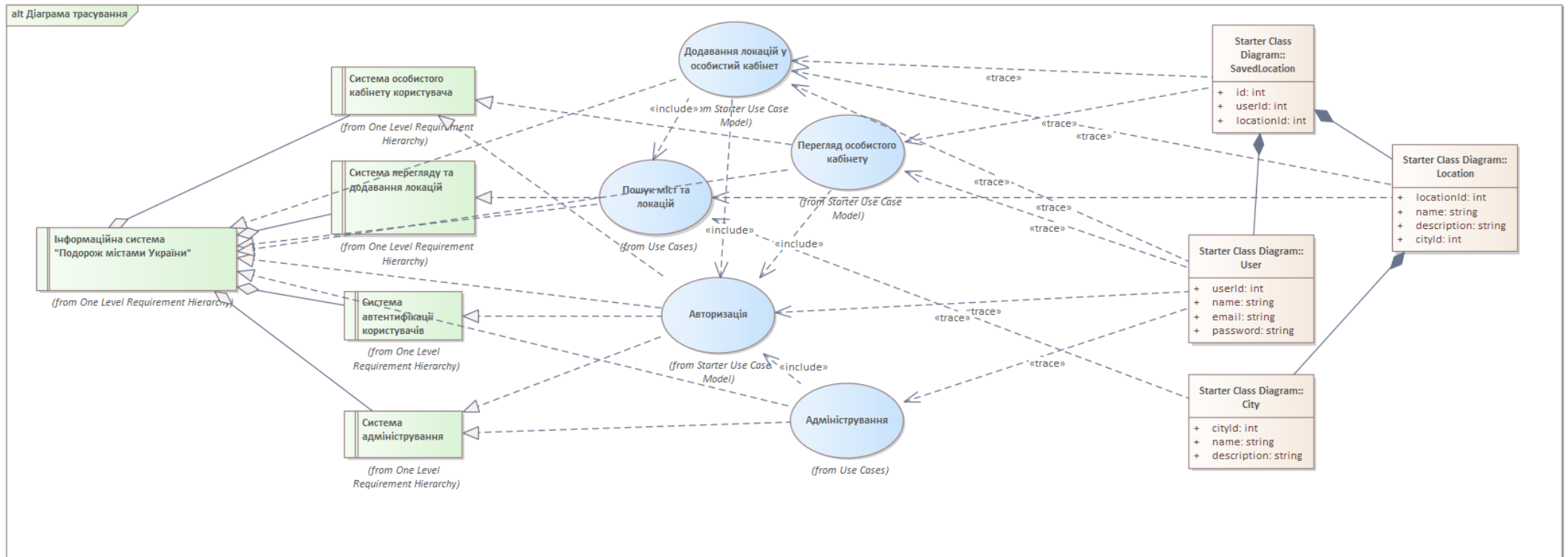


Рисунок А.1 – Діаграма трасування частина 1

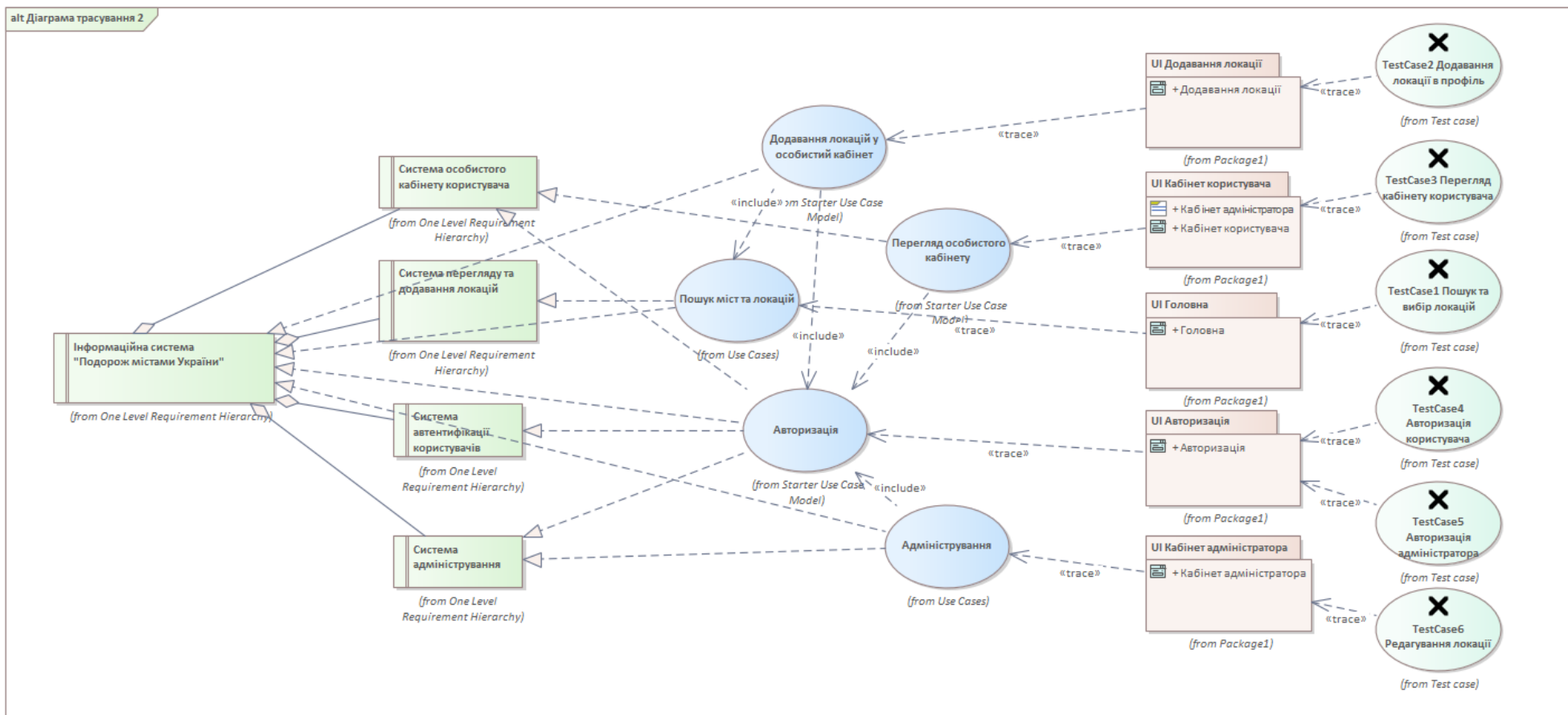


Рисунок А.2 – Діаграма трасування частина 2

Звіти Enterprise Architect – Scenario Builder

Test Report

12 May, 2025

TestCase1 Пошук та вибір локацій

UseCase «testcase» in package 'Test case'

TestCase1 Пошук та вибір локацій

Version 1.0 Phase 1.0 Proposed

Max created on 12.05.2025. Last modified 12.05.2025

TESTING
 Standard Unit test. TST-001
Description
1. Користувач відкрив головну сторінку IC
2. Користувач вибрав місто
3. Користувач вибрав локацію
Input
Місто: Київ
Локація: Золоті Ворота
Acceptance Criteria
Система відобразила міста та локації які відповідають вибраним критеріям
Results
1. Система коректно відобразила список міст
2. Система коректно відобразила список локацій
3. Система коректно відобразила локацію

Рисунок Б.1 – TestCase1 Пошук та вибір локацій

Test Report

12 May, 2025

TestCase2 Додавання локації в профіль

UseCase «testcase» in package 'Test case'

TestCase2 Додавання локації в профіль

Version 1.0 Phase 1.0 Proposed

Max created on 12.05.2025. Last modified 12.05.2025

TESTING
 Standard Unit test. TST-002
Description
1. Користувач додає локацію в маршрут
Input
Локація: Золоті Ворота
Кнопка: Додати в маршрут
Acceptance Criteria
1. Система додає локацію в профіль користувача
2. Система зберігає вибрану локацію
3. Система перенаправляє користувача на сторінку авторизації якщо він не увійшов в систему
4. Система у разі успіху повідомляє користувача що додала локацію в профіль
Results
Вибрана користувачем локація додалась в його профіль

Рисунок Б.2 – TestCase2 Додавання локації в профіль

TestCase3 Перегляд кабінету користувача

UseCase «testcase» in package 'Test case'

TestCase3 Перегляд кабінету користувача

Version 1.0 Phase 1.0 Proposed

Max created on 12.05.2025. Last modified 12.05.2025

TESTING
 Standard Unit test. TST-003
Description 1. Користувач зайшов в особистий кабінет 2. Користувач переглянув збережені пам'ятки
Input Особистий кабінет: збережені пам'ятки
Acceptance Criteria 1. Система перевірила чи увійшов користувач в свій обліковий запис 2. Система у разі якщо користувач не авторизований, перенаправляє на авторизацію 3. Система у разі якщо користувач авторизований відкриває його особистий кабінет
Results Користувач зайшов в особистий кабінет та має усю інформацію про свої введені дані та список збережених локацій

Рисунок Б.3 – TestCase3 Перегляд кабінету користувача

TestCase4 Авторизація користувача

UseCase «testcase» in package 'Test case'

TestCase4 Авторизація користувача

Version 1.0 Phase 1.0 Proposed

Max created on 12.05.2025. Last modified 12.05.2025

TESTING
 Standard Unit test. TST-004
Description 1. Користувач відкрив сторінку авторизації 2. Користувач ввів пошту та пароль 3. Користувач натиснув кнопку "Увійти" 4. Користувач перейшов в особистий кабінет
Input Пошта: user123@gmail.com Пароль: sequence123
Acceptance Criteria 1. Система перевірила наявність облікового запису в БД 2. Система авторизувала користувача 3. Система перенаправила користувача в особистий кабінет
Results Користувач успішно авторизувався в системі

Рисунок Б.4 – TestCase4 Авторизація користувача

TestCase5 Авторизація адміністратора

UseCase «testcase» in package 'Test case'

TestCase5 Авторизація адміністратора

Version 1.0 Phase 1.0 Proposed

Max created on 12.05.2025. Last modified 12.05.2025

TESTING

 Standard Unit test. TST-005

Description

1. Адміністратор відкрив сторінку авторизації
2. Адміністратор ввів пошту та пароль
3. Адміністратор натиснув кнопку "Увійти"
4. Адміністратор перейшов у особистий кабінет

Input

Пошта: admin1@gmail.com

Пароль: cooladmin322

Acceptance Criteria

1. Система перевірила наявність адміністратора в БД
2. Система авторизувала адміністратора
3. Система перенаправила адміністратора в особистий кабінет

Results

Адміністратор авторизувався в системі

Рисунок Б.5 – TestCase5 Авторизація адміністратора

TestCase6 Редагування локації

UseCase «testcase» in package 'Test case'

TestCase6 Редагування локації

Version 1.0 Phase 1.0 Proposed

Max created on 12.05.2025. Last modified 12.05.2025

TESTING

 Standard Unit test. TST-006

Description

1. Адміністратор авторизувався в системі
2. Адміністратор відкрив в особистому кабінеті місто
3. Адміністратор додав/змінив локації у вибраному місті

Input

Особистий кабінет: Київ

Локація: Золоті Ворота

Зміна Опис1 на Опис2

Затвердження зміни

Acceptance Criteria

1. Система відкрила особистий кабінет адміністратора
2. Система відкрила файл локацій міста
3. Система прийняла нову редакцію адміністратора
4. Система перезаписала дані про локацію в БД
5. Система сповістила адміністратора про успішне редагування

Results

Опис локації "Золоті Ворота" успішно було змінено

Рисунок Б.6 – TestCase6 Редагування локації

Фрагменти коду

Створення бази даних

```
CREATE DATABASE travel_cities_ua;
USE travel_cities_ua;
-- Таблиця користувачів (users)
CREATE TABLE users (
    id_user INT AUTO_INCREMENT PRIMARY KEY,
    email_user VARCHAR(50) NOT NULL UNIQUE,
    pass_user VARCHAR(255) NOT NULL,
    name_user VARCHAR(50),
    img_user BLOB,
    role_user ENUM('user', 'admin') DEFAULT 'user',
    stat_user BOOLEAN DEFAULT TRUE,
    sett_user TEXT
);
-- Таблиця локацій (locations)
CREATE TABLE locations (
    id_location INT AUTO_INCREMENT PRIMARY KEY,
    name_location VARCHAR(100) NOT NULL,
    description_location TEXT,
    image_location BLOB,
    category_location ENUM('history', 'culture', 'nature', 'entertainment', 'other'),
    address_location VARCHAR(255),
    rating_location DOUBLE DEFAULT 0
);
-- Таблиця особистих списків користувачів (user_lists)
CREATE TABLE user_lists (
    id_list INT AUTO_INCREMENT PRIMARY KEY,
    name_list VARCHAR(50) NOT NULL,
    color_list ENUM('blue', 'green', 'yellow', 'red', 'purple', 'orange'),
    id_user INT NOT NULL,
    FOREIGN KEY (id_user) REFERENCES users(id_user) ON DELETE CASCADE
);
-- Таблиця локацій у списках користувачів (list_locations)
CREATE TABLE list_locations (
    id_list_location INT AUTO_INCREMENT PRIMARY KEY,
    id_list INT NOT NULL,
    id_location INT NOT NULL,
    FOREIGN KEY (id_list) REFERENCES user_lists(id_list) ON DELETE CASCADE,
    FOREIGN KEY (id_location) REFERENCES locations(id_location) ON DELETE CASCADE
);
-- Таблиця адміністраторських дій (admin_logs)
CREATE TABLE admin_logs (
    id_log INT AUTO_INCREMENT PRIMARY KEY,
    id_admin INT,
    action VARCHAR(255),
    timestamp DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (id_admin) REFERENCES users(id_user) ON DELETE SET NULL
);
```

db.js:

```
const mysql = require('mysql2');

// Створення підключення
const db = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  password: 'your_password_here',
  database: 'travel_cities_ua'
});

// Підключення до бази даних
db.connect((err) => {
  if (err) {
    console.error('Помилка підключення до бази даних:', err.message);
    return;
  }
  console.log('Підключено до бази даних MySQL!');
});

module.exports = db;
```

Приклад API для отримання списку локацій (routes/locations.js)

```
const express = require('express');
const router = express.Router();
const db = require('../db');

// Отримати всі локації
router.get('/locations', (req, res) => {
  const sql = 'SELECT * FROM locations ORDER BY rating_location DESC';
  db.query(sql, (err, results) => {
    if (err) {
      console.error('Помилка при отриманні локацій:', err.message);
      res.status(500).send('Внутрішня помилка сервера');
      return;
    }
    res.json(results);
  });
});

module.exports = router;
```

Приклад клієнтського JS (на HTML-сторінці для взаємодії з API)

```
<script>
async function loadLocations() {
  try {
    const response = await fetch('http://localhost:3000/locations');
    const locations = await response.json();
    const list = document.getElementById('locationsList');
    list.innerHTML = '';

    locations.forEach(loc => {
```

```

        const item = document.createElement('li');
        item.textContent = `${loc.name_location} - Rating: ${loc.rating_location}`;
        list.appendChild(item);
    });
} catch (error) {
    console.error('Помилка при завантаженні локацій:', error);
}
}

// Завантажити локації при завантаженні сторінки
window.onload = loadLocations;
</script>

<ul id="locationsList"></ul>

```

Приклад POST-запиту - додавання нової локації:

```

// Додати нову локацію
router.post('/locations', (req, res) => {
    const { name_location, description_location, category_location, address_location, rating_location } =
    req.body;

    const sql = `
        INSERT INTO locations
        (name_location, description_location, category_location, address_location, rating_location)
        VALUES (?, ?, ?, ?, ?)
    `;

    db.query(sql, [name_location, description_location, category_location, address_location,
    rating_location], (err, result) => {
        if (err) {
            console.error('Помилка при додаванні локації:', err.message);
            res.status(500).send('Внутрішня помилка сервера');
            return;
        }
        res.status(201).send('Локацію успішно додано');
    });
});

```

Приклад клієнтського коду (JS) - відправлення POST-запиту на додавання нової локації:

```

<script>
    async function addLocation() {
        const name = document.getElementById('name').value;
        const description = document.getElementById('description').value;
        const category = document.getElementById('category').value;
        const address = document.getElementById('address').value;
        const rating = document.getElementById('rating').value;

        const response = await fetch('http://localhost:3000/locations', {
            method: 'POST',
            headers: {
                'Content-Type': 'application/json'
            },

```

```

        body: JSON.stringify({
            name_location: name,
            description_location: description,
            category_location: category,
            address_location: address,
            rating_location: parseFloat(rating)
        })
    });

    if (response.ok) {
        alert('Локацію успішно додано!');
    } else {
        alert('Помилка при додаванні локації.');
```

```
    }
}
```

```
</script>
```

```

<form onsubmit="event.preventDefault(); addLocation();">
    <input type="text" id="name" placeholder="Назва локації" required>
    <textarea id="description" placeholder="Опис локації" required></textarea>
    <input type="text" id="category" placeholder="Категорія (culture, nature, entertainment тощо)"
required>
    <input type="text" id="address" placeholder="Адреса локації" required>
    <input type="number" step="0.1" id="rating" placeholder="Рейтинг (0-5)" required>
    <button type="submit">Додати локацію</button>
</form>
```

Звіт подібності



Дата звіту6/13/2025

Дата редагування---



Звіт не був оцінений

Звіт подібності

метадані

Назва організації
Kyiv National Economic University named after Vadym Hetman KNEU

Заголовок
Проектування та розроблення інформаційної системи «Подорож містами України»

Автор
Науковий керівник / Експерт
Герасименко Максим ІгоровичСендзюк М.А.

підрозділ
кафедра інформаційних систем в економіці

Обсяг знайдених подібностей

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



Тривога

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

Заміна букв		1
Інтервали		0
Мікропробіли		2
Білі знаки		0
Парафрази (SmartMarks)		5

Подібності за списком джерел

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

10 найдовших фраз			Копіє тексту
ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)	
1	https://ir.kneu.edu.ua/server/api/core/bitstreams/1b35cb3a-b293-4677-9674-83983838c914/content	43	0.46 %
2	https://ir.kneu.edu.ua/server/api/core/bitstreams/1b35cb3a-b293-4677-9674-83983838c914/content	20	0.21 %
3	https://ir.kneu.edu.ua/bitstreams/850a2498-073e-45fe-874b-28be792ab58d/download	16	0.17 %
4	https://ir.kneu.edu.ua/server/api/core/bitstreams/add9ce43-9c83-4412-bccd-41d87f874e83/content	16	0.17 %