

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «магістр»
за спеціальністю 121 «Інженерія програмного забезпечення»
(освітня програма «Інженерія програмного забезпечення»)
на тему:

«Рекомендаційна система для бронювання курортних готелів»

Виконав студент групи ПЗм-24-1
МЕЛЬНИЧЕНКО Надія Миколаївна

Керівник роботи:
БАКАЛЮК Тетяна Анатоліївна

Рецензент:
ЄФРЕМОВ Юрій Миколайович

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

«ЗАТВЕРДЖУЮ»

Зав. кафедри інженерії
програмного забезпечення
Тетяна ВАКАЛЮК
«11» грудня 2025 р.

ЗАВДАННЯ
на кваліфікаційну роботу

Здобувач вищої освіти: **МЕЛЬНИЧЕНКО Надія Миколаївна**

Керівник роботи: **ВАКАЛЮК Тетяна Анатоліївна**

Тема роботи: **«Рекомендаційна система для бронювання курортних готелів»**, затверджена Наказом закладу вищої освіти від «11» грудня 2025 р., №479/с

Вихідні дані для роботи: **процес формування переліку рекомендованих готелів для користувача, який враховує історію бронювань, дані про готелі, оцінки інших користувачів.**

Консультанти з магістерської кваліфікаційної роботи із зазначенням розділів, що їх стосуються:

Розділ	Консультант	Завдання видав	Завдання прийняв
1	Тетяна ВАКАЛЮК	01.09.2025р.	01.09.2025р.
2	Тетяна ВАКАЛЮК	01.09.2025р.	01.09.2025р.
3	Тетяна ВАКАЛЮК	01.09.2025р.	01.09.2025р.

РЕФЕРАТ

Випускна кваліфікаційна робота магістра складається з алгоритму для формування рекомендацій на основі аналізу даних, онлайн-платформи для бронювання номерів курортних готелів та пояснювальної записки. Пояснювальна записка до випускної роботи містить 85 сторінки, 45 ілюстрацій, 3 таблиці, 4 формули, 3 додатка та 33 використаних джерел.

Метою роботи є розробка гібридної рекомендаційної системи із адаптивною вагою для формування списків готелів на платформі для бронювання курортних готелів, що дозволить підвищити монетизацію в додатку.

У роботі визначено основні завдання для розробки системи, проаналізовано наявні підходи, їхні переваги та недоліки. Проведено розробку прототипу покращеної гібридної системи та проведено її інтеграцію в розроблену платформу. Для реалізації було обрано відповідні сучасні технології. Рекомендаційна система була протестована та показала позитивні результати.

КЛЮЧОВІ СЛОВА: РЕКОМЕНДАЦІЙНА СИСТЕМА, ГІБРИДНИЙ ПІДХІД, КОЛАБОРАТИВНА ФІЛЬТРАЦІЯ, КОНТЕНТНИЙ ПІДХІД, ВЕБ-ПЛАТФОРМА, КУРОРТНІ ГОТЕЛІ, БРОНЮВАННЯ, ВІДГУКИ, БОНУСИ, PYTHON, REACT, API.

ABSTRACT

The master's thesis consists of an algorithm for generating recommendations based on data analysis, an online platform for booking resort hotel rooms, and an explanatory note. The explanatory note to the thesis contains 85 pages, 45 illustrations, 3 tables, 4 formulas, 3 appendices, and 33 sources used.

The purpose of the work is to develop a hybrid recommender system with adaptive weighting for generating hotel lists on a resort hotel booking platform, which will allow increasing monetization in the application.

The work identifies the main tasks for developing the system, analyzes existing approaches, their advantages and disadvantages. A prototype of an improved hybrid system was developed and integrated into the developed platform. Appropriate modern technologies were selected for implementation. The recommender system was tested and showed positive results.

KEYWORDS: RECOMMENDATION SYSTEM, HYBRID APPROACH, COLLABORATIVE FILTERING, CONTENT APPROACH, WEB PLATFORM, RESORT HOTELS, BOOKING, REVIEWS, BONUSES, PYTHON, REACT, API.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1. АНАЛІЗ І ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ ПОБУДОВИ РЕКОМЕНДАЦІЙНИХ СИСТЕМ ДЛЯ ПЛАТФОРМИ БРОНЮВАННЯ КУРОРТНИХ ГОТЕЛІВ	10
1.1 Аналіз предметної області	10
1.2 Огляд наявних аналогів та існуючі результати впровадження рекомендаційних систем	11
1.3 Наукові концепції та класифікація рекомендаційних систем ..	19
1.4 Сучасні проблеми та перспективи розвитку рекомендаційних систем ..	23
1.5 Постановка задачі дослідження	25
Висновки до першого розділу	27
РОЗДІЛ 2. РОЗРОБКА ТА ПРОЕКТУВАННЯ МОДЕЛІ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ	29
2.1 Модель рекомендаційної системи онлайн-платформи та її структурні складові	29
2.2 Методи та алгоритми обробки даних	32
2.3 Побудова та проектування моделі рекомендаційної системи ..	35
2.4 Оцінка точності та аналіз результатів моделі	40
Висновки до другого розділу	45
РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ	47
3.1 Основні вимоги до системи та умови експлуатації	47
3.2 Обґрунтування вибору засобів реалізації	51
3.3 Проектування та реалізація модулів системи	54
3.4 Опис інтерфейсу та сценарії роботи системи	58
3.5 Тестовий приклад роботи системи	66
Висновки до третього розділу	73
ВИСНОВКИ	74
ЛІТЕРАТУРА	76
ДОДАТКИ	80

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

CB (Content-Based) – контентна оцінка.

CF (Collaborative Filtering) – колаборативна оцінка.

MAE (Mean Absolute Error) – середня абсолютна помилка прогнозу.

RMSE (Root Mean Square Error) – середньоквадратична помилка.

NDCG (Normalized Discounted Cumulative Gain) – метрика якості ранжування.

API (Application Programming Interface) – інтерфейс програмування додатків, який дозволяє взаємодію між різними програмними компонентами.

JSON (JavaScript Object Notation) – формат обміну даними, який використовується для зберігання та передачі структурованої інформації

JWT (JSON Web Token) – стандарт відкритого JSON веб-токену, що використовується для безпечної передачі інформації між сторонами як об'єкт JSON.

SQL (Structured Query Language) – мова програмування, призначена для управління даними в реляційних базах даних.

ВСТУП

У сучасних умовах цифрової трансформації туристичної галузі питання персоналізації сервісів для пошуку та бронювання готелів набуває особливої ваги. Користувачі дедалі частіше очікують, що онлайн-платформи будуть здатні не просто відображати перелік доступних варіантів, а надавати індивідуальні рекомендації з урахуванням їхніх уподобань, історії пошуку, поведінкових патернів та контексту подорожі.

Актуальність теми «Рекомендаційна система для бронювання курортних готелів» зумовлена необхідністю створення точних та адаптивних персоналізованих рекомендацій, адже наявні системи часто не враховують специфічні потреби туристів, пов'язані з типом відпочинку, сезоном, розташуванням або наявністю інфраструктури, а стрімке зростання кількості пропозицій на ринку туристичних послуг лише ускладнює пошук необхідного варіанту.

Більш того, створення інтелектуальних рекомендаційних систем, що здатні підлаштовуватись під конкретного користувача, набуває ключового значення як у наукових дослідженнях, так і в практичних реалізаціях. Науковці у сфері інформаційних технологій активно досліджують принципи персоналізації, методи обробки даних, моделювання користувацьких вподобань та побудову рекомендацій, приділяючи окрему увагу проблемам «холодного старту», обмеженої інформації, контекстної залежності та ефективного використання історичних даних. В роботах також підкреслюється, що для туристичного ринку класичні моделі не завжди забезпечують високу якість рекомендацій, оскільки не враховують особливості курортних регіонів, сезонність попиту, варіативність пропозицій і різноманітність типів відпочинку, що підтверджує необхідність подальшого удосконалення існуючих алгоритмів.

Наразі активно розвивається напрям гібридного підходу, який поєднує контентний підхід із колаборативним фільтруванням. Саме вони дозволяють подолати більшість недоліків традиційних моделей. У контексті курортних

готелів, де важливими є такі параметри, як відстань до моря, наявність розважальних послуг, інфраструктура для дітей, тип харчування чи стиль відпочинку, гібридний підхід стає особливо ефективним і дозволяє суттєво розширити можливості персоналізації.

Мета дослідження полягає у розробці гібридної рекомендаційної системи для платформи бронювання курортних готелів, яка поєднує контентний та колаборативний підходи для формування персоналізованих рекомендацій із підвищеною точністю і ефективністю роботи в умовах обмежених або нерівномірних даних.

Для досягнення поставленої мети визначено такі завдання:

- провести огляд та аналіз сучасних наукових праць щодо алгоритмів побудови рекомендаційних систем;
- дослідити особливості контентного та колаборативного фільтрування у сфері туристичних сервісів;
- визначити обмеження та недоліки класичних моделей та обґрунтувати доцільність застосування гібридного підходу;
- сформулювати вимоги до системи рекомендацій у межах обраної предметної області;
- розробити структурну та алгоритмічну модель гібридної системи, що враховує характеристики користувачів, властивості готелів та історію взаємодій;
- провести тестування рекомендаційної системи, оцінити її точність та ефективність за відповідними метриками.

Об'єктом дослідження є процес формування персоналізованих рекомендацій у системах бронювання курортних готелів.

Предметом дослідження є методи та алгоритми побудови рекомендаційних моделей та принципи поєднання контентних і колаборативних алгоритмів у гібридну систему.

Методи дослідження включають машинне навчання, методи аналізу та попередньої обробки даних, нормалізації числових характеристик, кодування

категоріальних ознак, методи вимірювання схожості, кластеризацію, контент-орієнтоване фільтрування, колаборативну фільтрацію, а також статистичні методи аналізу та метрики оцінювання якості рекомендацій.

Інформаційну базу дослідження становлять наукові публікації у сфері інформаційних технологій, матеріали міжнародних, аналітичні звіти та джерела, присвячені рекомендаційним системам, а також тестові набори даних, сформовані для експериментальної оцінки моделі.

Наукова новизна дослідження полягає у створенні та обґрунтуванні гібридної моделі рекомендацій, адаптованої до особливостей курортного туризму, що поєднує переваги контентного та колаборативного підходів. Запропонована модель спрямована на покращення точності рекомендацій, зменшення впливу проблеми холодного старту та підвищення релевантності результатів у контексті індивідуальних запитів користувачів.

Практичне значення отриманих результатів полягає в тому, що розроблена модель може бути використана у реальних онлайн-сервісах бронювання для автоматизації процесу підбору пропозицій, покращення якості персоналізованого сервісу та підвищення задоволеності клієнтів. Розроблений підхід також може бути інтегрований у системи управління готелями, маркетплейси та інші цифрові туристичні сервіси, де важливу роль відіграє персоналізація.

Апробація. За темою кваліфікаційної магістерської роботи було опубліковано дві тези конференцій [1][2].

РОЗДІЛ 1. АНАЛІЗ І ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ ПОБУДОВИ РЕКОМЕНДАЦІЙНИХ СИСТЕМ ДЛЯ ПЛАТФОРМИ БРОНЮВАННЯ КУРОРТНИХ ГОТЕЛІВ

1.1 Аналіз предметної області

У сучасному світі онлайн-платформи для бронювання готелів перетворилися на ключовий інструмент взаємодії між користувачем і постачальниками послуг. Значну частину цього сегмента займають саме курортні готелі, які характеризуються високою залежністю від сезонності, орієнтацією на відпочинок та наданням широкого спектру сервісів. На відміну від міських готелів, де переважає бізнес-аудиторія або короткострокові зупинки, курортні комплекси мають більш комплексну структуру послуг: пляжні зони, розважальні програми, концепції «all inclusive», оздоровчі процедури, дитячі клуби та інші види додаткової інфраструктури.

Така різноманітність готельних об'єктів та відмінностей у сценаріях використання створює складний процес вибору, оскільки туристи керуються великою кількістю факторів, таких як наявність пляжу, якістю харчування, типом номерного фонду, близькістю до визначних місць, рівнем сімейного сервісу або орієнтацією на романтичний відпочинок. У результаті користувач стикається з необхідністю опрацювати суттєвий обсяг інформації, що підсилює проблему інформаційного перевантаження.

Сучасні онлайн-сервіси пропонують велику кількість варіантів проживання, що ускладнює процес прийняття рішення. Навіть за наявності фільтрів та сортування користувачам важко швидко оцінити, які саме пропозиції найбільше відповідають їх потребам. У багатьох випадках туристи обирають готель на основі попереднього досвіду або рекомендацій знайомих, що не гарантує об'єктивності вибору та не завжди відображає їхні актуальні вподобання.

Саме тому, рекомендаційні системи стають необхідним інструментом, що дозволяє автоматизувати процес підбору відповідних пропозицій.

Найважливішою функцією рекомендаційної моделі є формування персоналізованих списків готелів на основі індивідуальних характеристик користувача, його історії переглядів, бронювань, поведінкових патернів та контентних ознак готельних об'єктів. Впровадження таких систем дозволяє платформам підвищити якість взаємодії з клієнтом, збільшити конверсію, оптимізувати процеси пошуку та покращити користувацький досвід.

Таким чином, використання рекомендаційних систем у платформах бронювання готелів стає не лише додатковою функцією, а необхідною складовою сучасних інтелектуальних сервісів. Саме це визначає актуальність створення ефективної моделі, здатної поєднувати переваги різних підходів і забезпечувати більш точну персоналізацію результатів.

1.2 Огляд наявних аналогів та існуючі результати впровадження рекомендаційних систем

Рекомендаційні системи сьогодні є невід'ємною складовою сучасних онлайн-сервісів для бронювання житла, серед яких найбільш відомими є Booking.com, Airbnb, Expedia та інші великі платформи. Завдяки аналізу великих масивів даних про готелі, користувацьку активність та індивідуальні вподобання туристів такі системи дозволяють значно скоротити час пошуку, підвищити точність добору житла та покращити загальний користувацький досвід.

Booking.com [3] – це одна з найбільших платформ для пошуку і бронювання готелів (Рис. 1.1). Система використовує широкий набір параметрів для формування рекомендацій: ціну, рейтинг, кількість і зміст відгуків, місце розташування, наявні зручності, сезонні фактори та персональні цікавість користувача.

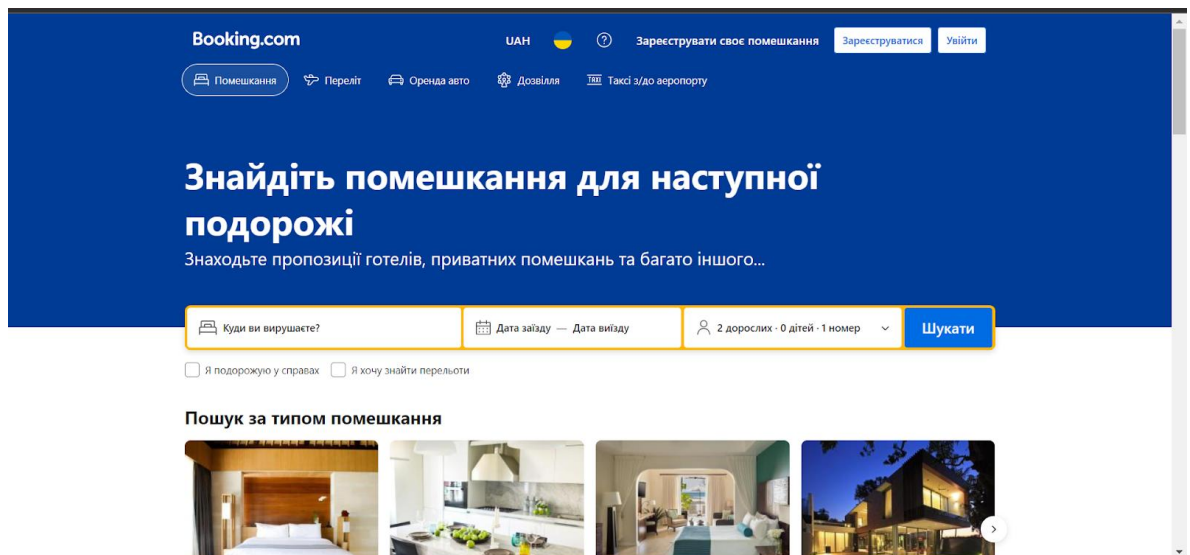


Рис. 1.1 Основна сторінка Booking.com

Особливістю Booking.com є те, що рекомендації формуються безпосередньо на сторінках інтерфейсу. На головній сторінці сервіс пропонує популярні напрямки, добірки спеціальних пропозицій та персоналізовані рекомендації, що переважно базуються на основі попередніх пошуків, місцезнаходженні користувача або загальних трендах (Рис. 1.2).

Сплануйте поїздки швидко та просто

Виберіть тип відпочинку та відвідайте найкращі місця в Україні

Романтика



Біла Церква
за 115 км



Вінниця
за 115 км



Ірпінь
за 116 км



Київ
за 133 км



Бровари
за 153 км



Хмельницький
за 153 км

Шукайте за типом помешкання в місті Житомир



Готелі



Апартаменти



Курортні готелі



Вілла

Пропозиції на вихідні



ACCENT Apart-Hotel
Львів, Україна
9.4 Чудово · 27 відгуків
Пропозиція "Вигідний початок року"
2 ночі ~~UAH 5 416~~ **UAH 4 333**



MYFREEDOM Апартаменти ЖК Нова Англія
Київ, Україна
8.8 Спокійно · 16 відгуків
Пропозиція "Вигідний початок року"
2 ночі ~~UAH 6 199~~ **UAH 3 720**



Maner hotel
Сєвастополь, Україна
8.2 Дуже добре · 425 відгуків
Пропозиція "Вигідний початок року"
2 ночі ~~UAH 1 650~~ **UAH 1 320**



DeMar Apart Caramel
Львів, Україна
9.4 Чудово · 19 відгуків
Пропозиція "Вигідний початок року"
2 ночі ~~UAH 4 490~~ **UAH 3 520**

Забронюйте наші найпопулярніші унікальні помешкання

Від замків і вілл до ботелів та іглу – у нас є все



Domki Wierszyki Shelters
Польща, Закопане
9.8 Відмінно · 88 відгуків



Ranczo w Dolinie
Польща, Kiszczowo
9.6 Відмінно · 183 відгуків



Tiny House Dreischwesternherz
Німеччина, Трір
9.5 Відмінно · 158 відгуків



Das rote Haus hinterm Deich
Німеччина, Simonsberg
9.2 Чудово · 47 відгуків

Рис. 1.2 Рекомендації незареєстрованого користувача

Якщо користувач авторизований, система відразу показує останні переглянуті варіанти, історію пошуків та готелі, які схожі на вже обрані ним раніше (Рис. 1.3).

Сплануйте поїздку швидко та просто

Виберіть тип відпочинку та відвідайте найкращі місця у Франції

Романтика Спокійний відпочинок Пляж Місто Лижі Активний відпочинок



Вільжуйф

7 км від напрямку Париж



Вітрі-сюр-Сен

8 км від напрямку Париж



Сюрен

9 км від напрямку Париж



Вірі-Шатійон

21 км від напрямку Париж



Villennes-sur-Seine

27 км від напрямку Париж



Вернон

68 км від напрямку Париж

Подорожуйте більше, витрачайте менше

[Докладніше про ваші винагороди](#)

Genius

Наділя, ви на 1-му рівні Genius у нашій програмі лояльності

Знижки 10% на проживання

Отримуйте знижки у відповідних помешканнях по всьому світі

Економія 10% на оренді авто

Зекономте на оренді певних автомобілів

Знижки 10-15% на проживання

Здійсніть 5 бронювань, щоб розблокувати Genius 2-го рівня

Пропозиції на вихідні

Зекономте на проживанні в період 31 січня - 2 лютого



Genius

ACCENT Apart-Hotel

Львів, Україна

9.4 Чудово - 27 відгуків

Пропозиція "Вигідний початок року"

2 ночі ~~UAH 5 416~~ UAH 3 900



Genius

Theatre Boutique Apartments

Київ, Україна

8.8 Блискуче - 2 437 відгуків

2 ночі ~~UAH 3 396~~ UAH 2 975



Genius

Bon Apart

Одеса, Україна

9.5 Відмінно - 766 відгуків

2 ночі ~~UAH 1 008~~ UAH 1 620



Genius

Jam Hotel Hnatyuka

Львів, Україна

8.8 Блискуче - 3 994 відгуків

2 ночі ~~UAH 5 607~~ UAH 3 532

Забронюйте наші найпопулярніші унікальні помешкання

Від замків і вілл до ботелів та іглу – у нас є все



Chez Helen

Франція, Marsac

9.3 Чудово - 173 відгуків

Від UAH 7 797



LE GITE DE RICHEMONT

Франція, Lachapelle-Saint-Pierre

8.8 Блискуче - 78 відгуків

Від UAH 9 450



Kaijansaaren Lomamökkit

Фінляндія, Tölvä

8.4 Дуже добре - 111 відгуків

Від UAH 14 766



Regina Vught

Нідерланди, Voort

7.9 Добре - 163 відгуків

Від UAH 12 049

Рис. 1.3 Рекомендації зареєстрованого користувача

Окрім великої кількості фільтрів, що дозволяють звузити пошук за бюджетом, типом житла, рейтингом або відстанню, Booking.com активно застосовує аналіз текстових відгуків, числових оцінок та фотографій для підвищення точності рекомендацій та формування довіри й впливу на рішення користувача (Рис. 1.4).

Головна > Франція > Іль-де-Франс > Париж > Результати пошуку

Париж: знайдено 2 258 помешкань

↓↑ Відсортувати за: Наші рекомендації

На нашому сайті на ваші дати недоступно 83% помешкань.

Сортувати за такими критеріями:

Ваш бюджет (за ніч)

За ніч За все перебування

UAH 1 000 - UAH 30 000+

Пропозиції

☐ Усі пропозиції 638

Популярні фільтри

☐ Центр Парижа 761

☐ Безкоштовне скасування 129

☐ Сніданок включено 790

☐ Два односпальні ліжка 631

☐ Готелі 1053

☐ 9-й округ - Опера 181

☐ 8-й округ - Єлисейські Поля 142

☐ 11-й округ - Площа Бастилії 141

Харчування

☐ З власною кухнею 1139

☐ Сніданок включено 790

☐ Сніданок і вечерю включено 8

Зручності

☐ Автостоянка 534

☐ Ресторан 207

☐ Можна з хатніми тваринами 594

La Belle Ville ★★★★★

19-й округ - Ла-Віллетт, Париж Показати на карті

4,6 км від центру Поряд з метро

Обмежена пропозиція

Покращений двомісний номер
1 двоспальне ліжко

На нашому сайті залишилося всього 4 номери за такою ціною

3 ночі, 2 дорослих
UAH 26 338

UAH 16 298

+UAH 2 218 податків та зборів

Показати наявні місця

Hotel 29 Lepic ★★★★★ Genius

18-й округ - Монмартр, Париж Показати на карті

3,5 км від центру Поряд з метро

Пропозиція "Вигідний початок року"

Стандартний двомісний номер
1 широке двоспальне ліжко

На нашому сайті залишилося всього 1 номер за такою ціною

3 ночі, 2 дорослих
UAH 21 701

UAH 13 330

+UAH 1 452 податків та зборів

Показати наявні місця

Le Monsieur By Sweett ★★★★★ Genius

9-й округ - Опера, Париж Показати на карті

1,9 км від центру

Таємна пропозиція

Двомісний номер
1 двоспальне ліжко

На нашому сайті залишилося всього 2 номери за такою ціною

3 ночі, 2 дорослих
UAH 23 992

UAH 17 274

+UAH 2 218 податків та зборів

Показати наявні місця

Рис. 1.4 Фільтрація результату запиту та показ пропозицій відносно рейтингу

Алгоритми платформи базуються на гібридному підході в якому колаборативна фільтрація враховує поведінку інших користувачів зі схожими вподобаннями (за переглядами, бронюваннями та оцінками), в той час як контентна фільтрація аналізує властивості самого готелю та порівнює їх з інтересами конкретного туриста.

На основі аналізу функціональних можливостей було сформовано таблицю переваг та недоліків даного сервісу:

Переваги та недоліки підходів Booking.com

Переваги	Недоліки
Масштабна база даних дозволяє порівнювати тисячі готелів та профілів користувачів.	Залишається проблема «холодного старту» для нових користувачів або готелів із невеликою кількістю даних.
Використання гібридних моделей забезпечує високу точність рекомендацій.	Алгоритми потребують значних обчислювальних ресурсів.
Актуальність даних гарантується регулярними оновленнями через API, що включають ціни, доступність та нові відгуки.	Не завжди враховує вузькоспеціалізовані вподобання.

Отже, Booking.com є потужним прикладом впровадження багаторівневих рекомендаційних систем. Водночас універсальність сервісу може бути певним обмеженням для вузькоспеціалізованих напрямків, наприклад, для підбору виключно курортних готелів із урахуванням їх специфічних характеристик. Тому у власній рекомендаційній системі бронювання курортних готелів було приділено більше уваги персоналізації на основі історії взаємодій і характеристик готелів, а не на географічних чи загальнопопулярних рекомендаціях, якщо користувач явно не вказує ці параметри.

Наступним важливим аналогом є Airbnb [4] – глобальна платформа короткострокової оренди житла, яка робить акцент на унікальних помешканнях, таких як приватні будинки, вілли, квартири, тематичні помешкання тощо. Платформа активно використовує рекомендаційні моделі для підвищення персоналізації та комфорту користувача.

Подібно до Booking.com, Airbnb формує рекомендації на головній сторінці, пропонуючи користувачу варіанти на основі історії переглядів, попередніх бронювань та геолокації (Рис. 1.5).

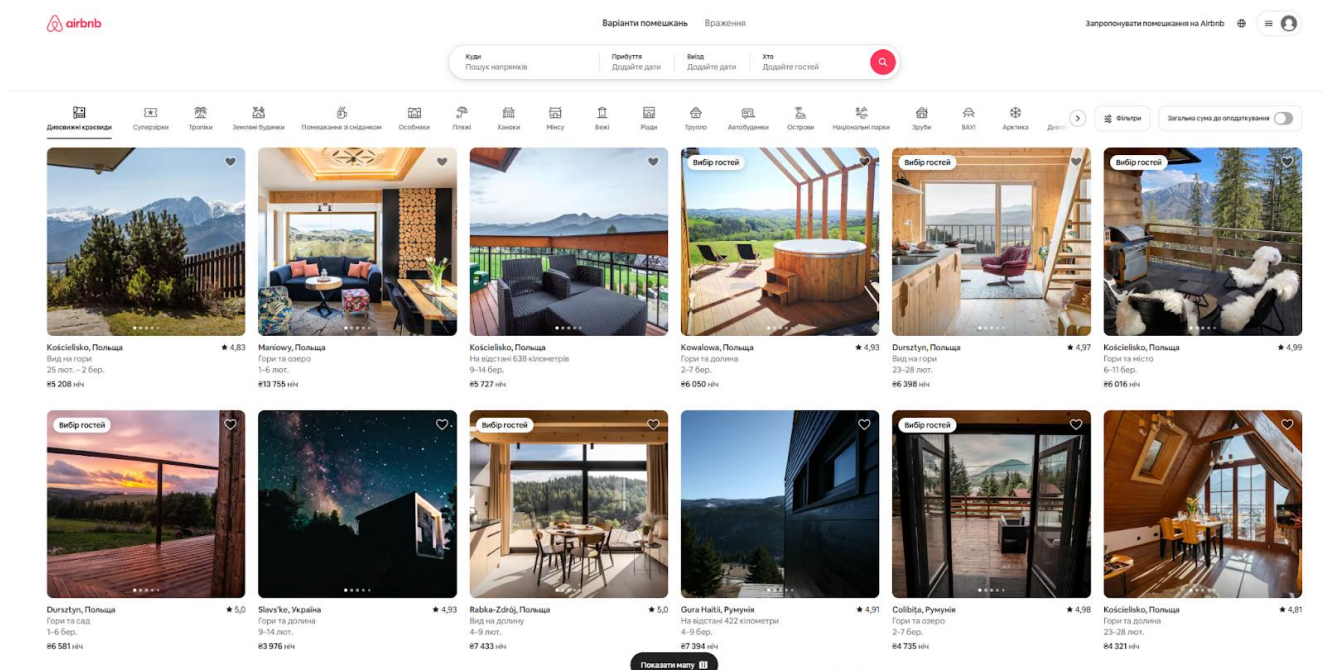


Рис. 1.5 Головна сторінка Airbnb

Окрім цього, важливою функцією є «Wishlist», списки бажань, які дозволяють користувачам зберігати вподобані пропозиції та створювати відповідні підбірки (Рис. 1.6). Ця функція надає алгоритмам більше інформації про інтереси користувача та покращує якість персоналізації.

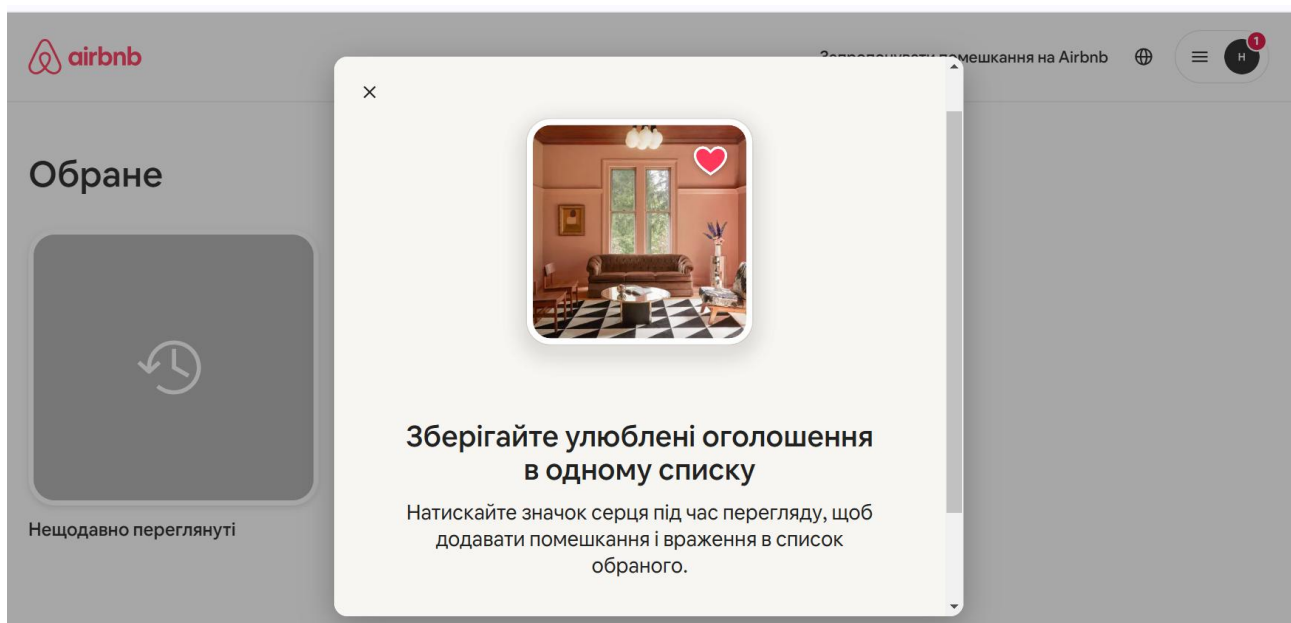


Рис. 1.6 Головна сторінка Airbnb

Платформа застосовує не лише контентну та колаборативну фільтрацію, а й активно використовує методи глибинного навчання для аналізу текстових

відгуків, обробки характеристик житла та фотографій. Це забезпечує більш природне розуміння якості об'єкта, врахування стилю оформлення, атмосфери та інших суб'єктивних факторів, які складно формалізувати звичайними методами (Рис. 1.7).

1 відгук

Середній рейтинг з'явиться після отримання 3 відгуків



Małgorzata
2 роки на Airbnb

★★★★★ · грудень 2024 р. · Перебування: кілька ночей
Ми вважаємо ваше перебування дуже успішним. Квартира чиста, простора. Багато обладнання (пральна машина, посудомийна машина, мікрохвильова піч). Повністю...

[Показати більше](#)

[Докладніше про відгуки](#)

Рис. 1.7 Відгуки до помешкання

На основі аналізу було сформовано узагальнену таблицю:

Таблиця 1.2

Переваги та недоліки підходів Airbnb

Переваги	Недоліки
Високий рівень персоналізації завдяки застосуванню глибинних нейронних мереж.	Значна обчислювальна складність моделей.
Залучення текстових відгуків для оцінки якості житла.	Проблема «холодного старту» все ще актуальна для нових об'єктів та користувачів.

Отже, Airbnb демонструє високий рівень адаптивності та здатність підлаштовуватися під реальні вподобання користувачів. Використання

глибинного навчання є одним із найбільш прогресивних підходів у сучасних рекомендаційних системах. Проте орієнтація на унікальне житло може не завжди забезпечувати високу точність для стандартних готельних запитів. Для власної системи бронювання курортних готелів доцільно перейняти підходи для аналізу текстових відгуків та використання гібридних моделей для підвищення релевантності рекомендацій.

1.3 Наукові концепції та класифікація рекомендаційних систем

Рекомендаційні системи сьогодні виступають важливими інструментами аналізу великих обсягів інформації та формування індивідуальних рекомендацій на основі вподобань користувача. Завдяки цим системам значно спрощується процес пошуку товарів або послуг, оскільки користувачеві не потрібно самостійно переглядати повні каталоги та порівнювати множину варіантів.

Загалом виділяють два основних типи рекомендаційних систем: персоналізовані, що враховують індивідуальні характеристики та історію взаємодії користувача з платформою, та неперсоналізовані, які формують однакові рекомендації для всіх користувачів, без урахування їхніх особистих характеристик [5].

Неперсоналізовані системи зазвичай використовуються для популяризації нових або популярних об'єктів, наприклад, нових готелів та номерів. Такий підхід також частково зменшує проблему «холодного старту», коли система ще не має достатньої кількості даних про самого користувача.

У персоналізованих рекомендаційних системах найпоширенішими підходами є колаборативна фільтрація, контентна фільтрація та гібридні моделі, які поєднують обидві попередні.

Колаборативна фільтрація

Колаборативна фільтрація є одним із найбільш розповсюдженим підходом сучасних рекомендаційних систем. Основна ідея методу полягає в тому, що користувачі зі схожими вподобаннями, як правило, цікавляться

подібними об'єктами. Цей принцип широко використовується у різних індустріях: від електронної комерції до музичних та відеосервісів.

У межах цього підходу виділяють два основні різновиди: User-based collaborative filtering та Item-based collaborative filtering. Для фільтрації на основі користувачів система знаходить користувачів із подібною історією оцінок або переглядів і формує рекомендації, спираючись на їхні вподобання. Алгоритм загалом включає такі етапи:

- визначення схожих користувачів за допомогою коефіцієнта кореляції Пірсона;
- вибір підмножини найбільш схожих користувачів (neighbourhood);
- формування рекомендацій або прогнозів на основі оцінок цієї підмножини [6].

Фільтрація на основі об'єктів аналізує схожість між об'єктами (наприклад, між готелями або номерами), які користувач вже оцінював. В основі методу лежить ідея, що користувач буде зацікавлений у схожих за характеристиками об'єктах. Даний метод аналізує характеристики об'єктів, такі як опис готелю, його рейтинг або розташування, та рекомендує варіанти з подібними параметрами [7]. Перевагою item-based підходу є висока масштабованість і менша обчислювальна складність у порівнянні з user-based моделями [8][9].

Метод колаборативної фільтрації широко використовується у багатьох відомих сервісах, таких як Amazon, Netflix, Spotify. Наприклад, Amazon використовує item-based підхід для прогнозування товарів, які можуть зацікавити користувача [10], а Netflix використовує user-based моделі, базуючись на історії переглядів та рейтингах. Spotify формує добірки відтворення на основі схожих користувачів.

Колаборативна фільтрація також активно застосовується у більш специфічних сферах: медицині, освіті, рекрутингу, HR-системах, що підкреслює універсальність підходу.

Серед переваг такої фільтрації, варто зазначити, що цей підхід не потребує аналізу характеристик об'єктів. Метод сприяє виявленню нового контенту, який може бути неочевидним для користувача, що розширює його інтереси [10], а також добре працює у платформах з великими масивами даних.

Однак, існує проблема «холодного старту», проблема розрідженості даних, де більшість користувачів оцінює мало об'єктів, та проблема масштабованості при збільшенні обсягів історичних даних [11].

Контентна фільтрація

Контентна фільтрація також вважається одним із ефективних методів для створення персоналізованих рекомендацій. Такий підхід фокусується на характеристиках об'єктів. Система порівнює атрибути об'єктів (наприклад, опис, рейтинг, тип готелю, зручності, відстань до моря) з вподобаннями користувача та пропонує ті варіанти, що найбільше відповідають його профілю.

Цей підхід дозволяє отримувати персоналізовані рекомендації навіть за умов обмеженої інформації про інших користувачів. Модель також забезпечує високу прозорість, адже користувач легко може зрозуміти, чому система рекомендувала саме цей готель. Саме це і являється ключовою перевагою, яка забезпечує точність результатів і мінімізує ризик включення нерелевантних рекомендацій [12].

Окрім цього, контентна фільтрація дозволяє інтегрувати нові об'єкти в систему без очікування на отримання оцінок чи відгуків. Це особливо важливо для нових продуктів або контенту, який ще не отримав оцінок чи відгуків. Відповідно, система може ефективно функціонувати навіть на ранніх етапах, коли інформації про вподобання користувачів ще мало.

Проте такий підхід має ризик «пастки схожості», коли система пропонує лише подібні до вже оцінених об'єкти, не пропонуючи нічого нового чи несподіваного [13]. Залишається часткова проблема «холодного старту» для нових об'єктів, якщо бракує опису або атрибутів, залишаючи їх непомітними серед інших, що негативно впливає на формування популярності [14].

Для систем бронювання готелів контентна фільтрація є надзвичайно корисною, оскільки готелі мають великий набір структурованих характеристик, що дозволяє точно описати об'єкти та побудувати ефективну модель.

Гібридні підходи

Гібридні рекомендаційні системи поєднують у собі переваги колаборативної та контентної фільтрації. Такий підхід дозволяє усунути недоліки кожного методу окремо та значно підвищити точність рекомендацій[15].

Гібридні моделі враховують одночасно контентні характеристики об'єктів, поведінку та історію взаємодій користувача, контекст запиту. На практиці це дозволяє суттєво зменшити проблему «холодного старту», враховувати як атрибутивні, так і поведінкові моделі, і відповідно підвищити релевантність рекомендацій.

Для прикладу, система Fab інтегрує аналіз контенту зі знаходженням схожостей у поведінці користувачів, що покращує якість рекомендацій та забезпечує адаптивність у різних сценаріях використання [14].

Відомі компанії, такі як Netflix, Amazon, YouTube, Coursera, Rozetka, також застосовують гібридні моделі. На потокових платформах, наприклад, YouTube, комбінується контентне порівняння із статистикою переглядів [15]. Все частіше ці технології інтегруються й у соціальні мережі, де допомагають знаходити нових друзів, події або групи, які можуть зацікавити користувача.

Окремим напрямом розвитку гібридних систем у сфері бронювання готелів є аналіз настроїв (sentiment analysis) на основі текстових відгуків. У дослідженнях часто показано, що врахування емоційної оцінки значно підвищує якість рекомендацій [16].

Таким чином, гібридні моделі є найбільш перспективними для побудови рекомендаційних систем у сфері курортних готелів, оскільки дозволяють комплексно враховувати характеристики готелів, поведінку користувачів, сезонність, популярність напрямків і навіть текстові відгуки. Саме тому для

розробки рекомендаційної системи курортних готелів вибір гібридного підходу є оптимальним рішенням.

1.4 Сучасні проблеми та перспективи розвитку рекомендаційних систем

Аналізуючи сучасні алгоритми формування рекомендацій, можна виділити низку проблем і обмежень, що впливають на точність, ефективність і стабільність роботи рекомендаційних систем, особливо у сфері бронювання готелів.

Однією з ключових проблем залишається проблема «холодного старту», яка виникає тоді, коли система не має достатньо інформації про нового користувача або новий об'єкт. У туристичній галузі ця проблема проявляється особливо помітно, оскільки багато користувачів здійснюють бронювання кілька разів на рік або навіть рідше, а готелі постійно оновлюють пропозиції. У результаті система не може сформувати повністю релевантні рекомендації, що негативно впливає на перший досвід взаємодії з сервісом. Подібні труднощі виникають і тоді, коли кількість схожих варіантів обмежена: модель формує занадто повторювані рекомендації, звужуючи інтереси користувача [17].

Ще одним важливим питанням є масштабованість системи. Зі збільшенням кількості користувачів, готелів, сезонних пропозицій зростає і навантаження на алгоритми. Система бронювання має працювати зі значними обсягами даних у режимі реального часу, забезпечуючи швидку відповідь та актуальність результатів. Це потребує оптимізації моделей, використання розподілених обчислень та підвищення ефективності зберігання даних.

Не менш суттєвою є проблема розрідженості даних [18]. Більшість користувачів не залишає оцінок або відгуків після кожного бронювання, а деякі взаємодії з платформою не дають достатньої інформації для формування профілю. Це ускладнює застосування класичних методів колаборативної фільтрації, які сильно залежать від матриці оцінок.

З метою підвищення точності та релевантності рекомендацій у туристичній галузі набуває популярності використання контекстної інформації [19]. Сезонність, місце перебування користувача, тип пристрою, погодні умови, бюджет поїздки або мета подорожі – усе це може суттєво вплинути на формування рекомендацій. Контекстно залежні моделі дозволяють створювати більш адаптивні рекомендації навіть тоді, коли даних про користувача ще недостатньо, частково розв'язуючи проблему холодного старту [20].

Окремим напрямком розвитку є пояснювальні рекомендації. Їхньою метою є підвищення прозорості та зрозумілості системи [21]. Наприклад, користувач може отримати пояснення: «рекомендовано, оскільки схоже на попередньо переглянуті вами готелі» або «популярно серед мандрівників вашого віку». Такий механізм підвищує довіру до рекомендацій і полегшує прийняття рішення.

Подібні підходи вже успішно використовуються у музичних сервісах, де плейлисти формуються «на основі прослуханих треків» або «на основі улюбленого жанру».

Використання методів глибинного навчання дозволяє враховувати настрої та емоційну складову текстових відгуків, розпізнавати тональність і навіть аналізувати фотографії. Такі моделі дають змогу виявляти приховані патерни, які складно формалізувати за допомогою традиційних алгоритмів, що є особливо корисним для оцінки якості готелів, рівня сервісу чи атмосфери закладу.

Отже, сучасні рекомендаційні системи у туристичній сфері стикаються з низкою технічних і концептуальних викликів. Водночас активний розвиток гібридних моделей, контекстного аналізу, методів глибинного навчання та пояснюваних рекомендацій створює широкі можливості для побудови точних, гнучких і персоналізованих систем. Ці тенденції визначають подальші напрями досліджень і відкривають перспективи для удосконалення систем бронювання курортних готелів.

1.5 Постановка задачі дослідження

Після проведення огляду сучасних підходів до побудови рекомендаційних систем, аналізу моделей та алгоритмів, що використовуються світовими онлайн-платформами, а також врахування ключових проблем галузі, стало можливим визначити фактори, які найбільш суттєво впливають на ефективність рекомендаційних систем у сфері бронювання курортних готелів. Результати аналізу дали змогу окреслити вимоги до майбутньої моделі, сформулювати основну мету та виділити етапи, необхідні для побудови повноцінної рекомендаційної системи.

Головною вимогою до системи є персоналізація, тобто здатність моделі враховувати поведінкові особливості користувача і формувати рекомендації відповідно до його інтересів. Для цього необхідно забезпечити можливість збору, попередньої обробки та аналізу користувацьких даних: історії бронювань, переглянутих готелів, переваг щодо типу відпочинку чи бюджету, а також особливостей взаємодії зі сторінками платформи. Додатково важливим є використання атрибутивної інформації про готелі, що дозволяє комбінувати інтереси користувача з описами, характеристиками та властивостями об'єктів.

Ще одним важливим критерієм є актуальність даних. Система повинна працювати з динамічними даними, такими як змінами вартості, доступності номерів, новими відгуками та рейтинговими показниками. Оскільки туристичні платформи функціонують у режимі реального часу, алгоритми мають бути здатними швидко та коректно оновлювати інформацію, забезпечуючи релевантність рекомендацій.

Для підвищення точності персоналізованих пропозицій доцільним є застосування алгоритмів машинного навчання. На основі проведеного аналізу визначено, що в умовах предметної області найбільш ефективними є три підходи, такі як колаборативна фільтрація, орієнтована на поведінку схожих користувачів, контентна фільтрація, що враховує атрибути готелів, та гібридні моделі, які поєднують переваги обох підходів і компенсують їх недоліки.

В межах даного дослідження було обрано саме гібридний підхід, оскільки він найповніше відображає специфіку курортної сфери та дозволяє врахувати комплексні взаємозв'язки між поведінковими та контентними даними. Така модель здатна працювати з неповними, застарілими або неоднорідними набором даних, зберігаючи точність формування рекомендацій.

На рисунку 1.8 подано узагальнену схему роботи рекомендаційної системи для бронювання курортних готелів, яка відображає ключові етапи: збір даних, їхню попередню обробку, інтеграцію у модель та формування вихідного списку рекомендацій.

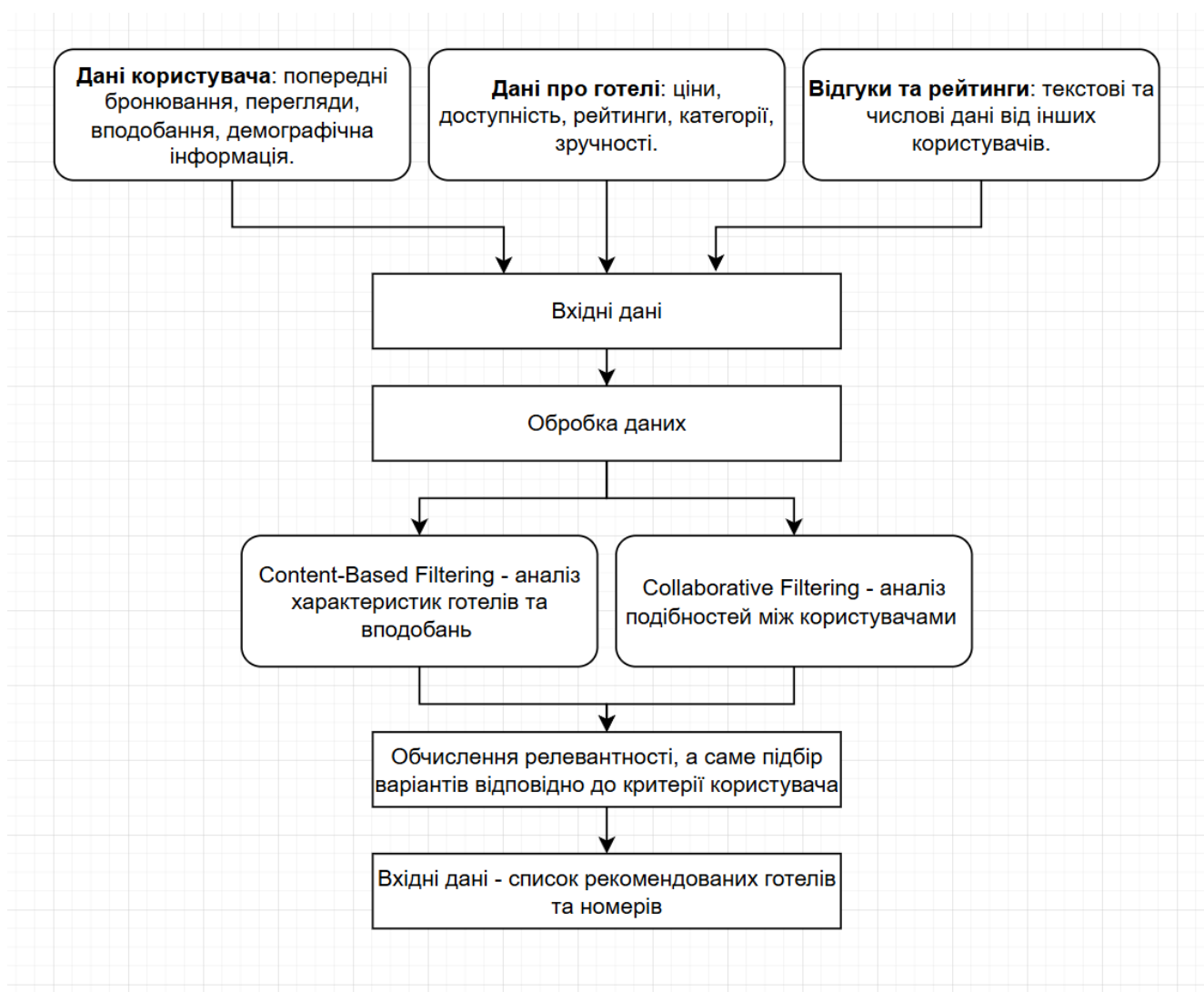


Рис. 1.8 Схема роботи рекомендаційної системи

Дана схема демонструє ключові компоненти системи, логіку їх взаємодії та роль кожного етапу у процесі формування персоналізованих рекомендацій.

На основі сформульованих вимог визначено основні задачі, які необхідно розв'язати в межах даного дослідження:

1. Визначити структуру вхідних даних та сформувати профіль користувача на основі доступної інформації;
2. Провести обробку даних про готелі, виділити ключові атрибути та забезпечити їх коректне структурування;
3. Реалізувати колаборативний компонент моделі для аналізу поведінкової схожості користувачів;
4. Розробити контентний компонент системи, що порівнює характеристики готелів з профілем користувача;
5. Створити гібридну модель, інтегрувавши колаборативний та контентний підходи;
6. Забезпечити механізм оновлення даних і адаптації системи до змін поведінки користувача та оновлення інформації про готелі;
7. Провести тестування моделі на реальних або тестових наборах даних, здійснити оцінку точності рекомендацій;
8. Реалізувати прототип рекомендаційної системи з можливістю інтеграції у веб-платформу для бронювання.

Таким чином, постановка задачі визначає логічну структуру майбутньої системи, її цілі, функціональні вимоги та основні етапи реалізації. Це створює чітку основу для подальшого проєктування та впровадження гібридної моделі рекомендацій у сфері бронювання курортних готелів.

Висновки до першого розділу

В першому розділі було проведено комплексний аналіз теоретичних засад побудови рекомендаційних систем та особливостей їх застосування у сфері бронювання курортних готелів. Розглянуто предметну область, проаналізовано найбільш відомі онлайн-платформи та визначено переваги і недоліки сучасних моделей рекомендацій, що використовуються у міжнародній практиці.

Особливу увагу було приділено трьом основним підходам, а саме: колаборативній фільтрації, контентному методу та гібридним моделям. Було з'ясовано сфери їх застосування, сильні сторони та ключові обмеження.

Аналіз сучасних наукових досліджень показав, що рекомендаційні системи для туристичної галузі стикаються з типовими проблемами: «холодним стартом», розрідженістю даних, складністю масштабування та обмеженою точністю при зміні зовнішніх умов (сезонність, тип відпочинку, місцезнаходження користувача). Водночас виявлено сучасні тенденції розвитку рекомендаційних технологій, серед яких найбільш перспективними є гібридні підходи, контекстна персоналізація, аналіз текстових відгуків та застосування моделей машинного навчання.

На основі проведеного аналізу було сформульовано постановку задачі дослідження та визначено ключові вимоги до майбутньої системи: забезпечення персоналізації, актуальності рекомендацій, адаптивності до змін поведінки користувача та здатності працювати з неповними або неоднорідними даними.

Таким чином, перший розділ забезпечив необхідне теоретичне підґрунтя для проєктування та реалізації прототипу рекомендаційної системи бронювання курортних готелів.

РОЗДІЛ 2. РОЗРОБКА ТА ПРОЕКТУВАННЯ МОДЕЛІ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ

2.1 Модель рекомендаційної системи онлайн-платформи та її структурні складові

Рекомендаційна система для бронювання курортних готелів у рамках даного дослідження розглядається як комплексна модель, що об'єднує різні типи даних, методи їх опрацювання та алгоритми визначення релевантних пропозицій для конкретного користувача. На основі результатів аналізу, представлених у першому розділі, були сформульовані вимоги до моделі, здатної забезпечити персоналізований підбір готельних об'єктів з урахуванням особливостей курортної сфери.

Побудова такої системи передбачає виділення основних типів даних, з якими вона працює. Для початку, це інформація про готелі: їх розташування, категорія, доступні зручності, рейтинг, відгуки, середня ціна та інші властивості. Також це дані про користувача, які відображають його взаємодію з платформою: перегляди, пошукові запити, попередні бронювання, збережені об'єкти та загальна поведінка під час користування сервісом. Саме комбінація цих двох груп інформації дозволяє формувати рекомендації, що максимально відповідають інтересам користувача.

Для досягнення високої точності було обрано архітектуру, яка складається з трьох основних модулів. Контентний аналізує характеристики готелів та визначає схожість між ними. Колаборативний досліджує поведінку користувачів і виявляє схожі патерни у виборі готелів серед різних людей. Гібридний поєднує результати двох попередніх і дозволяє компенсувати обмеження кожного окремого підходу. Така структура забезпечує стійкість системи до розрідженості даних та підвищує якість рекомендацій у ситуаціях, коли користувач має мало взаємодій з платформою.

Відповідно сам процес роботи системи складається з кількох послідовних етапів:

1. Попередня підготовка даних, що включає очищення, нормалізацію та кодування категоріальних атрибутів;
2. Побудова профілю готелів шляхом формування багатовимірних векторів ознак;
3. Формування профілю користувача на основі його взаємодій з платформою;
4. Обчислення схожості на рівні контентних характеристик та поведінкових даних;
5. Комбінування результатів у гібридну оцінку;
6. Формування фінального списку рекомендованих готелів.

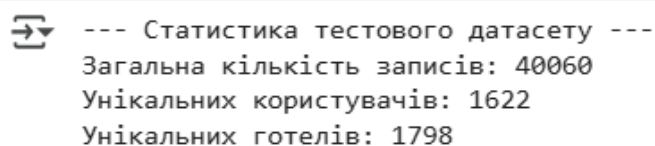
Окремим етапом дослідження був вибір технологій для реалізації моделі. Основною мовою програмування було обрано Python [22], оскільки він широко використовується у сфері аналізу даних та машинного навчання і забезпечує доступ до великої кількості спеціалізованих бібліотек для побудови рекомендаційних систем.

Під час проєктування були розглянуті такі інструменти, як TensorFlow [23] та PyTorch [24]. TensorFlow було оцінено як потенційний інструмент для реалізації складних моделей у перспективі, адже, хоча це потужна бібліотека, що підтримує роботу з великими датасетами, складними нейронними мережами та моделюванням високої глибини, його застосування потребує значних обчислювальних ресурсів та складної організації процесу навчання. PyTorch являється гнучкою бібліотекою для побудови нейронних мереж, що широко використовується у наукових дослідженнях. PyTorch дозволяє реалізовувати високорівневі deep learning-моделі, які можуть працювати з текстами, зображеннями або широкими наборами ознак. Однак для задачі рекомендацій курортних готелів такі складні моделі не є необхідними на початкових етапах, оскільки вимагають великих обсягів позначених даних та серйозних ресурсів для тренування.

Для цілей даного проєкту оптимальним рішенням стала бібліотека Scikit-learn [25], яка містить необхідний набір класичних алгоритмів, інструменти для

нормалізації, обчислення відстаней, побудови матриць подібності та обробки розріджених даних. Це дозволило реалізувати гібридний підхід без надмірних обчислювальних витрат та забезпечити зрозумілу й прозору структуру моделі.

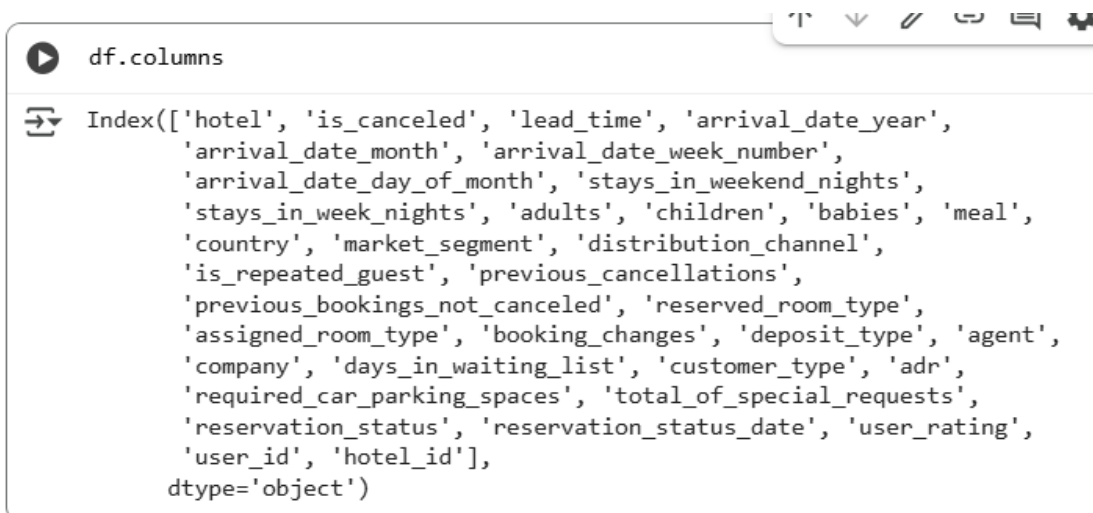
Відповідно для побудови та оцінювання системи необхідно попередньо дослідити структуру доступних даних, їх якість, повноту та відповідність вимогам моделі. Оскільки особливістю туристичної сфери є розрідженість взаємодій користувачів, тестовий набір даних має також відтворювати типові умови роботи реальної платформи бронювання. Вибраний датасет містить інформацію про користувачів, готелі, бронювання, властивості номерів та параметри взаємодії, що у сукупності забезпечує можливість побудови як контентної, так і колаборативної складової моделі. Загальна кількість даних наведена на рисунку 2.1.



```
--- Статистика тестового датасету ---  
Загальна кількість записів: 40060  
Унікальних користувачів: 1622  
Унікальних готелів: 1798
```

Рис. 2.1 Дані тестового датасету

Даний набір містить усі важливі дані (Рис. 2.2). Значна частина полів характеризує сам готель: його локацію, особливості відпочинку, категорію, рейтинг, набір зручностей, типи номерів та допоміжні властивості. Інша частина описує властивості користувачів: здійснені бронювання, оцінки, історію взаємодій, дати бронювання, тип користувача.



```
df.columns  
  
Index(['hotel', 'is_canceled', 'lead_time', 'arrival_date_year',  
      'arrival_date_month', 'arrival_date_week_number',  
      'arrival_date_day_of_month', 'stays_in_weekend_nights',  
      'stays_in_week_nights', 'adults', 'children', 'babies', 'meal',  
      'country', 'market_segment', 'distribution_channel',  
      'is_repeated_guest', 'previous_cancellations',  
      'previous_bookings_not_canceled', 'reserved_room_type',  
      'assigned_room_type', 'booking_changes', 'deposit_type', 'agent',  
      'company', 'days_in_waiting_list', 'customer_type', 'adr',  
      'required_car_parking_spaces', 'total_of_special_requests',  
      'reservation_status', 'reservation_status_date', 'user_rating',  
      'user_id', 'hotel_id'],  
      dtype='object')
```

Рис. 2.2 Поля тестового датасету

Дані містять як числові, так і категоріальні значення. Саме тому попередньо відбувається нормалізації форматів, перетворення категорій у числові ознаки та масштабування параметрів. Однак перед побудовою моделі було сформовано підмножину даних, яка включає лише релевантні для рекомендаційної системи поля, виключаючи не важливі чи ті, що містять другорядну інформацію. Це дозволило зменшити розмірність простору ознак, підвищити стабільність моделі та зосередитися на факторах, що дійсно визначають схожість готелів та поведінку користувачів.

На основі новоутвореного датасету будуються дві окремі моделі, які будуть об'єднані в єдину гібридну систему. Даний датасет використовувався для перевірки коректності всіх етапів обчислень, аналізу схожості готелів, визначення груп користувачів з подібними вподобаннями та формування фінального списку рекомендацій.

Отже, сформована архітектура рекомендаційної системи визначає логіку роботи всіх її складових та створює основу для подальшого опису алгоритмів, методів опрацювання даних і побудови фінальної гібридної моделі.

2.2 Методи та алгоритми обробки даних

Для побудови гібридної рекомендаційної системи, що враховує як характеристики курортних готелів, так і поведінку користувачів, необхідно застосувати комплекс методів обробки даних, здатних працювати з різномірною інформацією. Кожен із використаних алгоритмів виконує свою функцію у загальній структурі системи й забезпечує перетворення вихідних даних у форму, придатну для аналізу, порівняння та формування підсумкових рекомендацій.

Першим етапом є побудова векторних представлень готелів та користувачів [26]. Для цього використовуються атрибути готелів, які включають кількість зірок, рейтинг, кількість відгуків, розташування, перелік доступних зручностей і типи номерів. Для коректного перетворення категоріальних ознак застосовується метод one-hot encoding [27], який дозволяє

уникнути викривлення ваг між значеннями, оскільки всі категорії перетворюються у незалежні бінарні атрибути. Числові ознаки проходять нормалізацію за допомогою StandardScaler [28], що забезпечує однакову вагу кожної ознаки при подальших обчисленнях. У результаті формується багатовимірний вектор, який відображає ключові властивості кожного готелю.

Оцінювання схожості є центральним елементом обох контентної та колаборативної частин моделі. Найбільш ефективною метрикою для таких даних є косинусна схожість [29], яка визначає кут між двома векторами ознак. Наприклад, порівнює два готелі за їхніми ознаками (кількість зірок, тип харчування, наявність басейну) або два користувачі за історією бронювань. Вона добре працює в умовах розрідженості, коли значення векторів можуть бути неповними або мати різні масштаби, що є типовою ситуацією у туристичних системах.

Формула для обчислення косинусної схожості виглядає наступним чином (формула 1):

$$sim(A, B) = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} * \sqrt{\sum_{i=1}^n B_i^2}}$$

де $A_i B_i$ – значення ознак двох порівнюваних об'єктів [30].

Коефіцієнт кореляції Пірсона враховує різницю в тому, як користувачі ставлять оцінки. Наприклад, один користувач може оцінювати лише у високому діапазоні, інший – у низькому, але обом можуть подобатися схожі за якістю готелі. Кореляція дозволяє виявити ці спільні закономірності.

Формула виглядає так (формула 2):

$$r = \frac{\sum (A_i - \bar{A})(B_i - \bar{B})}{\sqrt{\sum (A_i - \bar{A})^2} \sqrt{\sum (B_i - \bar{B})^2}}$$

де r – коефіцієнт кореляції; A_i, B_i – значення у двох відповідних векторах (наприклад, оцінки готелів двома користувачами); $\bar{A}, \bar{B}$ – середні значення відповідних векторів.

Для подальшого вдосконалення системи можуть застосовуватися методи кластеризації, як-от k-means, що дозволяють групувати як готелі, так і користувачів за схожими характеристиками [31]. Алгоритм роботи даного методу наведено на схемі (Рис. 2.3).

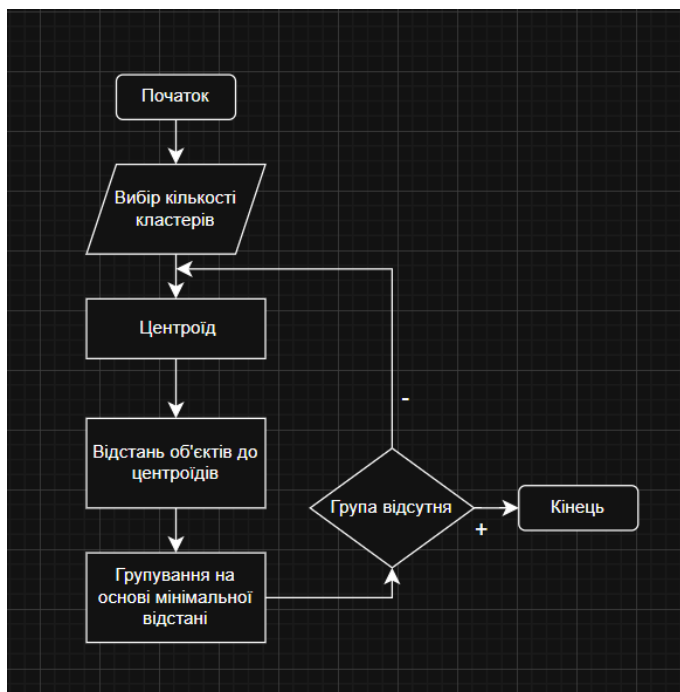


Рис. 2.3 Блок-схема процесу кластеризації k-means

Такі алгоритми дозволяють групувати як готелі, так і користувачів за схожими характеристиками. У сфері курортного туризму це може допомогти виокремити групи готелів, орієнтованих, наприклад, на сімейний відпочинок, молодіжні поїздки або SPA-відпочинок.

Колаборативний компонент системи працює з матрицею «користувач – готель», яка відображає взаємодії користувачів: бронювання, перегляди або оцінки. На основі цієї матриці визначається, які користувачі мають подібну історію взаємодій, а також які готелі часто цікавлять схожих користувачів. Це дозволяє формувати рекомендації навіть тоді, коли готелі суттєво відрізняються за своїми характеристиками, але виявляють спільні закономірності на рівні поведінки користувачів. Відповідно гібридна модель поєднує результати, отримані з контентної та колаборативної частин. При цьому, наприклад, якщо

користувач новий і ще не має історії взаємодій, вагу колаборативної частини можна зменшити, щоб уникнути нерелевантних прогнозів.

Отже, незважаючи на те, що кожен із використаних методів має певні обмеження, їх поєднання у межах гібридної моделі дозволяє мінімізувати недоліки кожного окремого підходу та підвищує якість підсумкових рекомендацій.

2.3 Побудова та проєктування моделі рекомендаційної системи

Так як проєктування рекомендаційної системи для бронювання курортних готелів передбачає інтеграцію кількох підходів, важливо опрацювати кожний етап, адже кожен модуль виконує окрему функцію в загальній логіці системи, а їх поєднання забезпечує стабільність, адаптивність та можливість роботи в умовах розріджених даних, характерних для туристичної галузі.

Контентна модель

Контентна модель ґрунтується на аналізі властивостей готелів і формуванні їх векторних описів, що дає змогу вимірювати подібність між об'єктами на основі атрибутів. Цей підхід є надзвичайно важливим для курортного сегмента, оскільки вибір користувача значною мірою залежить від таких характеристик, як тип відпочинку, наявність інфраструктури, пляжу, SPA-зони, сезонність локації, рівень сервісу чи популярність напрямку.

Побудова контентної моделі розпочинається з формування структурованого набору ознак готелів. Використано як числові дані (кількість зірок, рейтинг, число відгуків), так і категоріальні параметри (країна, місто, тип готелю, перелік доступних зручностей, типи номерів). Оскільки модель має працювати з різномірними типами даних, попередня обробка є ключовим етапом.

Категоріальні характеристики перетворюються у числові за допомогою one-hot encoding, що запобігає створенню штучної ієрархії між значеннями. Іслові ознаки проходять стандартне масштабування, завдяки чому всі

параметри набувають однакового масштабу та рівної ваги під час обчислення подібності.

Фрагмент коду, використаний у прототипі моделі для формування матриці ознак, наведено нижче:

```
# Формуємо набір унікальних готелів
hotels = df[["hotel_id"] +
features].drop_duplicates(subset="hotel_id").copy()

# Кодування категоріальних змінних
hotels_encoded = pd.get_dummies(hotels[features])

# Масштабування числових ознак
scaler = StandardScaler()
hotels_scaled = scaler.fit_transform(hotels_encoded)
```

Отримана матриця ознак дозволяє обчислювати міру схожості між готелями. У рамках прототипу контентної моделі було сформовано матрицю подібності між готелями, яка дозволяє визначити топ об'єктів, що є найбільш схожими до обраного готелю. Дана матриця відображена на рисунку 2.4, а приклад визначення найближчих за характеристиками готелів наведено на рисунку 2.5.

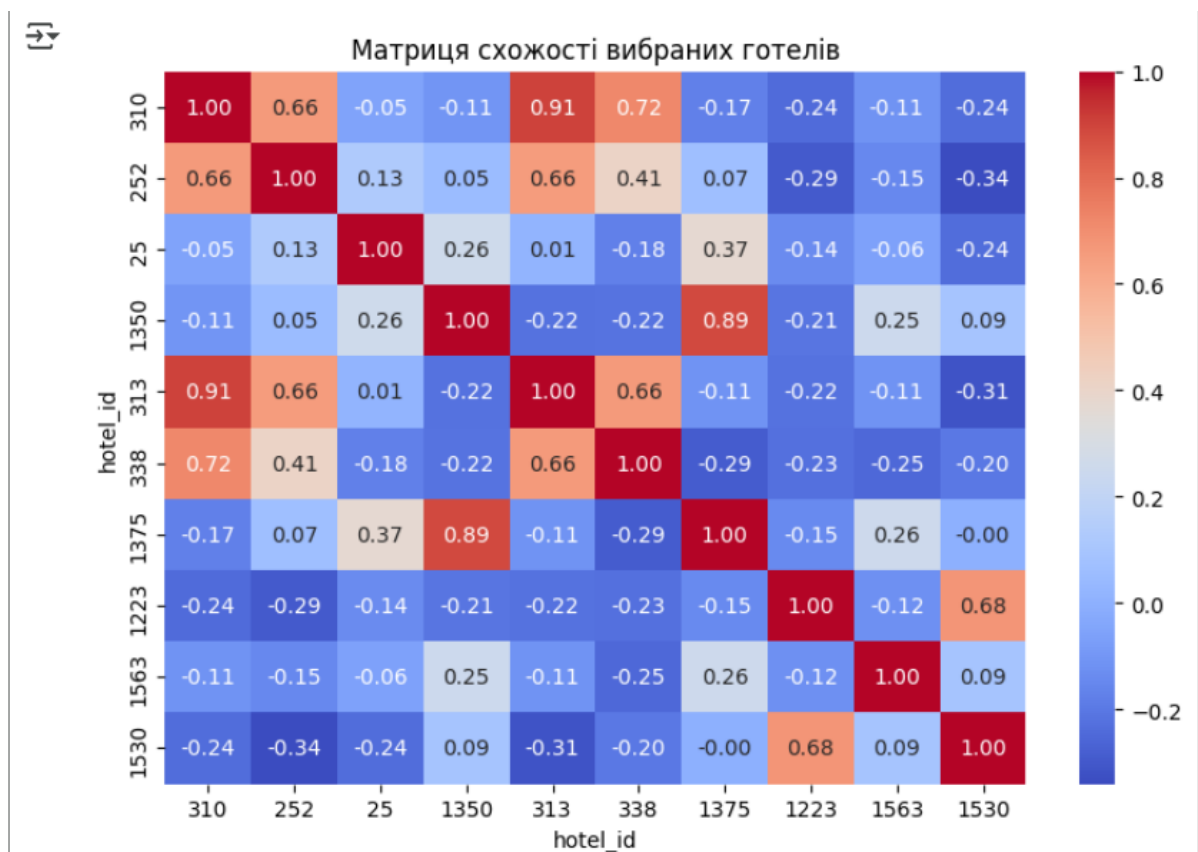


Рис. 2.4 Матриця схожості

Топ-5 схожих готелів до 310:

```
hotel_id
332      0.992764
336      0.984358
337      0.980267
314      0.968506
334      0.952554
Name: 310, dtype: float64
```

Рис. 2.5 Топ подібних готелів

Таким чином, завдяки використанню контентної моделі формуються рекомендації навіть за відсутності історії користувача, а подібність ґрунтується на чітких ознаках, що забезпечує прозорість рекомендацій.

Колаборативна модель

Колаборативна модель відповідає за використання поведінкових закономірностей. Такий підхід базується на припущенні, що користувачі, які обирали подібні готелі в минулому, найімовірніше будуть зацікавлені у схожих варіантах і надалі. Це особливо актуально для курортної тематики, де стиль відпочинку є визначальним фактором: сімейний, романтичний, пляжний, активний, релаксаційний або SPA-орієнтований.

Першим етапом побудови колаборативної частини є формування матриці взаємодій «користувач–готель», де фіксуються оцінки або факти бронювання. У межах прототипу роль оцінки виконує агрегований рейтинг готелю, що дозволяє відобразити загальну суб'єктивну якість об'єкта. Хоча така оцінка не є індивідуальною, вона дає змогу зафіксувати факт взаємодії користувача з певним об'єктом і формує основу для даного методу фільтрації.

Фрагмент коду, використаний на етапі проєктування моделі, наведений нижче:

```
# Формуємо матрицю користувач-готель
user_hotel_matrix = ratings_df.pivot_table(index='user_id',
columns='hotel_id', values='user_rating').fillna(0)

# Обчислюємо схожість між користувачами
user_similarity = cosine_similarity(user_hotel_matrix)
user_similarity_df = pd.DataFrame(user_similarity,
index=user_hotel_matrix.index, columns=user_hotel_matrix.index)
```

У моделі використано косинусну подібність, що добре працює у розріджених даних, властивих системам бронювання, адже більшість користувачів не взаємодіє активно з більшістю об'єктами. На основі матриці подібності між користувачами формується user-user similarity map, візуалізована у вигляді теплової карти (Рис. 2.6). Для окремого користувача виділяється список найбільш подібних користувачів (Рис. 2.7), що слугують основою для прогнозування оцінок готелів, з якими він ще не взаємодіяв.

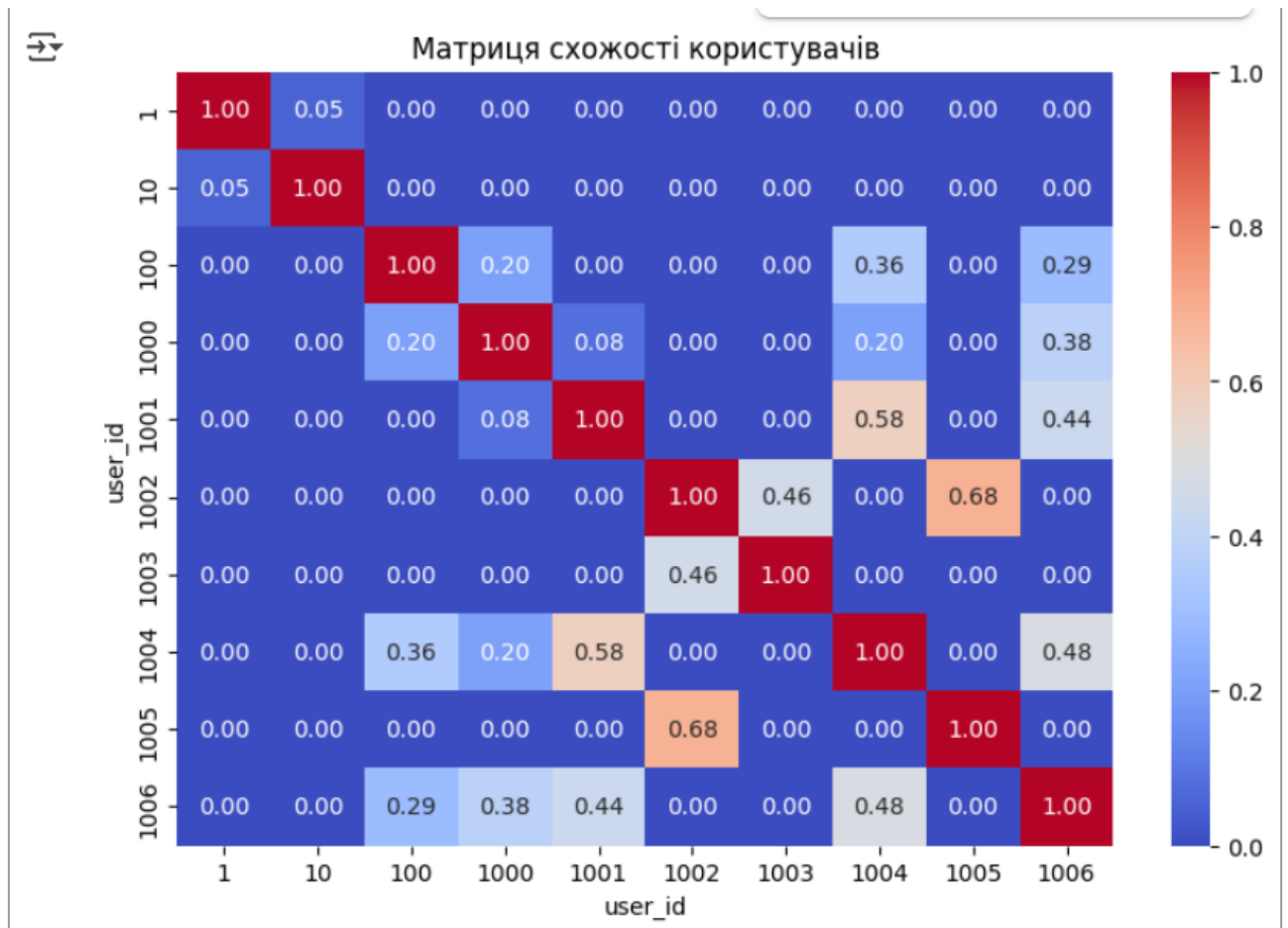


Рис. 2.6 Матриця схожості користувачів

Топ-5 схожих користувачів для 1:

```
user_id
699      0.476683
1069     0.364027
1485     0.364027
634      0.364027
119      0.279653
Name: 1, dtype: float64
```

Рис. 2.7 Топ подібних користувачів до 1

Отже, дана модель дозволяє переносити вподобання схожих користувачів та рекомендувати готелі, які можуть кардинально відрізнятися за характеристиками, але мають високий попит у подібних туристів.

Гібридна модель

Гібридна модель об'єднує результати контентного та колаборативного модулів у єдину оцінку релевантності готелю. На відміну від класичних підходів, де ваги є фіксованими коефіцієнтами (наприклад, 0.5 і 0.5), у даному дослідженні реалізовано адаптивну гібридну оцінку, яка змінює баланс між методами залежно від доступних даних.

Класична формула має вигляд:

$$\text{score} = 0.5 \cdot \text{content} + 0.5 \cdot \text{collaborative}$$

Однак вона не враховує реальні умови роботи системи, такі як відсутність історії користувача, сезонність бронювань, різну густоту поведінкових даних та нерівномірність структури вибору. Саме тому запропоновано адаптивний коефіцієнт α , залежний від характеристик користувача (кількості попередніх бронювань, різноманітності оцінених готелів та розрідженості матриці user-hotel та наявності сезонних даних).

Гібридна оцінка описується формулою:

$$\text{score} = \alpha \cdot CB + (1 - \alpha) \cdot CF$$

, де CB – контентна оцінка, CF – колаборативна оцінка, а α – адаптивна вага, що визначається параметрами користувача.

Таким чином, якщо користувач новий, домінує контентна модель, якщо користувач має велику історію, домінує колаборативна частина. Відповідно якщо несезонність система спирається на статичні контентні ознаки, якщо користувач обирає вузький клас готелів – колаборативна частина стає менш надійною. Це створює персоналізоване, стабільне та адаптивне комбінування.

Більше переваг наведено у таблиці:

Таблиця 2.1

Перевага	Пояснення
Розв'язання cold-start	Нові користувачі отримують контентні

	рекомендації.
Адаптивність до досвідчених користувачів	При великій історії домінує колаборативна частина.
Урахування сезонності	У сезон поведінкові дані підсилюються, матриця user-hotel стає густішою.
Відповідність специфіці курортної сфери	Різні типи туристів отримують релевантні профілі.
Зменшення впливу шумових даних	При низькій якості поведінкових даних модель переходить до контенту.

На рисунку 2.8 подано структурну схему гібридної моделі, де контентний та колаборативний модулі передають свої оцінки у шар адаптивного зважування, після чого формується підсумкова оцінка та top-N рекомендацій.

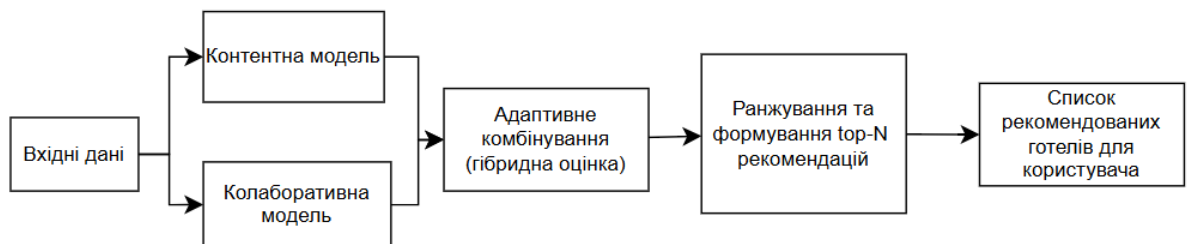


Рис. 2.8 Схема гібридної моделі

Таким чином, гібридний модуль забезпечує збалансоване, персоналізоване та стійке до розрідженості даних ранжування, що робить його найбільш адаптивним рішенням для систем бронювання курортних готелів.

2.4 Оцінка точності та аналіз результатів моделі

Оцінка точності побудованої рекомендаційної системи є ключовим етапом дослідження, оскільки дозволяє підтвердити адекватність застосованих математичних методів, визначити ефективність окремих модулів та оцінити переваги гібридного підходу порівняно з класичними моделями. У межах

дослідження проведено оцінювання на основі тестових даних, що містили історію взаємодій користувачів та набір атрибутів курортних готелів. Це дало змогу дослідити поведінку моделі в умовах, характерних для туристичної галузі, а саме високої розрідженості матриці «користувач–готель», сезонності даних та значної нерівномірності в обсязі історії різних користувачів.

Оцінка точності прогнозування

Так як колаборативна частина моделі відповідає за прогнозування оцінок готелів на основі схожості між користувачами, для оцінювання застосовано дві класичні метрики: середня абсолютна помилка прогнозу (MAE) та середньоквадратична помилка (RMSE). Ці метрики відображають, наскільки точно модель здатна відтворювати реальні оцінки на основі поведінкових даних.

Фрагмент коду, використаний у прототипі, наведено нижче:

```
from sklearn.metrics import mean_squared_error, mean_absolute_error
mse = mean_squared_error(true_ratings, pred_ratings)
rmse = np.sqrt(mse)
mae = mean_absolute_error(true_ratings, pred_ratings)
```

Отриманні значення RMSE та MAE (Рис. 2.9) демонструють, що прогнозування виконується в умовах значної розрідженості даних, адже більшість користувачів мають мінімальну кількість історичних взаємодій, а інформація про готелі часто неповна. Незважаючи на це, гібридна модель показує кращі результати та вищу стабільність порівняно з окремими контентною та колаборативною частинами. Це пояснюється тим, що комбінування двох незалежних джерел інформації частково компенсує неточності поведінкового прогнозу та дозволяє згладити вплив шумових даних.

RMSE: 5.489742546743264
MAE: 5.048098502425794

Рис. 2.9 Показники точності прогнозування

Оцінка якості рекомендацій

Для рекомендаційних систем важливим є не лише прогнозування числових оцінок, а й якість формування списку Тор-К готелів. Для цього було

застосовані метрики Precision@K (частка релевантних готелів серед рекомендованих) та Recall@K (частка релевантних готелів, які система змогла знайти серед усіх можливих). Для K у цьому дослідженні використано 5.

Відповідно, релевантними вважаються ті готелі, з якими користувач взаємодіяв у тестовій вибірці, що відповідає реальному сценарію перевірки рекомендацій у сфері туризму.

Фрагмент коду для обчислення:

```
def precision_recall_at_k(user_id, k=5):
    true_hotels = set(test_by_user[user_id])
    recs = cache[user_id].sort_values(ascending=False).head(k)
    recommended_hotels = set(recs.index)

    TP = len(true_hotels & recommended_hotels)
    FP = len(recommended_hotels - true_hotels)
    FN = len(true_hotels - recommended_hotels)

    precision = TP / (TP + FP + 1e-9)
    recall = TP / (TP + FN + 1e-9)

    return precision, recall
```

Результати Precision@K та Recall@K, наведені на рисунку 2.10, демонструють, що гібридна модель формує значно релевантніші списки рекомендацій порівняно з окремими складовими. Це підтверджує її здатність адаптуватися до різних типів користувачів, як нових, так і тих, що мають значний обсяг історії.

```
Precision@5: 0.0022727272727272724
Recall@5: 0.007102272721235795
```

Рис. 2.10 Показники якості рекомендацій

Оцінка якості ранжування

У рекомендаційних системах туристичного спрямування найбільший вплив мають саме перші позиції у списку рекомендацій, оскільки саме на них припадає основна частка кліків та бронювань. Для аналізу порядку елементів у списку застосовано метрику NDCG@K (Normalized Discounted Cumulative Gain).

Фрагмент коду:

```
def ndcg_k(user_id, k=5):
    true_hotels = set(test_by_user[user_id])
    recs = cache[user_id].sort_values(ascending=False).head(k)

    gains = [1 if h in true_hotels else 0 for h in recs.index]
    discounts = [1 / np.log2(i+2) for i in range(len(gains))]

    dcg = np.sum(np.array(gains) * np.array(discounts))
    ideal_gains = sorted(gains, reverse=True)
    idcg = np.sum(np.array(ideal_gains) * np.array(discounts))

    return dcg / (idcg + 1e-9)
```

Отримані значення NDCG (Рис. 2.11) свідчать про те, що перші позиції списків рекомендуються коректно, проте точність ранжування зростає для користувачів із більшою кількістю історичних взаємодій. Це підтверджує ефективність адаптивного вагового компонента гібридної моделі, який збільшує роль контентного блоку для нових користувачів та посилює колаборативний компонент для досвідчених.

NDCG@5: 0.007169656283415088

Рис. 2.11 Показники якості ранжування

Оцінка охоплення та різноманітності

На відміну від класичних прогнозних задач, рекомендаційні системи повинні не лише бути точними, а й забезпечувати різноманітність та охоплення, особливо у сфері туризму, де користувачі очікують індивідуальні та унікальні варіанти.

Охоплення (coverage) показує, яку частку всіх доступних готелів система здатна рекомендувати хоча б одному користувачу. Різноманітність рекомендацій (diversity) обчислюється як середня відмінність наборів рекомендацій між різними користувачами.

Фрагмент коду формування оцінок:

```
coverage = len(recommended_hotels_all) / len(hotels_matrix.index)
diversity = np.mean(div_scores)
```

Результат, наведений на рисунку 2.12, показує, що отримане значення coverage є задовільним і відображає властивості туристичних даних, наприклад високу сезонність, нерівномірну популярність напрямків та кластеризацію готелів. Високе значення diversity свідчить про те, що система не створює однакових списків для різних користувачів, а отже адаптивні ваги забезпечують персоналізацію, а контентний модуль враховує унікальні особливості курортних готелів.

Coverage: 0.07174638487208009
Diversity: 0.9229478458161483

Рис. 2.12 Показники охоплення та різноманітності

На рисунку 2.13 наведено гістограму розподілу різноманітності між користувачами. Більшість значень знаходиться в діапазоні 0.85-1.0, що означає майже повну неперетинність рекомендацій та високий ступінь персоналізації. Невелика кількість низьких значень (0.0-0.3) пояснюється окремими випадками відсутності історії або схожістю профілів певних користувачів, що є типовим для розріджених туристичних даних. Такий правосторонній розподіл підтверджує здатність моделі формувати унікальні рекомендації навіть за обмеженої інформації.

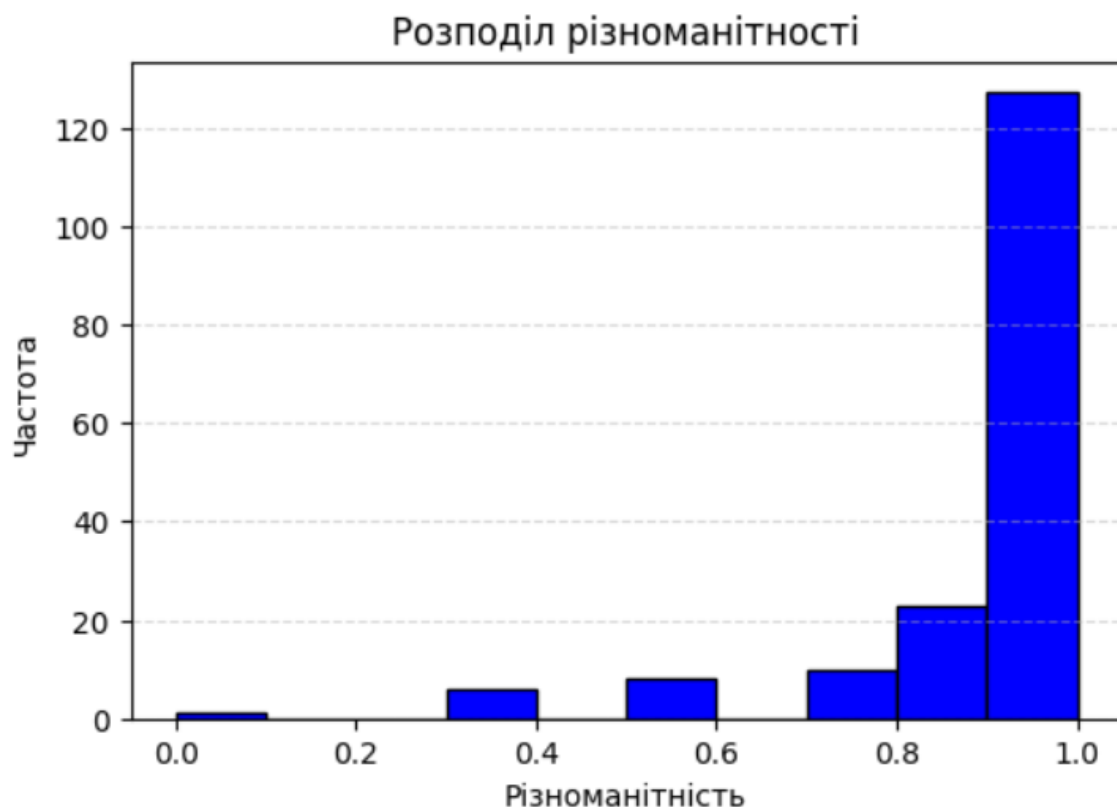


Рис. 2.13 Розподіл різноманітності рекомендацій між користувачами

Отже, проведене оцінювання підтвердило ефективність побудованої гібридної моделі та доцільність її використання в реальній онлайн-платформі. У порівнянні з окремими контентними та колаборативними підходами вона демонструє вищу точність прогнозування, кращу якість формування списків, більш коректне ранжування популярних готелів, ширше охоплення та більшу різноманітність рекомендацій. Таким чином, адаптивне вагове комбінування дозволило врахувати поведінкову насиченість користувачів, їх різноманітність вибору та сезонні коливання, що є важливим для систем бронювання курортних готелів.

Висновки до другого розділу

У другому розділі було сформовано науково обґрунтовану модель рекомендаційної системи для онлайн-платформи бронювання курортних готелів. На основі аналізу вимог, структури вхідних даних та специфіки туристичної галузі побудовано кілька взаємопов'язаних компонентів моделі, а

саме контентний, колаборативний та гібридний модулі, що разом формують комплексний алгоритм персоналізованого підбору готелів.

Було визначено структурну архітектуру системи, описано логіку проходження даних через усі етапи. Такий підхід дозволив чітко окреслити взаємозв'язки між компонентами моделі та обґрунтувати вибір обчислювальних методів. Також було приділено увагу вибору метрик схожості, методам кодування ознак, нормалізації та особливостям роботи з розрідженими матрицями, характерними для систем бронювання.

Запропонований адаптивний коефіцієнт ваги змінює баланс між складовими моделі враховуючи досвід користувача, сезонні коливання та структуру даних. Такий підхід дозволяє підвищити стійкість рекомендацій у реальних умовах функціонування туристичних платформ. Відповідні оцінки підтвердили коректність побудованої моделі, її здатність працювати з розрідженими даними та формувати персоналізовані й варіативні рекомендації.

Отже, в загальному було сформовано та досліджено повноцінну модель рекомендаційної системи, адаптовану до потреб курортної сфери, яка створює основу для подальшої програмної реалізації та інтеграції в онлайн-платформу бронювання готелів.

РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ

3.1 Основні вимоги до системи та умови експлуатації Розроблений програмний прототип рекомендаційної системи для бронювання курортних готелів є веб-орієнтованою платформою, що складається з серверної частини, клієнтського інтерфейсу та окремого модуля інтелектуальних рекомендацій, у якому реалізовано результати математичного моделювання, описані раніше. Для забезпечення коректної роботи системи необхідно визначити вимоги до функціональності, середовища виконання та обмеження, що впливають з характеру даних туристичної галузі.

Функціональні вимоги

Система повинна підтримувати основні сценарії взаємодії користувача з платформою, а також забезпечувати формування персоналізованих рекомендацій у режимі реального часу. До основних функціональних вимог належать:

1. Реєстрація та автентифікація користувачів, де відбувається створення облікових записів, авторизація з використанням JWT-токена та доступ до даних акаунта відповідно до ролі (користувач, власник готелю, адміністратор);
2. Пошук та перегляд інформації про готелі із можливістю здійснювати пошук за країною, містом, категорією готелю, фільтрувати за характеристиками, а також переглядати детальні сторінки з описом та відгуками;
3. Бронювання номерів, що включає перевірку доступності на обрані дати, створення та управління бронюваннями, враховуючи бонуси лояльності;
4. Модуль рекомендацій, в якому компонент формує персоналізований список готелів на основі історії взаємодій, подібності користувачів, атрибутів готелів та гібридної оцінки. Рекомендації оновлюються при появі нових оцінок або бронювань;

5. Управління готелями для власників передбачає CRUD-операції для керування інформацією про готелі, номери та їх характеристики.

Нефункціональні вимоги

Для забезпечення стабільності та якості роботи створеного прототипу було сформульовано низку нефункціональних вимог. Зокрема, система має відповідати вимогам продуктивності та масштабованості, час відповіді API у типовому режимі не повинен перевищувати 1 секунди при навантаженні до 20 паралельних користувачів, а модуль рекомендацій повинен повертати результат не довше ніж за 0.5 секунди при роботі із типовим набором даних. Архітектура системи повинна підтримувати горизонтальне масштабування та окреме розміщення модуля рекомендацій.

Вимоги надійності передбачають безперервну роботу системи за умов часткової відсутності даних, коректну роботу рекомендаційного модуля у cold-start сценаріях та наявність механізмів обробки помилок, що запобігають збоєм у роботі серверної частини.

Значна увага має бути приділена безпеці. У системі застосовується хешування паролів за алгоритмом bcrypt, авторизація та автентифікація реалізовані на основі JWT, а доступ до ресурсів контролюється відповідно до ролей користувачів. Крім того, архітектура серверної частини побудована таким чином, щоб забезпечити захист від SQL-ін'єкцій.

Окремою вимогою є підтримуваність та модульність програмного комплексу. Проєкт має чітку структуру, що включає окремі рівні routers, services та models. Такий підхід дозволяє оновлювати рекомендаційний модуль без зміни інших компонентів системи, забезпечує ізоляцію бізнес-логіки від інтерфейсної частини та полегшує розширення платформи у майбутньому.

На основі сформульованих вимог також визначено програмне середовище та апаратні обмеження. Серверна частина потребує Python 3.10+, FastAPI, SQLAlchemy 2.0, Uvicorn або Gunicorn для розгортання, а також бібліотеки Scikit-learn, NumPy та Pandas для реалізації рекомендаційного алгоритму. Клієнтська частина використовує Node.js 18+, React.js та Axios. В межах

прототипу застосовано реляційну базу SQLite. Для повноцінного розгортання рекомендовані мінімальні апаратні ресурси сервера мають бути 2 CPU, 4 GB RAM, 20 GB SSD.

За результатом аналізу вимог для формування архітектури системи були створені UML-діаграми, що відображають логіку взаємодії компонентів.

Діаграма сутностей (Рис. 3.1) класів демонструє ключові сутності, такі як користувач, готель, номер, бронювання, відгуки та пов'язані з ними атрибути. Вона відображає типи зв'язків між сутностями, наприклад one-to-many чи many-to-many, що визначають правила формування даних у системі. Таким чином дана діаграма показує логіку обробки інформації та взаємодію об'єктів.

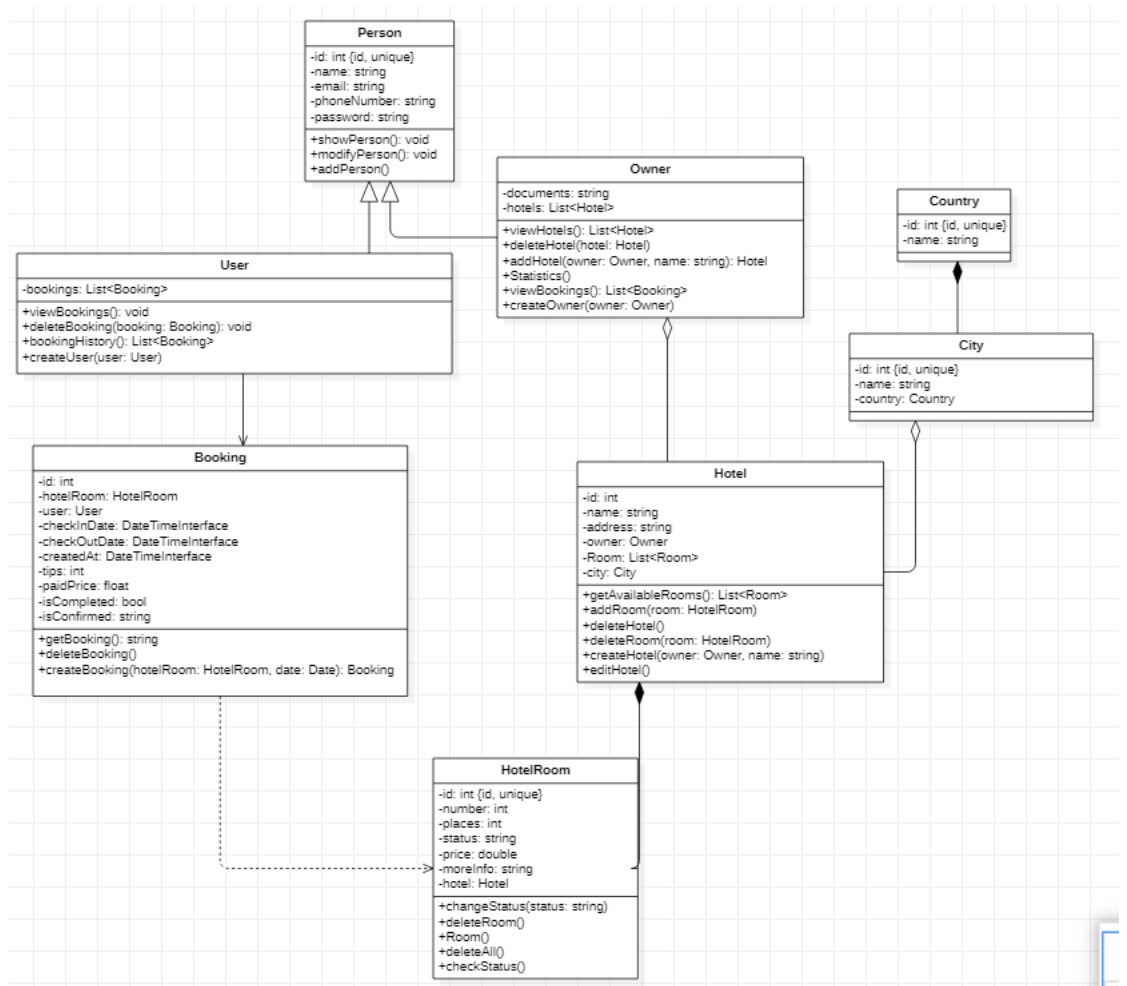


Рис. 3.1 Діаграма сутностей класів

Діаграма розгортання (Рис. 3.2) демонструє структуру системи, відображаючи вузли, на яких виконується клієнтська частина, сервер FastAPI,

база даних. Саме це показує як система функціонує в реальному середовищі та яким чином забезпечується масштабованість.

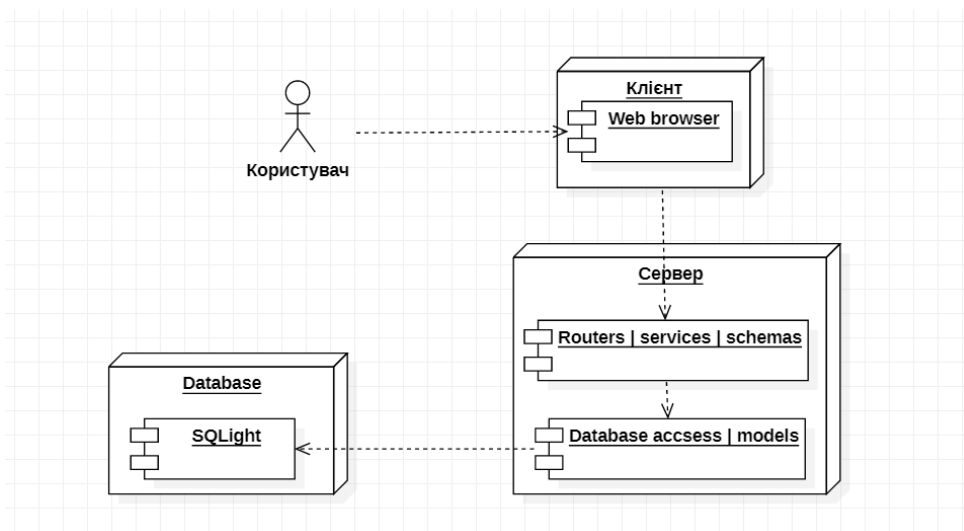


Рис. 3.2 Діаграма розгортання платформи

Діаграма класів для бізнес-логіки (Рис. 3.3) відображає об'єктно-орієнтовану структуру та показує, як основні елементи представлені у вигляді класів із відповідними полями, зв'язками та асоціаціями. Діаграма демонструє логічні залежності між класами, які використовуються у моделі застосунку. Кожен клас містить набір атрибутів, що точно відображають його функціональне призначення в бізнес-логіці. Дана структура є важливою для коректної роботи рекомендаційної системи, оскільки вона використовує дані різних класів коректної роботи рекомендаційної системи, оскільки вона використовує дані різних класів.

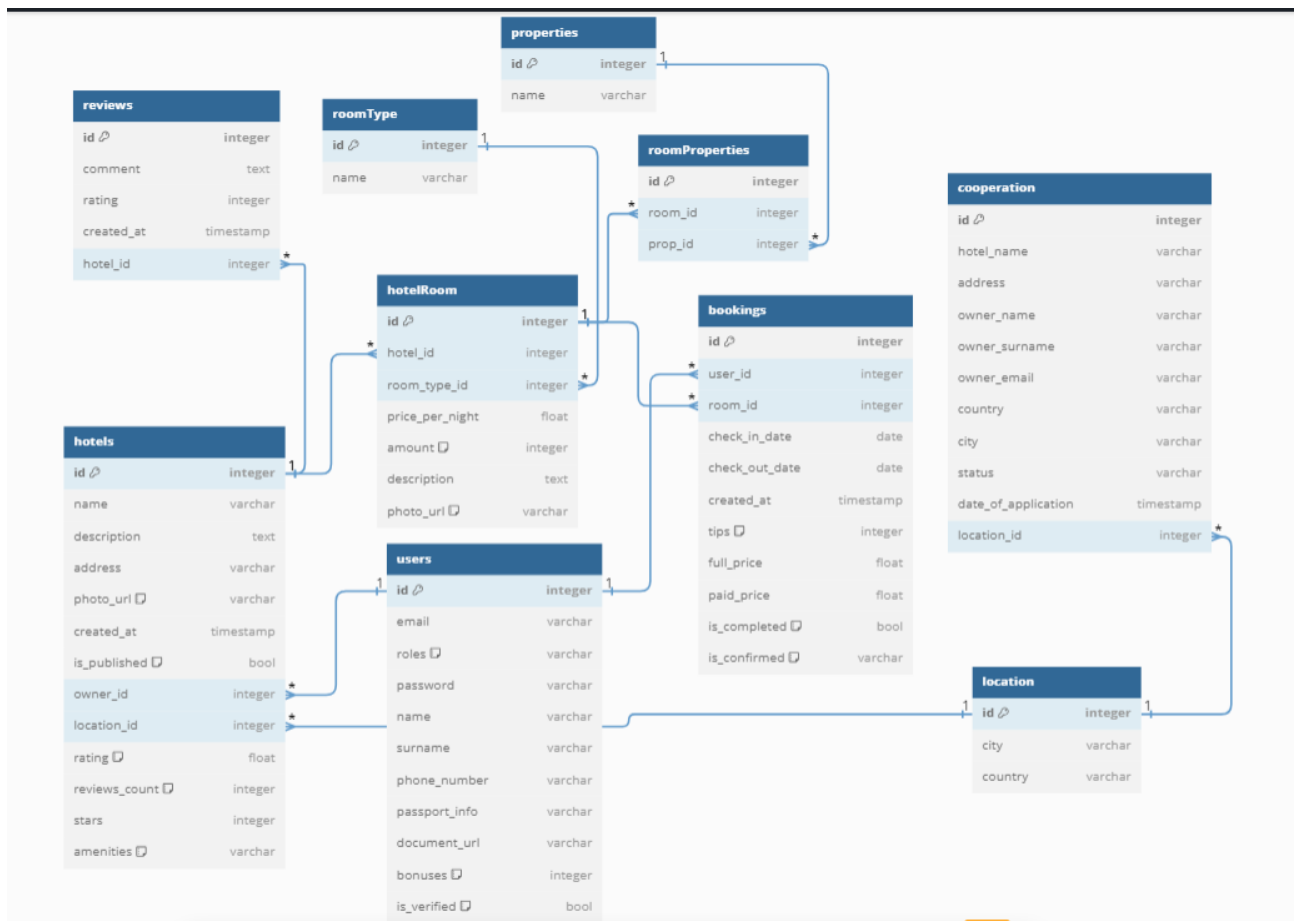


Рис. 3.2 Діаграма класів системи для бронювання готелів

Система також має низку обмежень, характерні для туристичної сфери. В першу чергу це розрідженість матриці user-hotel, оскільки більшість користувачів залишають мало оцінок, та проблема «холодного старту» для нових користувачів, де колаборативна модель може працювати неточно. Також наявні сезонні коливання, що впливають на активність у різних категоріях курортів.

Отже, так як результати залежать від історії чи зміни пори року, для користувачів, які не здійснили чи завершили лише одне бронювання для різного типу курорту, колаборативний метод працює погано. Саме тому потрібно використовувати адаптивні ваги у гібридній моделі.

3.2 Обґрунтування вибору засобів реалізації

Розробка програмного прототипу рекомендаційної системи для онлайн-платформи бронювання курортних готелів потребувала вибору технологій, які

забезпечують стійкість роботи алгоритмів у умовах розріджених даних, підтримують інтелектуальний аналіз та дозволяють інтегрувати рекомендаційну модель у веб-орієнтоване середовище. Під час вибору інструментів враховувалися такі критерії: відповідність науковим цілям роботи, продуктивність, можливість масштабування, відкритість технологій, простота розвитку системи та підтримуваність коду.

Основною мовою програмування було обрано Python, який є стандартом у сфері машинного навчання та наукових обчислень. Python має широкую екосистему бібліотек, що забезпечують роботу з великими масивами даних, статистичний аналіз, моделювання та побудову рекомендаційних систем. Такий вибір забезпечує прозоре відтворення алгоритмів, можливість швидкого тестування різних підходів та реалізацію адаптивного гібридного методу рекомендацій.

На етапі розробки моделі було розглянуто TensorFlow та PyTorch як засоби для реалізації глибинних нейронних мереж. Проте для задач сфери курортних готелів, де матриця взаємодій є значно розрідженою та містить обмежений обсяг позначених даних, застосування високорівневих нейромережових моделей є недоцільним. Тому основним інструментом для реалізації алгоритмів обрано Scikit-learn, оскільки ця бібліотека:

- підтримує методи нормалізації ознак, необхідні для формування якісної контентної моделі;
- містить ефективні алгоритми обчислення міри схожості, що є основою контентної та колаборативної фільтрації;
- працює зі структуровано-розрідженими матрицями без суттєвого зниження продуктивності;
- дозволяє поєднувати декілька моделей у гібридний підхід;
- не потребує тривалого процесу навчання, що важливо у системах із динамічним оновленням даних.

Таким чином, Python у поєднанні зі Scikit-learn забезпечує оптимальну основу для побудови адаптивної гібридної моделі рекомендацій, що відповідає науковим та технічним вимогам.

Відповідно для створення серверної частини платформи обрано FastAPI [32], який забезпечує високу продуктивність завдяки асинхронній архітектурі, автоматичну генерацію документації API (Swagger [33]/OpenAPI) та просту інтеграцію з Python-модулями. FastAPI є зручним для побудови мікросервісів, що дозволяє розмістити рекомендаційний модуль як окремий сервіс без блокування інших компонентів системи.

Роботу з базою даних реалізовано за допомогою SQLAlchemy, що забезпечує ORM-підхід, чітке розділення бізнес-логіки від SQL-інструкцій та можливість гнучкого керування транзакціями. У рамках прототипу використано SQLite як легку СУБД, достатню для тестових сценаріїв. Для реального розгортання передбачено використання PostgreSQL, що відповідає вимогам до обробки паралельних запитів та роботи з великим обсягом структурованих даних.

Клієнтський інтерфейс платформи реалізовано на React.js, який забезпечує компонентний підхід, високу продуктивність завдяки віртуальному DOM та гнучкі механізми побудови інтерфейсу для веб-застосунків. Для роботи з API використовується бібліотека Axios, що дозволяє стабільно виконувати HTTP-запити та застосовує JWT-авторизацію.

Отже, вибір інструментальних засобів був зумовлений необхідністю поєднання наукової складової моделі та реальних вимог до веб-платформи бронювання курортних готелів. Python і Scikit-learn забезпечили основу для побудови адаптивної рекомендаційної моделі, тоді як FastAPI, SQLAlchemy та React.js сформували продуктивне, модульне та масштабоване програмне середовище для її інтеграції.

3.3 Проєктування та реалізація модулів системи

Створений програмний прототип веб-платформи для бронювання курортних готелів реалізує всі основні функції предметної області: пошук, фільтрацію, управління готелями, роботу з відгуками та бронюваннями, та доповнюється окремим модулем рекомендаційної системи, який реалізує алгоритмічні рішення, описані у другому розділі. Система реалізована за архітектурою клієнт-сервер, де серверна частина працює на FastAPI (Python), а клієнтська – на React із використанням Axios та Material UI.

Загальна структура та огляд основних модулів серверної частини

Серверна частина організована у вигляді окремого бекенд-продукту з модульною структурою каталогів, що забезпечує розділення бізнес-логіки, роботи з базою даних та реалізації API-інтерфейсу. Структура проєкту має вигляд:

- `main.py` – точка входу, підключення роутерів, конфігурація FastAPI;
- `database.py` – створення підключення до СУБД (SQLite для прототипу), створення `SessionLocal`;
- `models.py` – визначення ORM-моделей SQLAlchemy: `Users`, `Hotels`, `HotelRooms`, `Bookings`, `Reviews`, `Locations`, `Properties`;
- `auth.py`, `users.py` – авторизація користувачів, створення та перевірка JWT-токенів, контроль доступу згідно з ролями;
- `schemas/` – Pydantic-схеми для валідації вхідних і вихідних даних;
- `routers/` – окремі роутери для логічних підсистем (`hotels`, `hotel_rooms`, `bookings`, `reviews`, `locations`, `properties`, `room_types`, `cooperations`, `search`, `upload`, `recommendations`);
- `services/` – сервісна логіка, зокрема модуль `recommendation_service.py`.

Таке відокремлення забезпечує гнучкість розробки, дозволяючи змінювати чи тестувати окремі підсистеми незалежно від серверного каркасу.

Модуль бронювання, у роутері `bookings`, разом із сервісами реалізує повний цикл операцій, таких як створення бронювання користувачем, перевірка

доступності номерів на вибрані дати, перегляд активних і завершених бронювань, підтвердження або скасування заявок власником готелю.

Таблиця Booking містить `user_id`, `hotel_room_id`, дату заїзду та виїзду, вартість, бонуси та статуси бронювання. Ці дані є ключовими для формування матриці «user-hotel», що використовується колаборативною складовою рекомендаційної системи.

Модулі управління готелями та номерами (`hotels` та `hotel_rooms`) відповідають за створення та редагування готелів, керування номерами, ціновими планами, наповнення, фото, публікацію чи приховування готелів. Тут задіяні моделі `Hotels`, `Hotel_rooms`, `Room_types`, `Properties` і таблицю зв'язку «багато-до-багатьох» між номерами та властивостями. Для рекомендаційної моделі саме ці дані є джерелом контентних ознак, таких як кількість зірок, рейтинг, кількість відгуків, країна, місто, перелік зручностей, типи номерів.

Модулі локацій та властивостей (`locations` та `properties`) забезпечують створення списку країн і міст із перевіркою їх існування та додавання властивостей номерів (балкон, ванна, кондиціонер, SPA-зона тощо). У рекомендаційній системі ці категорії використовуються після `one-hot encoding` та входять до фінального вектору ознак готелів.

Модуль `users` реалізує систему ролей, де `ROLE_USER` – звичайний користувач, який отримує рекомендації, `ROLE_OWNER` – власник готелю, який керує своїми об'єктами, та `ROLE_ADMIN` – адмін платформи, який обробляє заявки на співпрацю.

Модуль відгуків реалізує додавання користувацьких оцінок, збереження текстових коментарів та оновлення агрегованих показників `rating` та `reviews_count` у таблиці `Hotels`. Відповідно ці значення використовуються і як контентні ознаки, і як `fallback`-індикатори у випадку відсутності особистої історії користувача.

Модуль співпраці підтримує бізнес-логіку підключення нових готелів до платформи, що безпосередньо впливає на розширення бази даних і, відповідно, на якість та обсяг інформації, доступної рекомендаційній системі.

Таким чином, всі модулі бекенду формують єдиний інформаційний простір, у якому дані користувачів, готелів, бронювань та властивостей використовуються рекомендаційним сервісом як джерело для побудови контентних та поведінкових моделей.

Загальна структура backend та frontend наведена в Додатку А.

Модуль рекомендаційної системи

Основним результатом роботи є окремий сервіс RecommendationService, який реалізує гібридну модель рекомендацій та інтегрується через роутер.

Фрагмент коду:

```
router = APIRouter(prefix="/api/recommendations", tags=["recommendations"])
```

Ендпоінт отримує поточного користувача через JWT-токен, створює екземпляр RecommendationService з активною сесією БД та в результаті повертає JSON зі списком рекомендованих готелів.

Відповідно RecommendationService використовує ORM-моделі SQLAlchemy для зчитування даних із таблиць для ідентифікації користувача, ролі, історії бронювань, отримання базових атрибутів готелів, додаткових властивостей та розташування. На основі цих даних і формується контентна матриця ознак готелів.

Фрагмент коду формування матриці:

```
df = pd.get_dummies(base[["stars", "rating", "reviews_count",  
"country", "city", "amenities", "property"]], columns=["country",  
"city"], drop_first=True)  
df = self._get_dummies_text(df, "amenities")  
df = self._get_dummies_text(df, "property")  
  
# масштабування числових  
df = self._standard_scale(df, ["stars", "rating", "reviews_count"])  
  
hotels_matrix = pd.DataFrame(df, index=hotels["hotel_id"])
```

Після цього обчислюється косинусна подібність.

Фрагмент коду:

```
sim = cosine_similarity(hotels_matrix)  
similarity = pd.DataFrame(sim, index=hotels_matrix.index,  
columns=hotels_matrix.index)
```

Отримана оцінка показує, наскільки кожен готель схожий на ті, що вже цікавили користувача.

Колаборативна складова формує оцінки, які визначаються як зважені рейтинги схожих користувачів.

Фрагмент коду:

```
collab_scores = pd.Series(0.0,
index=user_hotel_matrix.columns.astype(str))
    for hotel in user_hotel_matrix.columns.astype(str):
        other_ratings = user_hotel_matrix[hotel].astype(float)
        common_users =
other_ratings.index.intersection(similar_users.index)
        if len(common_users) == 0:
            continue
        common_users_rating = other_ratings.loc[common_users]
        weight_com_u_rating = similar_users.loc[common_users]
        mask = common_users_rating > 0
        if mask.sum() == 0:
            continue
        weighted_sum = (common_users_rating[mask] *
weight_com_u_rating[mask]).sum()
        sum_weights = weight_com_u_rating[mask].sum()
        collab_scores.at[hotel] = (weighted_sum / sum_weights) if
sum_weights > 0 else 0.0
```

Під час застосування адаптивної гібридної оцінки для користувачів із малою історією та сильно розрідженими даними система надає більшу вагу контентній моделі, яка спирається на властивості готелів, а для досвідчених користувачів із багатою історією взаємодій вага колаборативної частини збільшується, що підсилює роль поведінкових патернів.

Фрагмент обчислення за адаптивною формулою:

```
alpha_raw = 0.2 + 0.4 * ratings_factor + 0.2 * sparsity_factor + 0.2 *
diversity_score
    alpha = max(0.2, min(0.8, alpha_raw))

    content_scores = self._normalize_scores(content_scores)
    collab_scores = self._normalize_scores(collab_scores)

    # гібридна оцінка
    hybrid_scores = alpha * content_scores + (1.0 - alpha) *
collab_scores
```

В результаті, після обчислення гібридних оцінок, із переліку виключаються готелі, які користувач уже бронював чи оцінював, список сортується за спаданням оцінки, повертаючи лише топ.

Клієнтська частина реалізована на React із використанням Axios. Модуль рекомендацій інтегрований так, щоб автоматично виконувати запит після авторизації користувача та відображати список рекомендованих готелів на головній сторінці та на сторінці перегляду готелю чи номеру.

Приклад виклику API на фронтенді:

```
axios.get(`${ApiUrl}/api/recommendations`, userAuthenticationConfig())
  .then((res) => {
    const data = res.data?.recommendations || [];
    setRecommendations(data);
  })
  .catch((err) => console.error("Помилка отримання рекомендацій:",
err));
}, []);
```

Список рекомендацій передається у компоненти інтерфейсу та відображається у вигляді карток готелів. Таким чином, фронтенд і бекенд працюють узгоджено, а гібридна модель безпосередньо впливає на персоналізований порядок результатів, які бачить користувач.

Фрагмент основної частини коду рекомендаційного сервісу наведено в Додатку Б.

3.4 Опис інтерфейсу та сценарії роботи системи

Ключові сторінки та елементи на платформі

Графічний інтерфейс онлайн-платформи EasyStayChoice розроблений відповідно до принципів сучасних веб-систем бронювання та орієнтований на зручність користувача, швидкий доступ до інформації та прозору інтеграцію з модулем рекомендаційної системи. Інтерфейс є адаптивним і коректно масштабується для настільних ПК, планшетів та мобільних пристроїв, забезпечуючи стабільність роботи у різних сценаріях використання.

Головна сторінка (Рис. 3.4) є початковим елементом взаємодії користувача із сервісом. Центральним блоком є пошукова панель із вибором

країни, міста та дат. Додатково відображаються популярні напрямки, добірка готелів із високими рейтингами та персональний блок «Рекомендовані готелі», який активується після авторизації та базується на гібридній моделі рекомендацій.

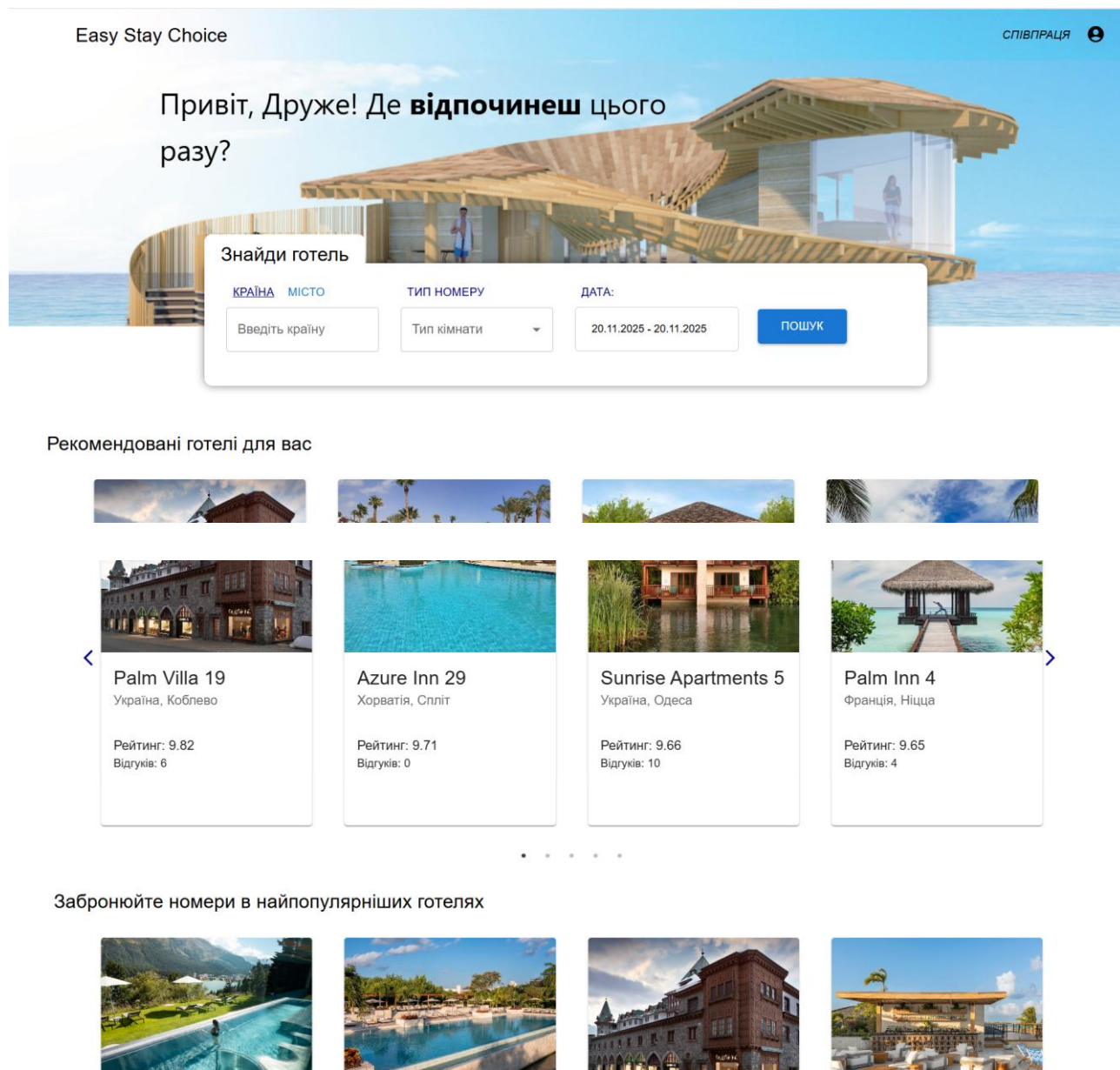


Рис. 3.4 Головна сторінка платформи

Пошукова форма (Рис. 3.5) дозволяє користувачеві задати країну, місто, тип готелю та дату перебування. Після надсилання запиту формується сторінка результатів (Рис. 3.6), яка включає картки номерів з коротким описом, фото та рейтингом, а також фільтри за ціною, рейтингом, типом номерів.

Знайди готель

КРАЇНА МІСТО
Введіть країну
Італія

ТИП НОМЕРУ
Тип кімнати
Standard

ДАТА:
22.11.2025 - 24.11.2025

ПОШУК

Nov 22, 2025 Nov 24, 2025

November 2025

Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

illa 19
Коблево

Azure Inn 29
Хорватія, Спліт

Palm Inn 4
Хорватія, Ніцца

Рис.3.5 Пошукова форма

Фільтрація

☐ Показати готелі

Вартість від 0 до 10000 грн

Тип кімнати

Додаткові властивості

ЗАСТОСУВАТИ

Головна / Пошук

Результат пошуку:

Luxury Apartment номер в Коблево
Palm Villa 19, Sea street, 109, Україна
Рейтинг готелю: 9.82
Відгуків: 6
Ціна: 2642 грн за ніч
ЗАБРОНЮВАТИ

Standard номер в Коблево
Palm Villa 19, Sea street, 109, Україна
Рейтинг готелю: 9.82
Відгуків: 6
Ціна: 1784 грн за ніч
ЗАБРОНЮВАТИ

Рис. 3.6 Сторінка результату пошуку за фільтрами для номерів

Користувач також може обрати перегляд результатів у форматі списку готелів (Рис. 3.7), що забезпечує більшу гнучкість у виборі. Це дозволяє йому більш точно оцінити готель в загальному вигляді, враховуючи його оцінку, кількість відгуків, фасадний вигляд та розташування.

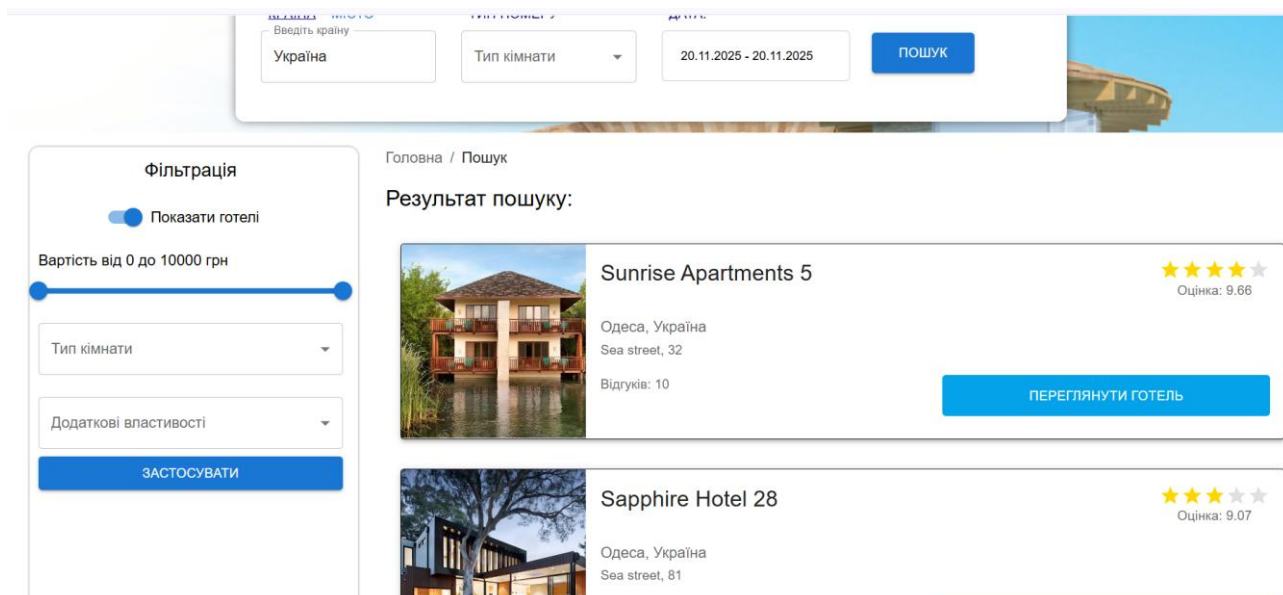


Рис. 3.7 Сторінка результату пошуку за фільтрами для готелів

У випадках, коли користувацький запит дає обмежену кількість результатів, система автоматично додає блок «Схожі готелі» (Рис. 3.8). Він формується на основі контентної частини рекомендаційної моделі та дозволяє заповнити прогалини, що виникають через сезонність або вузькі параметри пошуку.

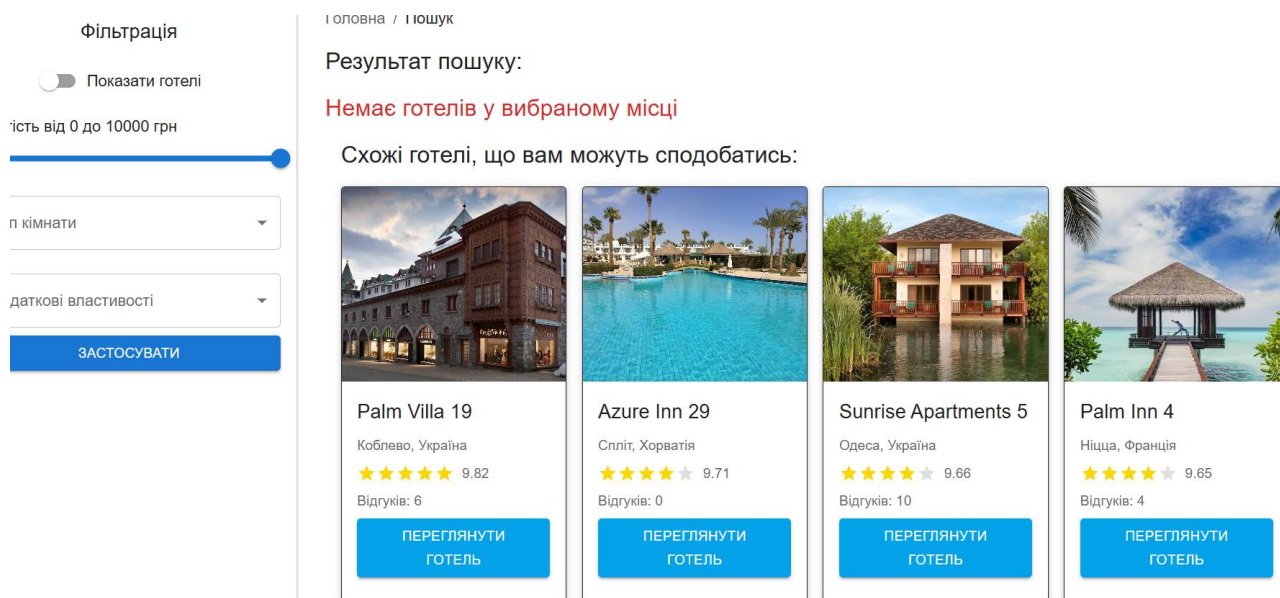


Рис. 3.8 Додатковий блок на сторінці пошуку

Сторінка окремого готелю (Рис. 3.9) містить галерею зображень, детальний опис, перелік зручностей, рейтинг та відгуки користувачів. На цій сторінці інтегровано рекомендаційний блок, який формує список альтернативних готелів.

Palm Villa 19

★★★★★
Рейтинг: 9.82 / 10



Країна: Україна
Місто: Коблево
Адреса: Sea street, 109
Кількість відгуків:

ПЕРЕПЛЯНУТИ ДОСТУПНІ НОМЕРИ

ОПИС ВІДГУКИ

Відгуки



Прекрасний вид з вікна!

16.10.2025 23:54



Прекрасний вид з вікна!

14.10.2025 23:54

ОПИС ВІДГУКИ

Palm Villa — це прекрасний курорт на узбережжі в Барселоні, Іспанія. Комфортні номери, басейн та відмінний сервіс.

Зручності готелю:

Free Wi-Fi

Басейн

Сніданок включено

Зручності готелю:

Free Wi-Fi

Басейн

Сніданок включено

Рекомендуємо

Схожі готелі, що вам можуть сподобатись:



Azure Inn 29

Спліт, Хорватія

★★★★★ 9.71

Відгуків: 0

ПЕРЕПЛЯНУТИ ГОТЕЛЬ



Sunrise Apartments 5

Одеса, Україна

★★★★★ 9.66

Відгуків: 10

ПЕРЕПЛЯНУТИ ГОТЕЛЬ



Palm Inn 4

Ніцца, Франція

★★★★★ 9.65

Відгуків: 4

ПЕРЕПЛЯНУТИ ГОТЕЛЬ



Azure Inn 8

Суцца, Туніс

★★★★★ 9.61

Відгуків: 10

ПЕРЕПЛЯНУТИ ГОТЕЛЬ

Рис.3.9 Сторінка готелю

Сторінка номеру (рис. 3.10) відображає характеристики конкретного номеру, фотографії, доступність на вибрані дати, перелік властивостей та кнопку «Забронювати». При виборі дат система виконує перевірку доступності через бекенд.

Palm Villa 19

★★★★★
Рейтинг: 9.82 / 10

Тип кімнати: Standard

Адреса: Sea street, 109, Коблево, Україна

Ціна за ніч: 1784 грн

Загальна вартість: 3568 грн [РОЗРАХУВАТИ](#)

Виберіть дати

22.11.2025 - 24.11.2025

ЗАБРОНЮВАТИ

Рис. 3.10 Сторінка номеру

Особистий кабінет надає користувачу можливість переглядати персональні дані, історію бронювань із ключовими даними, такі як готель, тип номеру, проміжок часу, на який здійснено бронювання, інформацію по вартості, бонусній програмі, а також статуси заявок із можливістю скасування бронювання, якщо не було погоджено (рис. 3.11).

Easy Stay Choice СПІВПРАЦЯ

Юлія Опінова <

Головна / Бронювання

Мої бронювання

Інше

Профіль

Готель	Номер	Дата	Вартість	До сплати	Нараховано бонусів	Статус
Французе малъ	Трьохмісний (Triple)	29.05.2024 - 31.05.2024	900 грн	850 грн	10	Підтверджено
Four Seasons Resort	Чотиримісний (Quadruple)	03.06.2024 - 06.06.2024	600 грн	590 грн	7	Очікує підтвердження СКАСУВАТИ
Belmond Hotel Splendido	Трьохмісний (Triple)	01.06.2024 - 04.06.2024	510 грн	510 грн	6	Відхилено СКАСУВАТИ

Рис. 3.11 Перегляд бронювань

Також платформа містить інформаційні розділи, такі як бонуси, спеціальні пропозиції, FAQ та контакти, що сприяють кращому ознайомленню з можливостями системи та підтримують зручність для користувачів.

Система передбачає розмежування доступу за ролями. Адміністратор має можливість переглядати заяви на співпрацю (Рис. 3.12), змінювати статуси

заявок, надсилати повідомлення власникам та додавати інформацію про щойно зареєстровані готелі.

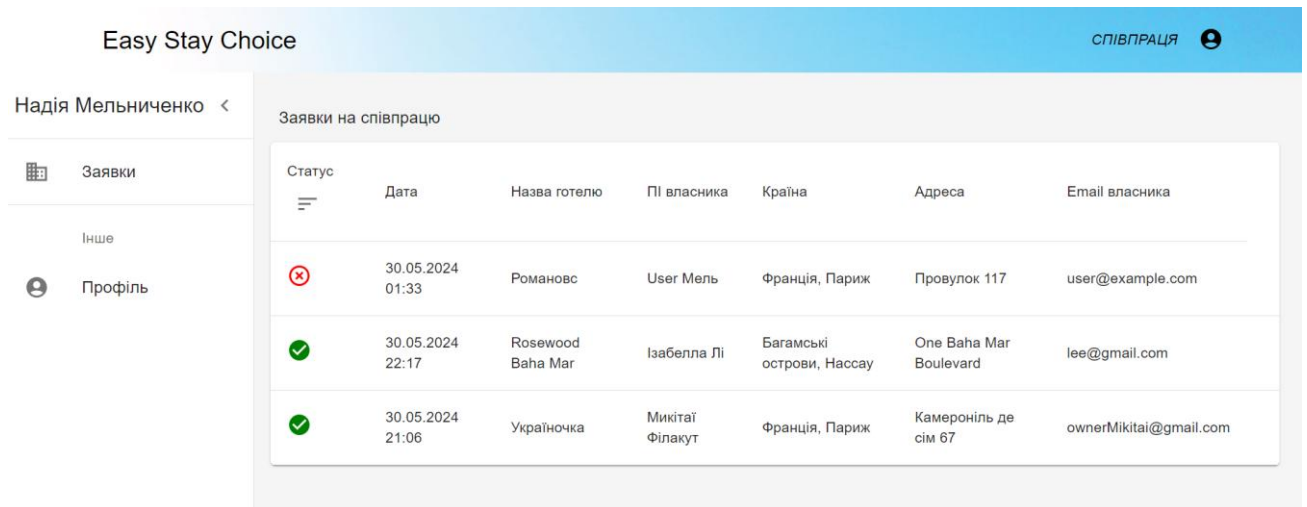


Рис. 3.12 Адмінпанель. Заявки на співпрацю

Після схвалення співпраці роль користувача змінюється на `ROLE_OWNER`. Власнику стає доступним інтерфейс управління готелями, де він може переглянути список своїх об'єктів (Рис. 3.13) та керувати інформацією про готелі та номери (Рис. 3.14).

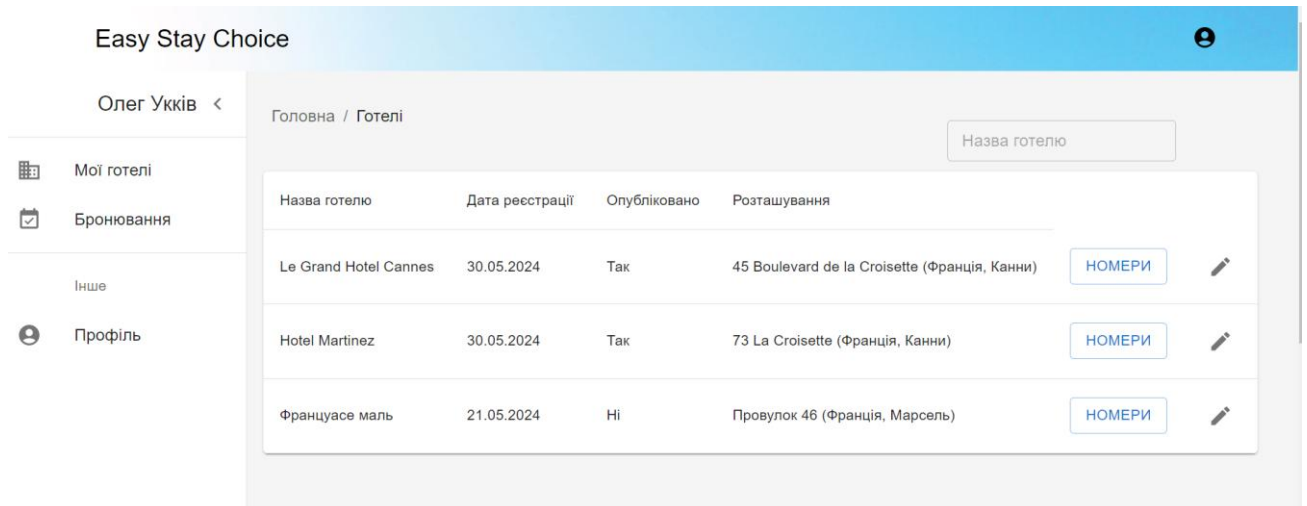


Рис.3.13 Список зареєстрованих готелів власника

Олег Укків <

Мої готелі

Бронювання

Інше

Профіль

Головна / Готелі / 27

Редагувати готель

Назва

Le Grand Hotel Cannes

Опис

Le Grand Hotel Cannes пропонує гостям елегантні номери з видом на Середземне море, приватний пляж, ресторан т

Адрес

45 Boulevard de la Croisette (Франція, Канни)

ОБРАТИ ФАЙЛ

ЗАВАНТАЖИТИ ФОТО

Дата створення

30.05.2024

ЗНЯТИ З ПУБЛІКАЦІЇ

ЗБЕРЕГТИ ЗМІНИ

Рис. 3.14 Сторінка редагування готелю

Також власник має доступ до обробки бронювань (Рис. 3.15), що впливає на історію бронювань користувачів. На етапі формування бронювання, власник може підтвердити бронювання для фіксування місця за користувачем чи відхилити запит із поясненням причини. При завершенні, користувачеві будуть нараховані бонуси.

Easy Stay Choice

Олег Укків <

Мої готелі

Бронювання

Інше

Профіль

Головна / Бронювання

Статус	Дата створення	Готель	Номер	Дата	Повна вартість	До сплати	Завершено	Дії
✓	31.05.2024 14:52	Hotel Martinez	Двомісний (Double)	11.06.2024 - 14.06.2024	300 грн	300 грн	ЗАВЕРШИТИ	
✓	31.05.2024 14:53	Hotel Martinez	Двомісний (Double)	11.06.2024 - 14.06.2024	300 грн	300 грн	ЗАВЕРШИТИ	
✗	31.05.2024 14:53	Hotel Martinez	Двомісний (Double)	05.06.2024 - 10.06.2024	500 грн	500 грн		
⚠	31.05.2024 14:53	Hotel Martinez	Двомісний (Double)	03.06.2024 - 11.06.2024	800 грн	800 грн	ЗАВЕРШИТИ	✓ ПІДТВЕРДИТИ ВІДХИЛИТИ

Рис. 3.15 Список бронювань зі сторінки власника

Сценарій роботи рекомендаційної системи

Алгоритмічна робота рекомендаційного модуля інтегрована у три основні сценарії: після авторизації на головній сторінці, під час перегляду конкретного готелю, у результатах пошуку, коли потрібно компенсувати брак релевантних пропозицій.

Відповідно сценарій роботи включає такі етапи:

1. Збір даних про взаємодії користувача (його бронювання, перегляди готелів, оцінки). Ці дані автоматично формують матрицю user-hotel;
2. Формування профілю користувача в результаті аналізу кількості попередніх взаємодій, ступеню розрідженості даних та різноманітності вибору;
3. Побудова контентної моделі, де для кожного готелю формується вектор ознак (країна, місто, рейтинг, зірковість, зручності, властивості номерів), після чого обчислюється контентна міра схожості;
4. Побудова колаборативної моделі, під час якої визначаються найбільш схожі користувачі та формується прогнознний рейтинг готелів на основі зваженої суми оцінок;
5. Обчислення адаптивного коефіцієнта α : якщо користувач має мінімум історії, α наближається до 0.8, то переважає контентна модель; якщо користувач має багату історію, α наближається до 0.2, то більша вага переходить до колаборативної частини;
6. Формування підсумкової гібридної оцінки, готелі сортуються за оцінкою і повертаються як структурований об'єкт, який фронтенд відображає у вигляді карток.

UML activity diagram для сценарію роботи рекомендаційної системи зображена в Додатку В.

3.4 Тестовий приклад роботи системи

Для підтвердження працездатності інтегрованого алгоритму рекомендаційної системи було проведено тестування у реальних умовах роботи

веб-платформи. Основною метою тестування було перевірити коректність проходження користувацького запиту, правильність формування рекомендацій модулем гібридної моделі та відповідність результатів очікуваному сценарію на клієнтському інтерфейсі.

Для початку, було обрано обліковий запис, який містить кілька завершених бронювань, що дозволяє системі використовувати як поведінкову частину, так і контентні ознаки готелів (Рис. 3.16).

The screenshot shows the 'Easy Stay Choice' application interface. At the top, there is a blue header with the text 'Easy Stay Choice' and a 'СПІВПРАЦЯ' button. Below the header, the user's name 'Іадія Мельниченко' is displayed with a back arrow. A sidebar menu on the left contains options: 'Мої бронювання' (selected), 'Інше', and 'Профіль'. The main content area is titled 'Головна / Бронювання' and contains a table of bookings.

Готель	Номер	Дата	Вартість	До сплати	Нараховано бонусів	Статус
Seaside Hotel 15		10.11.2025 - 12.11.2025	5476 грн	5770 грн	121	Підтверджено
Seaside Hotel 15		03.11.2025 - 10.11.2025	19166 грн	19662 грн	500	Підтверджено
Azure Inn 8		27.10.2025 - 29.10.2025	2154 грн	2142 грн	27	Підтверджено
Sunrise Resort 30		09.10.2025 - 15.10.2025	20430 грн	20624 грн	289	Підтверджено
Sapphire Apartments 16		22.09.2025 - 27.09.2025	4755 грн	4824 грн	113	Підтверджено
Sunrise Resort 21		04.09.2025 - 11.09.2025	15456 грн	15201 грн	415	Підтверджено
Azure Apartments 3		07.08.2025 - 12.08.2025	13530 грн	12340 грн	374	Підтверджено
Palm Villa 19		23.07.2025 - 30.07.2025	18494 грн	16920 грн	224	Підтверджено

Рис. 3.16 Історія обраного користувача

У неавторизованому стані на головній сторінці платформи відображаються базова блоки, такі як популярні країни, форма для заповнення пошти, інформація про бонуси для зареєстрованих. Для гостьового користувача формуються «Популярні готелі» для рекомендацій (Рис. 3.17):

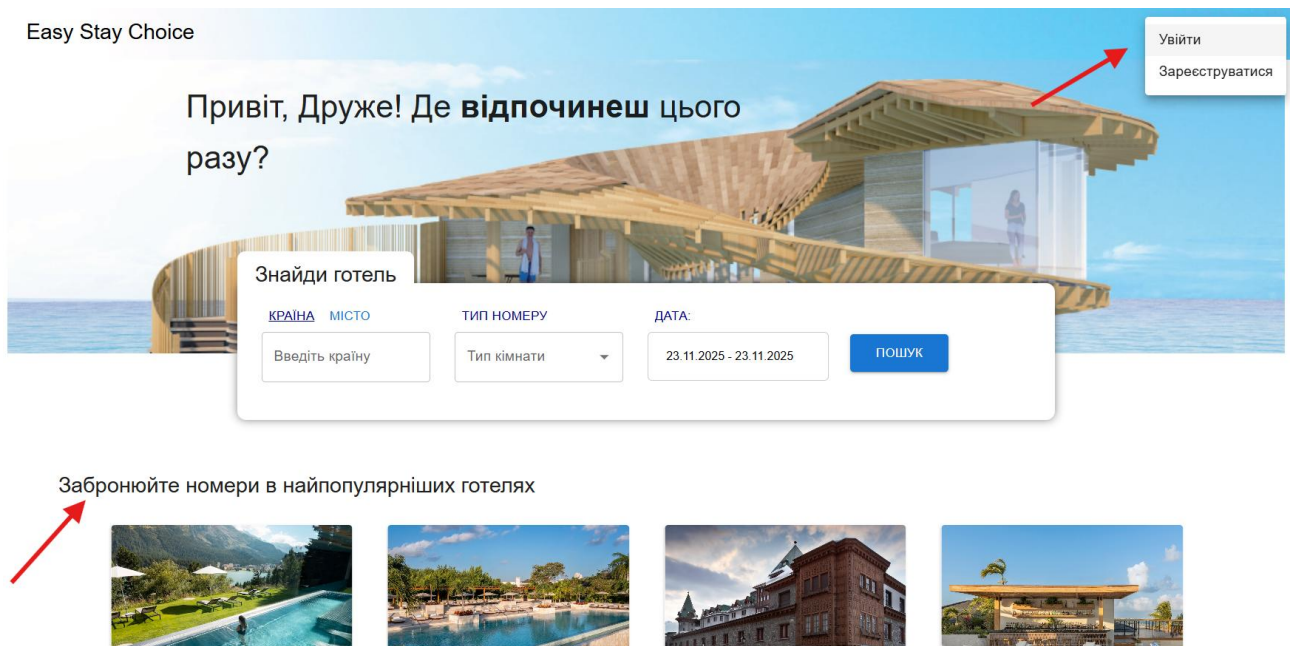


Рис. 3.17 Рекомендації для неавторизованого користувача

Після авторизації система автоматично виконує запит та формує персоналізований блок рекомендованих готелів (Рис. 3.18). Таким чином перевіряється зв'язність між JWT-аутентифікацією, модулем рекомендацій та клієнтською частиною інтерфейсу.

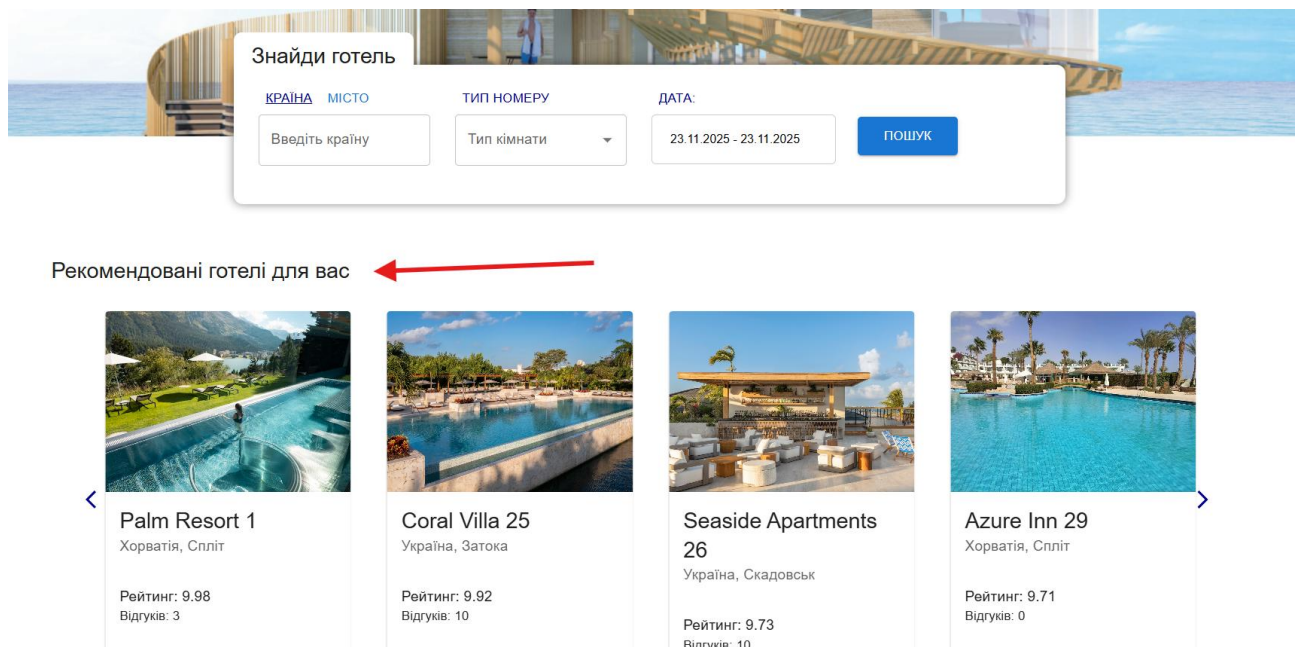


Рис. 3.18 Головна сторінка з відображенням рекомендацій

На клієнтському рівні запит виконується автоматично під час завантаження головної сторінки. Формується GET-запит `/api/recommendations` (Рис. 3.19), а у відповідь сервер повертає структуру JSON, що містить список

рекомендованих готелів. У даному тестовому прикладі сервер повернув масив із п'яти готелів із зазначенням назви, локації, рейтингу, набору зручностей, посилань на фото, інших полів готелю та значення гібридної оцінки (Рис. 3.20).

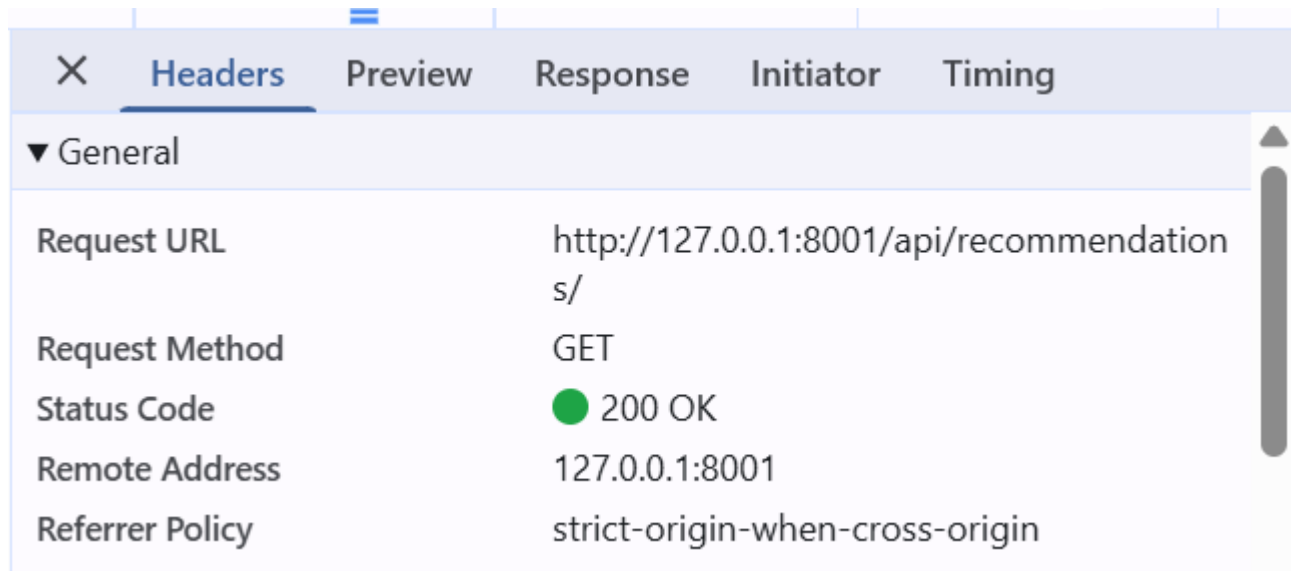


Рис. 3.19 Запит api/recommendations

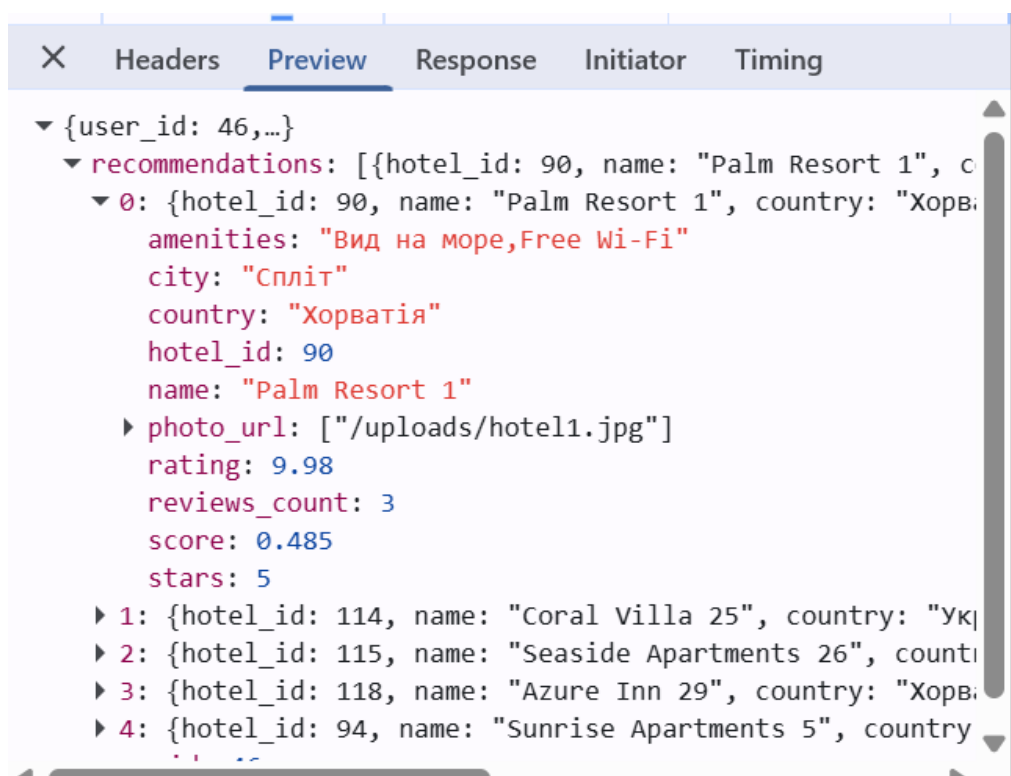


Рис. 3.20 Отримана відповідь

Відповідно отримані дані відображаються за допомогою React-компонентів у вигляді інтерактивних карток готелів. Користувач може поверхнево ознайомитись із виглядом готелю, назвою, місцем розташування,

рейтингом та кількістю відгуків. Відповідно користувач по картках може перейти до повної сторінки певного готелю (Рис. 3.21).

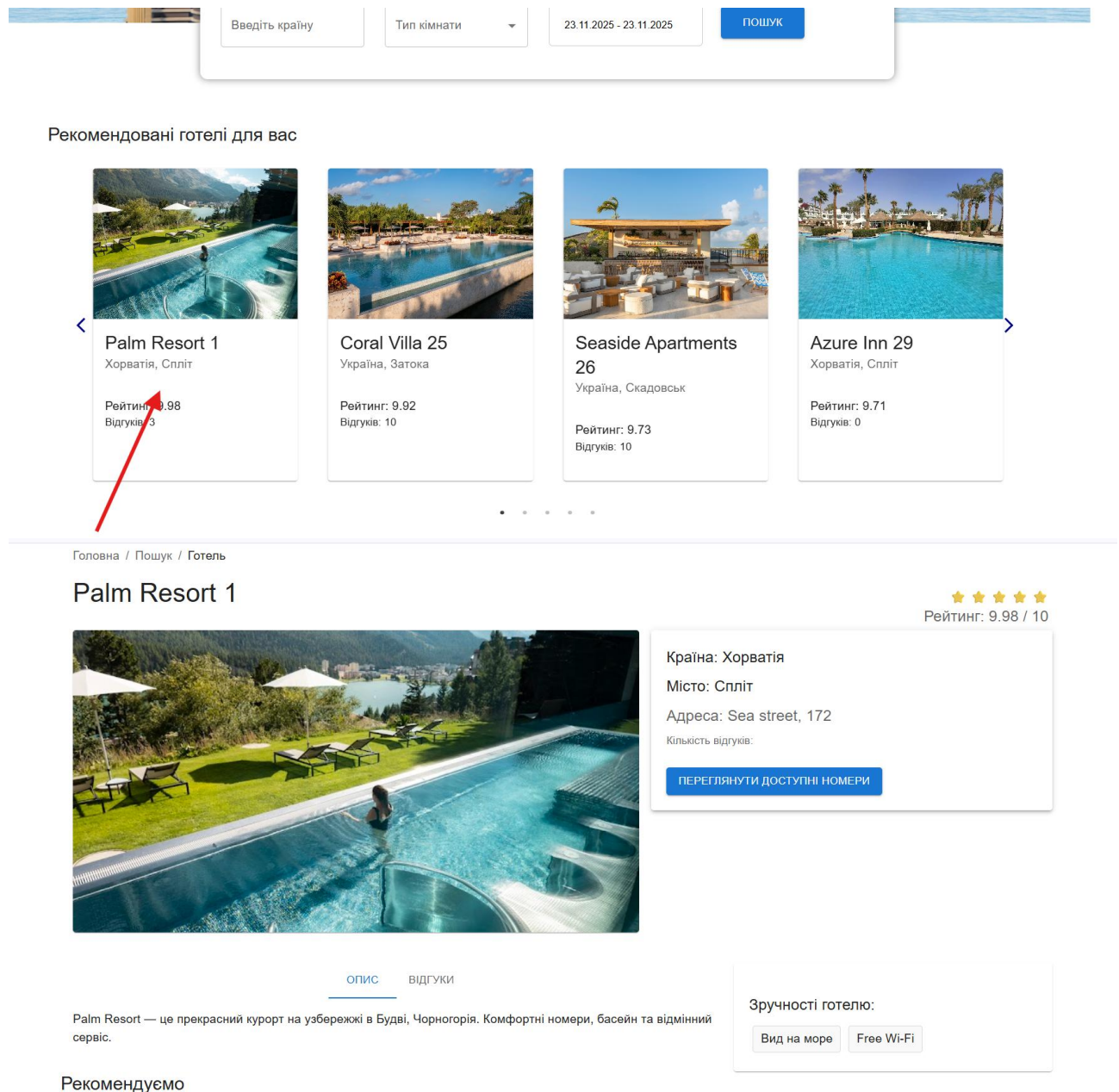


Рис. 3.21 Перехід на рекомендований готель

На самій сторінці готелю міститься більш розширена інформація, повний опис та відгуки користувачів. Додатково нижче система формує блок «Схожі готелі», який також переважно базується на контентній складовій гібридної моделі, при цьому не показуючи вже обраний готель (Рис. 3.22).

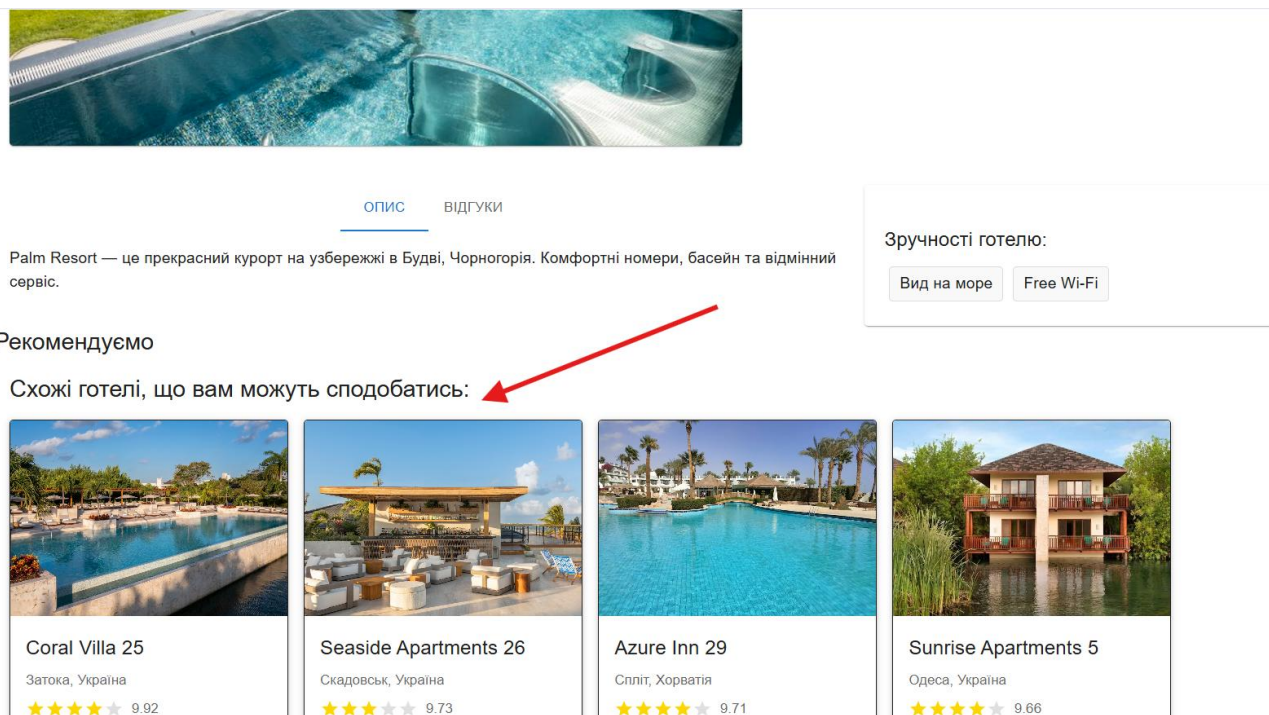


Рис. 3.22 Рекомендація схожих готелів на сторінці

Окремо було перевірено сценарій, коли користувач не має історії взаємодій. Для цього було створено новий тестовий обліковий запис без бронювань і оцінок. У цьому випадку система переходить у режим fallback, що передбачає формування списку рекомендацій на основі загальних показників популярності готелів (Рис. 3.23). Відповідно система дійсно повернула найбільш популярні готелі, що підтверджує реалізацію цього сценарію.

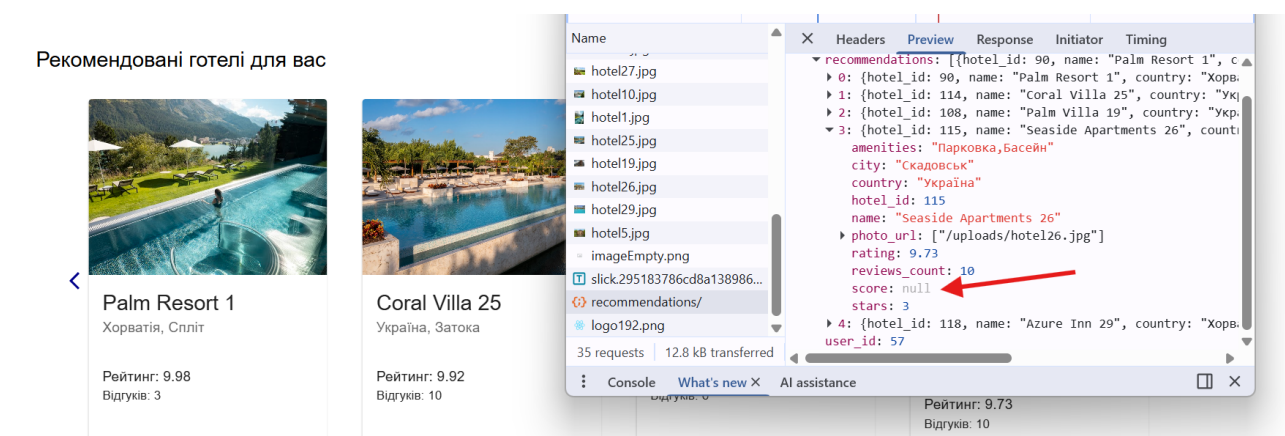


Рис. 3.23 Рекомендація популярних за відсутності історії

Для ще однієї перевірки коректної роботи рекомендаційної системи при кожному випадку, було розглянуто дії користувача без історії. Нехай він здійснив одне бронювання певного готелю, наприклад в Одесі, щоб було зафіксовано лише одну взаємодію із конкретною країною (Рис. 3.24):

різних режимах: від повної відсутності історії до багатой взаємодії. Отже, створена рекомендаційна модель показує її практичну придатність у сфері бронювання курортних готелів.

Висновки до третього розділу

У розділі 3 було здійснено комплексний опис створеного програмного прототипу веб-платформи для бронювання курортних готелів, у якому реалізовано запропоновану в роботі гібридну рекомендаційну модель. Наведено вимоги системи та умов її експлуатації, обґрунтовано вибір інструментальних засобів реалізації, описано архітектуру програмного комплексу, структуру його основних модулів та логіку їхньої взаємодії.

Проведений аналіз показав, що архітектура платформи побудована за модульним принципом, що забезпечує гнучкість, підтримуваність та масштабованість. Використання Python і FastAPI для серверної частини, SQL-бази даних для зберігання інформації та React для клієнтського інтерфейсу дозволило створити високопродуктивну веб-платформу з інтегрованими алгоритмами інтелектуального аналізу даних. Рекомендаційний модуль реалізовано як окремий сервіс, що забезпечує незалежність та чітке розділення від решти компонентів платформи, що спрощує її оновлення та розширення.

Проведений тестовий приклад підтвердив працездатність інтегрованої гібридної моделі у реальних умовах функціонування системи. Алгоритм коректно реагує на різні сценарії: від повної відсутності історії користувача до наявності значної кількості взаємодій, що свідчить про адекватність адаптивного механізму вагових коефіцієнтів. Взаємодія між серверною та клієнтською частинами відбувається коректно, результати рекомендацій відображаються своєчасно та узгоджено з інтерфейсом платформи. Отже, отримані результати підтверджують відповідність системи поставленим вимогам і перспективність її застосування у сфері бронювання курортних готелів.

ВИСНОВКИ

В процесі виконання атестаційної роботи магістра проведено комплексне дослідження методів та моделей побудови рекомендаційних систем у сфері онлайн-бронювання курортних готелів, а також створено програмний прототип веб-платформи з інтегрованою гібридною рекомендаційною моделлю.

У відповідності до поставленої мети було отримано низку наукових і практичних результатів, а саме: виконано аналіз предметної області курортних готелів, у якому визначено специфіку, ключові фактори вибору об'єктів відпочинку та роль рекомендаційних систем у сучасних онлайн-платформах.

Було проведено огляд існуючих платформ та їхніх механізмів рекомендацій, визначено сильні й слабкі сторони застосовуваних підходів. Це дозволило сформулювати функціональні вимоги до моделі власної рекомендаційної системи.

Досліджені сучасні проблеми рекомендаційних систем у туризмі показали низку типових проблем, зокрема «холодний старт», розрідженість матриць, неоднорідність даних, сезонність та необхідність балансування між контентними й поведінковими характеристиками. Відповідно було розроблено рекомендаційну модель, яка частково вирішує ці проблеми та включає три взаємопов'язані складові: контентну, колаборативну та адаптивну гібридну моделі. Запропонована адаптивна формула суттєво покращила якість рекомендацій.

Результати проведеного оцінювання точності моделі за допомогою відповідних метрик показали, що гібридна модель демонструє вищу стабільність і точність у порівнянні з окремими контентною та колаборативною складовими, особливо у випадках обмеженої історії користувачів.

Розроблено програмний прототип веб-платформи включає серверну частину (FastAPI + SQLAlchemy), клієнтський інтерфейс (React) і повністю інтегровану рекомендаційну систему. Програмний комплекс підтримує

реєстрацію, пошук, фільтрацію, бронювання, управління готелями, роботу з відгуками, а також автоматичне формування персоналізованих рекомендацій.

Реалізований тестовий приклад системи підтвердив коректність взаємодії між усіма компонентами та правильність адаптивних механізмів моделі у різних сценаріях.

У результаті виконаного дослідження було досягнуто основної мети, розроблено науково обґрунтовану та програмно реалізовану гібридну рекомендаційну модель, орієнтовану на вибір курортних готелів, інтегровану у функціонуючий прототип веб-платформи. Модель забезпечує персоналізовані, стабільні та адаптивні рекомендації, демонструючи високу практичну цінність для сфери онлайн-туризму.

Таким чином, запропоновані рішення можуть бути використані для підвищення якості взаємодії користувачів із платформами бронювання, збільшення конверсій, покращення задоволеності клієнтів та оптимізації внутрішніх бізнес-процесів компаній туристичної галузі.

Подальший розвиток роботи може включати удосконалення гібридної моделі через застосування глибинного навчання для аналізу текстових відгуків та фотографій готелів, впровадження пояснюваних рекомендацій для підвищення довіри користувачів та розширення системи на багатокритеріальні рекомендації з урахуванням бюджету та тривалості відпочинку.

ЛІТЕРАТУРА

1. Melnychenko N., Vakaliuk T. Collaborative filtering recommendation algorithms. Тези VII Всеукраїнської науково-технічної конференції «Комп'ютерні технології: інновації, проблеми, рішення», м. Житомир, 02-03 грудня 2024 р. – Житомир: Житомирська політехніка, 2024. С. 193-194.
2. Melnychenko N., Vakaliuk T. Methods of content-based filtering in recommender systems. Тези XV Міжнародної науково-технічної конференції «Інформаційно-комп'ютерні технології», м. Житомир, 28-29 березня 2025 р. – Житомир: Житомирська політехніка, 2025. С. 35-36.
3. Booking.com, Офіційний сайт [Електронний ресурс] – Режим доступу до ресурсу: <https://www.booking.com>
4. Airbnb, Офіційний сайт [Електронний ресурс] – Режим доступу до ресурсу: <https://www.airbnb.com>
5. Hyeyoung Ko. A Survey of Recommendation Systems: Recommendation Models, Techniques, and Application Fields. Electronic Solutions for Artificial Intelligence Healthcare, Volume II, 2022, <https://doi.org/10.3390/electronics11010141>
6. Yurchak, I., Hryhlevych M. Recommender system based on collaborative filtering. Computer-integrated technologies: education, science, production, No 53, 78-85, 2023, <https://doi.org/10.36910/6775-2524-0560-2023-53-12>
7. Голенко М. Ю. Застосування та види рекомендаційних систем. Інформаційно-комп'ютерні технології: стан, досягнення та перспективи розвитку, Розділ 3, С. 41-42, 2020. [Електронний ресурс] – Режим доступу до ресурсу: https://conf.ztu.edu.ua/wp-content/uploads/2021/02/tezy-dopovidej-ikt2020_os.pdf
8. Зейнеп Б. А. Системи рекомендацій: фільтрація на основі вмісту. [Електронний ресурс] – Режим доступу до ресурсу: <https://medium.com/@zbeyza/recommendation-systemscontent-based-filtering-e19e3b0a309e>

9. Hongde Zhou. A Comprehensive Survey of Recommender Systems Based on Deep Learning. Artificial Intelligence and Information Visualization in Social and Industrial Systems. 2023, <https://doi.org/10.3390/app132011378>
10. Yarasu Madhavi Latha, B. Srinivasa Rao, Amazon product recommendation system based on a modified convolutional neural network, ETRI Journal Volume 46, Issue 4 pp. 633-647, 2024, <https://doi.org/10.4218/etrij.2023-0162>
11. Deepjyoti Roy, Mala Dutta. A systematic review and research perspective on recommender systems. Journal of Big Data. 2022. <https://doi.org/10.1186/s40537-022-00592-5>
12. Stitini O., Kaloun S. Towards a robust solution to mitigate all content-based filtering drawbacks within a recommendation system. etc./Int. J. Systematic Innovation, 7(7), 89-111, 2023, [https://doi.org/10.6977/IJoSI.202309_7\(7\).0006](https://doi.org/10.6977/IJoSI.202309_7(7).0006)
13. Kidwai U. T., Akhtar D. N., Nadeem M. Unravelling Filter Bubbles in Recommender Systems: A Comprehensive Review. International Journal of Membrane Science and Technology. 10(2):1650-1680, 2023, <https://doi.org/10.15379/ijmst.v10i2.1839>
14. Bihi Sabiri, Amal Khtira. Hybrid Quality-Based Recommender Systems: A Systematic Literature Review. J. Imaging, V: 11, N: 12, 2025, [Електронний ресурс] – Режим доступу до ресурсу: <https://www.mdpi.com/2313-433X/11/1/12>
15. Годовиченко М. А., Горбатенко А. А. Рекомендаційні системи: моделі, виклики та можливості. Вісник передових інформаційних технологій, том. 6, № 4, 308–319, 2023. [Електронний ресурс] – Режим доступу до ресурсу: <https://hait.od.ua/index.php/journal/article/view/190/224>
16. Sharma, M., Mittal, R., Bharati, A., Saxena, D., Singh, A.K. Emotional Information-Based Hybrid Recommendation System. In: Thakur, M., Agnihotri, S., Rajpurohit, B.S., Pant, M., Deep, K., Nagar, A.K. (eds) Soft Computing for Problem Solving. Lecture Notes in Networks and Systems, vol 547. Springer, Singapore, 2023. https://doi.org/10.1007/978-981-19-6525-8_20

17. Fang H., Zhang D. Deep Learning for Sequential Recommendation: Algorithms, Influential Factors, and Evaluations, ACM Transactions on Information Systems (TOIS), Volume 39, Issue 1, No.: 10, Pages 1 - 42, 2020, <https://doi.org/10.26906/SUNZ.2018.4.120>
18. H. Yin, Q. Wang, K. Zheng, Z. Li and X. Zhou. Overcoming Data Sparsity in Group Recommendation. IEEE Transactions on Knowledge and Data Engineering, vol. 34, no. 7, pp. 3447-3460, 1 July 2022, <https://doi.org/10.1109/TKDE.2020.3023787>
19. Rehman Khan, H.U.; Lim, C.K.; Ahmed, M.F.; Tan, K.L.; Bin Mokhtar, M. Systematic Review of Contextual Suggestion and Recommendation Systems for Sustainable e-Tourism. Sustainability 2021, 13, 8141, <https://doi.org/10.3390/su13158141>
20. Yang Li. Recent Developments in Recommender Systems: A Survey, journal of latex class files, VOL. 14, NO. 8, AUGUST 2021, [Электронный ресурс] – Режим доступа до ресурсу: <https://arxiv.org/pdf/2306.12680>
21. Balasubramaniam N., Kauppinen M. Transparency and explainability of AI systems: From ethical guidelines to requirements. Information and Software Technology. Vol. 159, 107197, ISSN 0950-5849, 2023. <https://doi.org/10.1016/j.infsof.2023.107197>
22. Hug N. Surprise: A Python library for recommender systems. Journal of Open Source Software. T. 5, № 52. С. 2174, 2020. <https://doi.org/10.21105/joss.02174>
23. TensorFlow. An end-to-end platform for machine learning [Электронный ресурс] – Режим доступа до ресурсу: <https://www.tensorflow.org>
24. PyTorch [Электронный ресурс] – Режим доступа до ресурсу: <https://pytorch.org>
25. Scikit-learn Machine Learning in Python [Электронный ресурс] – Режим доступа до ресурсу: <https://scikit-learn.org/stable>
26. Egger, R. Text Representations and Word Embeddings. Applied Data Science in Tourism. Tourism on the Verge. Springer, Cham, 2022. https://doi.org/10.1007/978-3-030-88389-8_16

- 27.Dahouda M. K. Joe I. A Deep-Learned Embedding Technique for Categorical Features Encoding. IEEE Access, vol. 9, pp. 114381-114391, 2021, <https://doi.org/10.1109/ACCESS.2021.3104357>
- 28.StandardScaler [Электронный ресурс] – Режим доступа до ресурсу: <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html>
- 29.Kisung You, Semantics at an Angle: When Cosine Similarity Works Until It Doesn't. Department of Mathematics Baruch College, City University of New York. arXiv:2504.16318, 2025. <https://doi.org/10.48550/arXiv.2504.16318>
- 30.Ramni Harbir Singh. Movie Recommendation System using Cosine Similarity and KNN. International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-9 Issue-5, June 2020 <https://doi.org/110.35940/ijeat.E9666.069520>
- 31.Eric U. Oti, Michael O. Olusola , Francis C. Eze, and Samuel U. Enogwe. Comprehensive Review of K-Means Clustering Algorithms. International Journal of Advances in Scientific Research and Engineering (ijasre), E-ISSN: 2454-8006, Volume 7, Issue 8 August - 2021, <https://doi.org/10.31695/IJASRE.2021.34050>
- 32.FastAPI framework, high performance, easy to learn, fast to code, ready for production. [Электронный ресурс] – Режим доступа до ресурсу: <https://fastapi.tiangolo.com>
- 33.Swagger [Электронный ресурс] – Режим доступа до ресурсу: <https://swagger.io/>

ДОДАТКИ

Загальна структура backend та frontend

1	D:.	▼ frontend
2	add_data.py	▼ src
3	auth.py	▼ components
4	database.py	> book
5	main.py	> cabinet
6	models.py	> cooperation
7	users.py	> elemets
8		> hotelRooms
9	+---routers	> hotels
10	bookings.py	> hotelsEdit
11	cooperations.py	> login
12	hotels.py	> mainPageElements
13	hotel_rooms.py	▼ recommendations
14	locations.py	⚙ RecommendationContainer.jsx
15	properties.py	⚙ SimilarHotelsContainer.jsx
16	recommendations.py	> registration
17	reviews.py	> searchMainForm
18	room_types.py	> css
19	search.py	> pages
20	update_user.py	> routes
21	upload.py	> storage
22		> utils
23		
24	+---schemas	
25	hotel_room.py	
26		
27	+---services	
28	location_validation.py	
29	recommendation_service.py	

Фрагмент основної частини коду рекомендаційного сервісу:

```

class RecommendationService:
    def get_recommendations(self, user_id: int, top_n: int = 5):
        hotels, bookings = self._prepare_data()
        similarity_matrix = self._build_content_similarity(hotels)
        user_hotel_matrix, user_similarity_df =
self._build_collaborative(bookings)

        # користувач без історії
        if user_hotel_matrix is None or user_similarity_df is None or
user_id_str not in user_hotel_matrix.index:
            top = hotels[hotels["rating"] >= 4.0].copy()
            top["rating"] = pd.to_numeric(top["rating"],
errors="coerce").fillna(0.0)
            top["reviews_count"] = pd.to_numeric(top["reviews_count"],
errors="coerce").fillna(0)
            top = top.sort_values(by=["rating", "reviews_count"],
ascending=[False, False]).head(top_n)
            return [self._format_hotel_row(row) for _, row in
top.iterrows()]

        # контент-бейз
        rated_mask = user_hotel_matrix.loc[user_id_str] > 0
        rated_hotels = rated_mask[rated_mask].index

        if similarity_matrix.empty:
            content_scores = pd.Series(0.0, index=user_hotel_matrix.columns)
        else:
            rated_in_sim = [h for h in rated_hotels if h in
similarity_matrix.index]
            if len(rated_in_sim) == 0:
                content_scores = pd.Series(0.0,
index=user_hotel_matrix.columns)
            else:
                cs = similarity_matrix.loc[rated_in_sim].mean(axis=0)
                content_scores =
cs.reindex(user_hotel_matrix.columns.astype(str)).fillna(0.0)

        # колаборативна
        similar_users =
user_similarity_df.loc[user_id_str].drop(user_id_str).sort_values(ascending=
False).head(10)

        collab_scores = pd.Series(0.0,
index=user_hotel_matrix.columns.astype(str))
        for hotel in user_hotel_matrix.columns.astype(str):
            other_ratings = user_hotel_matrix[hotel].astype(float)

```

```

        common_users =
other_ratings.index.intersection(similar_users.index)
        if len(common_users) == 0:
            continue
        common_users_rating = other_ratings.loc[common_users]
        weight_com_u_rating = similar_users.loc[common_users]
        mask = common_users_rating > 0
        if mask.sum() == 0:
            continue
        weighted_sum = (common_users_rating[mask] *
weight_com_u_rating[mask]).sum()
        sum_weights = weight_com_u_rating[mask].sum()
        collab_scores.at[hotel] = (weighted_sum / sum_weights) if
sum_weights > 0 else 0.0

        # скільки вже оцінив
        num_ratings = len(rated_hotels)

        # розріджена матриця user-hotel
        uh_bool = user_hotel_matrix.astype(bool)
        total_cells = uh_bool.size
        non_zero = uh_bool.values.sum()
        sparsity = 1.0
        if total_cells > 0:
            sparsity = 1.0 - (non_zero / total_cells)

        # різноманітність профілю користувача
        diversity_score = 0.0
        if not user_hotel_matrix.empty:
            rated_in_matrix = [h for h in rated_hotels if h in
user_hotel_matrix.index]
            if len(rated_in_matrix) > 0:
                sub = user_hotel_matrix.loc[rated_in_matrix]
                diversity = float(sub.var(axis=0).mean())
                diversity_score = max(0.0, min(1.0, diversity))

        ratings_factor = min(1.0, num_ratings / 10.0)
        sparsity_factor = 1.0 - sparsity

        alpha_raw = 0.2 + 0.4 * ratings_factor + 0.2 * sparsity_factor + 0.2
* diversity_score
        alpha = max(0.2, min(0.8, alpha_raw))

        content_scores = self._normalize_scores(content_scores)
        collab_scores = self._normalize_scores(collab_scores)

        # гібридна оцінка
        hybrid_scores = alpha * content_scores + (1.0 - alpha) *
collab_scores
        hybrid_scores = hybrid_scores.drop(rated_hotels, errors="ignore")

```

```

        valid_hotels = hotels[hotels["rating"] >=
4.0]["hotel_id"].astype(str)
        hybrid_scores =
hybrid_scores[hybrid_scores.index.isin(valid_hotels)]
        top_hotels = hybrid_scores.sort_values(ascending=False).head(top_n)

# результат
hotels_map = hotels.set_index("hotel_id")
results = []
for hid, score in top_hotels.items():
    try:
        row = hotels_map.loc[int(hid)]
    except Exception:
        continue

    results.append(self._format_hotel_row(row, score))

return results

```

UML activity diagram для сценарію роботи рекомендаційної системи

