

Міністерство освіти і науки України Харківський національний університет
радіоелектроніки

Факультет _____

Комп'ютерних наук

(повна назва)

Кафедра _____

Системотехніки

(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

другий (магістерський)

(освітньо-кваліфікаційний рівень)

Дослідження та розробка рекомендаційної системи вибору кондитерських

магазинів

(тема роботи)

Виконав:

студент II курсу, групи ІТІм-24-2

Виноградов М.Ю.

(прізвище, ініціали)

Спеціальність 122 – «Комп'ютерні науки»

(код і повна назва спеціальності)

Освітня програма Інформаційні технології

проектування

(повна назва освітньої програми)

Керівник Тітов С.В.

(прізвище, ініціали)

Допускається до захисту

Зав. кафедри СТ _____

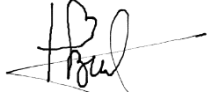
(підпис)

проф. Гребеннік І. В.

(прізвище, ініціали)

2025

Я як студент ХНУРЕ розумію і підтримую політику закладу із академічної доброчесності. Я не надавав і не одержував недозволену допомогу під час підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.



Кваліфікаційна робота не містить відомостей заборонених до відкритого опублікування.

Кваліфікаційна робота виконана у відповідності до стандартів, що діють в Україні.

Керівник кваліфікаційної роботи



доц. Тітов С. В.

(підпис)

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра системотехніки

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки

(код і повна назва)

Освітня програма Інформаційні технології проектування

ЗАТВЕРДЖУЮ:
Зав. кафедри СТ

(підпис)

" _____ " _____
2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Виноградову Микиті Юрійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та розробка рекомендаційної системи вибору кондитерських магазинів

затверджена наказом по університету від 24.11.2025 № 1058СТ

2. Термін подання студентом роботи 10 грудня 2025 р.

3. Вихідні дані до роботи (проєкту) Проаналізувати існуючі методи фільтрування в рекомендаційних системах. Обрати оптимальний метод та програмно реалізувати його використовуючи системний підхід, об'єктно орієнтований аналіз, методи структурного моделювання баз даних, об'єктно орієнтованого програмування на мові Python і фреймворку Django. Розробити рекомендаційну систему, яка дозволить кінцевому користувачу підібрати кондитерський магазин на базі його уподобань. Реалізований оптимальний метод пошуку рекомендацій з використанням баз даних та web-інтерфейсу.

4. Перелік питань, що потрібно опрацювати в роботі

4.1 Вступ. 4.2 Аналіз предметної області та існуючих підходів до проектування рекомендаційних систем. 4.3 Дослідження існуючих підходів, методів. 4.4 Постановка задачі. 4.5 Вибір оптимального методу фільтрації. 4.6 Розробка вимог до рекомендаційної системи. 4.7 Опис прийнятих проєктних рішень. 4.8 Висновки 4.9 Перелік посилань. 4.10 Додатки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) _____

5.1 Контекстна діаграма IDEF0 інформаційної системи (1 аркуш формату А4). 5.2 IDEF0-діаграма декомпозиції (3 аркуші формату А4). 5.3 Use Case діаграма. (1 аркуш формату А4). 5.4 Модель даних. (1 аркуш формату А4). 5.5 Структурна схема. (1 аркуш формату А4). 5.6 Реалізована версія інтерфейсу. (15 аркушів формату А4).

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		(підпис)	(дата)
<i>Аналіз предметної області.</i>	<i>доц. Тітов С.В.</i>		
<i>Опис прийнятих проектних рішень</i>	<i>доц. Тітов С.В.</i>		

КАЛЕНДАРНИЙ ПЛАН

Пор. №	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1.	<i>Отримання завдання кваліфікаційної роботи</i>	13.10.2025	
2.	<i>Ознайомлення з інформаційними технологіями, які використовуються для організації роботи на підприємстві</i>	14.10.2025 – 17.10.2025	
3.	<i>Вивчення предметної області</i>	17.10.2025 – 23.10.2025	
4.	<i>Підбір та вивчення джерел інформації з теми розробки</i>	23.10.2025 – 04.11.2025	
5.	<i>Огляд існуючих підходів технологій та методів у предметній області</i>	04.11.2025 – 05.11.2025	
6.	<i>Постановка задачі розробки та дослідження</i>	23.11.2025	
9.	<i>Вибір оптимального методу фільтрування рекомендованих кондитерських магазинів</i>	05.11.2025	
11.	<i>Розробка рекомендаційної системи</i>	12.11.2025 – 03.12.2025	
12.	<i>Планування обчислювальних експериментів</i>	за 5 днів	
13.	<i>Представлення на рецензування</i>	за 3 дні	
	<i>Представлення кваліфікаційної роботи в ДЕК</i>	за 2 дні	

Студент



(підпис)

Виноградов М.Ю.

Керівник роботи



(підпис)

Тітов С.В.

РЕФЕРАТ

Кваліфікаційна робота: 82 стор., 26 рис., 6 табл., 2 додатки, 37 джерел.

Графічний матеріал кваліфікаційної роботи містить 14 аркушів.

АНАЛІЗ, МЕТОДИ ПІДБОРУ РЕКОМЕНДАЦІЙ, ФІЛЬТРАЦІЯ, РЕКОМЕНДАЦІЙНА СИСТЕМА, КОНДИТЕРСЬКІ МАГАЗИНИ, БАЗА ДАНИХ, WEB-САЙТ, СЕРВЕР, ІНТЕРФЕЙС, WEB-СТОРІНКА, SQL, PYTHON, DJANGO, JAVASCRIPT.

Об'єкт дослідження – процес підбору рекомендованих кондитерських магазинів для користувача на базі його уподобань.

Предмет дослідження – методи підбору рекомендацій, які можуть бути використані в системі, що розробляється.

Мета роботи – аналіз та розробка методів підбору рекомендованих кондитерських магазинів.

Методи дослідження – аналіз існуючих методів підбору рекомендацій, системний підхід, об'єктно орієнтований аналіз, методи структурного моделювання баз даних, об'єктно орієнтованого програмування на мові Python

Результати роботи – розроблена рекомендаційна система, що дозволяє користувачу підібрати кондитерський магазин на базі його уподобань. Реалізований оптимальний метод пошуку рекомендацій з використанням баз даних та web-інтерфейсу.

Область застосування – корпоративні компанії та сервіси, які мають за мету надати кінцевим користувачам можливість швидко знайти бажаний магазин використовуючи можливість підбору рекомендацій, а також відправити заявку на створення замовлення.

ABSTRACT

Qualifying work: 82 p., 26 pic., 6 tables, 2 sources, 37 applications.

Graphic material attestation work contains 14 poster.

ANALYSIS, RECOMMENDATION SELECTION METHODS, FILTRATION, RECOMMENDATION SYSTEM, CONFECTIONERY SHOPS, DATABASE, WEBSITE, SERVER, INTERFACE, WEB-PAGE, SQL, PYTHON, DJANGO, JAVASCRIPT.

Object of the research – the process of selecting recommended confectionery stores for a user based on their preferences.

Subject of the research – the methods of generating recommendations that can be used in the system being developed.

Purpose of the work – analysis and development of methods for selecting recommended confectionery stores.

Research methods – analysis of existing recommendation selection methods, system approach, object-oriented analysis, methods of structural database modeling, and object-oriented programming in Python.

Results of the work – a recommendation system has been developed that allows the user to select a confectionery store based on their preferences. An optimal recommendation search method has been implemented using databases and a web interface.

Field of application – corporate companies and services aimed at providing end users with the ability to quickly find the desired store using recommendation selection capabilities, as well as to send a request for order creation.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧУВАНЬ І ТЕРМІНІВ	9
ВСТУП.....	10
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ ПІДХОДІВ ДО ПРОЕКТУВАННЯ РЕКОМЕНДАЦІЙНИХ СИСТЕМ	12
1.1 Аналіз предметної області	12
1.2 Визначення області застосування системи, що розробляється	14
1.3 Огляд існуючих рекомендаційних систем	15
1.4.1 Amazon	15
1.4.2 Netflix.....	16
1.4.3 Spotify	17
1.4.4 YouTube.....	18
1.4.5 LinkedIn	18
1.5 Огляд методів фільтрації	19
2 ДОСЛІДЖЕННЯ ІСНУЮЧИХ ПІДХОДІВ, МЕТОДІВ	22
2.1 Базові підходи при формуванні рекомендацій	22
2.2 Колаборативна фільтрація	23
2.2.1 Заснований на сусідстві	24
2.2.2 Заснований на пам'яті.....	25
2.2.3 Заснований на моделі	25
2.2.4 Гібридний підхід в колаборативній фільтрації.....	26
2.3 Фільтрація на основі знань	27
2.4 Фільтрація на основі популярності.....	27
2.5 Гібридна фільтрація	28
2.5.1 Гібридна фільтрація на основі зважування	29
2.5.2 Гібридна фільтрація на основі перемикачів.....	29
2.5.3 Змішана гібридна фільтрація.....	30
2.5.4 Гібридна фільтрація на основі комбінацій функцій.....	31
2.5.5 Гібридна фільтрація на основі розширення функцій	32
2.5.6 Гібридна фільтрація на основі каскаду	32
2.6 Оцінка якості рекомендаційних систем	32
2.6.1 Метрики точності.....	33

2.6.2 Якісні показники	34
2.6.3 Бізнес-орієнтовані індикатори	34
2.6.4 Комплексний підхід	35
3 ПОСТАНОВКА ЗАДАЧІ	37
4 ВИБІР ОПТИМАЛЬНОГО МЕТОДУ ФІЛЬТРАЦІЇ	38
4.1 Визначення критеріїв для формування рекомендацій кондитерських магазинів	38
4.2 Визначення оптимального методу фільтрації	39
4.3 Порівняння обраного методу фільтрації з іншими	40
5 РОЗРОБКА ВИМОГ ДО РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ	45
5.1 Розробка системних вимог до інформаційної системи	45
5.2 Визначення функціональних вимог до системи	46
5.2.1 Розробка та опис діаграми IDF0	46
5.2.2 Розробка та опис діаграми UseCase діаграми	53
6 ОПИС ПРИЙНЯТИХ ПРОЕКТНИХ РІШЕНЬ	56
6.1 Обґрунтування мови програмування для реалізації серверної частини	56
6.2 Обґрунтування інструментів для розробки клієнтської частини	58
6.3 Обґрунтування вибору СУБД	60
6.4 Моделювання даних інформаційної системи	63
6.5 Розробка гібридного методу фільтрування рекомендацій	69
6.6 Розробка інтерфейсу клієнтської частини	73
6.7 Тестування розробленого програмного забезпечення	78
ВИСНОВОК	82
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	84

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧУВАНЬ І ТЕРМІНІВ

ІС – інформаційна система; РС – рекомендаційна система; БД – база даних;

СУБД – система управління базами даних;

ERD – Entity Relationship Diagram (модель даних, яка дозволяє описувати концептуальні схеми предметної області);

HTML – HyperText Markup Language (мова гіпертекстової розмітки);

HTTP – HyperText Transfer Protocol (протокол передачі гіпертекстових файлів);

JS – Java Script (скриптова мова програмування);

MVC – Model-View-Controller (архітектурний шаблон, підхід до створення об'єктно-орієнтованих web-додатків);

SQL – Structured Query Language (мова структурованих запитів);

URL – Uniform Resource Locator (уніфіковане посилання на інформаційний ресурс);

CSS – Cascading Style Sheets (мова, що описує стиль документа HTML); яка компілюється в каскадні таблиці стилів);

JIRA – система яка допомагає відстежувати помилки та управляти проектами;

DOM – Document Object Model (модель об'єкта документа).

ВСТУП

Упродовж останніх десятиліть цифрові технології суттєво трансформували спосіб життя суспільства. Зокрема, активне впровадження програмних рішень у різні галузі – від освіти до торгівлі – спричинило появу нових форматів взаємодії між людьми та інформаційними системами. Одним із ключових чинників цієї трансформації стала доступність Інтернету, що відкрила можливість миттєвого обміну даними, виконання складних обчислень, управління бізнес-процесами та організації споживчого вибору.

Сьогодні користувач, маючи доступ до мережі через смартфон або комп'ютер, може не лише переглядати контент, а й здійснювати покупки, бронювання, замовлення послуг, а також шукати заклади поблизу – наприклад, кондитерські магазини, які відповідають його смаковим уподобанням або нагоді (святкування, подарунок, щоденне споживання). Проте велика кількість доступної інформації часто ускладнює процес вибору: користувач стикається з надлишком варіантів, серед яких важко знайти саме той, що відповідає його очікуванням.

У таких випадках на допомогу приходять рекомендаційні системи – програмні модулі, що аналізують дані про користувача (його попередні дії, вподобання, геолокацію тощо) і пропонують найбільш релевантні варіанти. Такі системи вже активно застосовуються у сфері медіа (фільми, музика), електронної комерції, новинних платформ, а також у сервісах пошуку закладів кондитерських магазинів. У контексті даної роботи розглядається застосування рекомендаційного підходу саме для вибору кондитерських магазинів, що є актуальним з огляду на зростання попиту на нішеві гастрономічні продукти.

Існує кілька основних методів побудови рекомендацій:

- колаборативна фільтрація – на основі схожості між користувачами;
- контентний підхід – враховує характеристики об'єктів (наприклад, тип продукції, стиль оформлення, ціновий сегмент);
- метод на основі знань – використовує експертні правила та логіку;

- гібридні моделі – поєднують кілька підходів для підвищення точності.

У межах магістерської роботи планується провести порівняльний аналіз зазначених методів, обрати оптимальний для задачі вибору кондитерських магазинів, та реалізувати прототип рекомендаційної системи. Основна функціональність системи полягатиме у тому, щоб на основі профілю користувача (його вподобань, історії вибору, геолокації) пропонувати релевантні заклади – наприклад, лише ті, що спеціалізуються на «французьких десертах» або «тортах на замовлення» в межах міста Харків.

Ключова перевага запропонованого рішення – персоналізований підхід, який дозволяє уникнути перегляду нерелевантних варіантів і зосередитися на тих, що мають найбільшу ймовірність зацікавити користувача. Система буде адаптована для використання як на десктопних, так і на мобільних пристроях, що забезпечить зручність доступу в будь-який момент.

З огляду на активний розвиток приватного сектору кондитерських закладів – як в Україні, так і за її межами – можна прогнозувати, що така рекомендаційна система матиме практичну цінність і буде затребуваною серед кінцевих користувачів.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ ПІДХОДІВ ДО ПРОЕКТУВАННЯ РЕКОМЕНДАЦІЙНИХ СИСТЕМ

1.1 Аналіз предметної області

Сучасна економіка та ІТ-індустрія демонструють стрімке зростання інтересу до рекомендаційних систем, що пояснюється їхнім безпосереднім впливом на прибутковість бізнесу та якість взаємодії з користувачами. За даними провідних компаній, впровадження алгоритмів персоналізованих рекомендацій дозволяє збільшити продажі на десятки відсотків, адже користувач отримує саме ті товари чи послуги, які найбільше відповідають його інтересам. У сфері електронної комерції це означає пряме зростання доходів, у медіа – збільшення часу взаємодії з платформою, а у сервісних галузях – підвищення рівня задоволеності клієнтів. Саме тому, інвестиції у створення рекомендаційних систем є економічно виправданими, оскільки вони забезпечують швидке повернення вкладених коштів та формують довгострокову лояльність користувачів. [12]

Сфера кондитерських виробів є однією з найбільш конкурентних у роздрібній торгівлі. Асортимент продукції постійно розширюється: від класичних тортів і тістечок до сучасних десертів, ексклюзивних шоколадних виробів та індивідуально оформлених замовлень. Для покупця це створює проблему вибору, адже кількість варіантів часто перевищує його можливість швидко прийняти рішення. Саме тут рекомендаційні системи стають ключовим інструментом, який допомагає не лише спростити процес вибору, але й підвищити прибутковість бізнесу.

Прибутки кондитерських магазинів значною мірою залежать від повторних покупок та лояльності клієнтів. Якщо система здатна запропонувати покупцеві новий десерт, який відповідає його смаковим уподобанням, або нагадати про улюблений торт до святкової дати, це безпосередньо стимулює

додаткові продажі. За оцінками ринку, персоналізовані рекомендації можуть збільшити середній чек на 15–30%, адже клієнт охочіше додає до замовлення товари, які йому «підказала» система. Таким чином, інвестиції у створення рекомендаційної системи для кондитерських магазинів є економічно вигідними, оскільки вони забезпечують стабільне зростання доходів та формують довгострокову лояльність клієнтів. [16]

Популярність теми пояснюється тим, що кондитерські вироби мають багато різноманітного використання. Люди купують їх не лише для себе, а й для подарунків, святкувань, корпоративних заходів. Рекомендаційна система може враховувати сезонність (наприклад, пасхальні чи новорічні вироби), індивідуальні вподобання (любов до шоколаду чи фруктових начинок), а також соціальні фактори (замовлення для сім'ї чи колективу). Це робить систему корисною не лише для окремого клієнта, а й для бізнесу, який отримує можливість точніше прогнозувати попит і планувати виробництво.

Вкладатися у створення такої системи варто тому, що вона допомагає вирішити одразу кілька проблем: зменшити час вибору для клієнта, збільшити середній чек, оптимізувати складські запаси та виробничі процеси. Для покупця це означає зручність, швидкість і задоволення від того, що його смаки враховані. Для компанії – зростання прибутків, зменшення витрат на маркетинг та формування позитивного іміджу сучасного бізнесу, який дбає про клієнта.

Позитивні сторони очевидні: персоналізація, підвищення продажів, формування лояльності, можливість аналітики поведінки клієнтів. Але водночас існують і певні недоліки. Наприклад, проблема «холодного старту» для нових клієнтів, коли система ще не має даних про їхні вподобання. Також можливі ризики надмірної персоналізації, коли клієнт отримує лише вузьке коло пропозицій і не бачить новинок. Крім того, створення та підтримка системи потребує витрат на обчислювальні ресурси та регулярне оновлення даних про асортимент.

Потенційними користувачами таких систем є як постійні клієнти кондитерських магазинів, так і нові покупці, які шукають щось особливе.

Кількість користувачів може вимірюватися тисячами для окремої мережі магазинів і мільйонами у випадку великих онлайн-платформ. Це означає, що ринок рекомендаційних систем для кондитерських виробів має значний потенціал розвитку, адже майже кожна людина хоча б кілька разів на рік купує солодощі для себе чи на подарунок.

1.2 Визначення області застосування системи, що розробляється

Розроблювана рекомендаційна система для вибору кондитерських магазинів має суттєві переваги порівняно з традиційними механізмами пошуку, які зазвичай пропонують користувачеві повний перелік доступних закладів без урахування його індивідуальних вподобань. Основна функціональність системи полягає у тому, щоб автоматично формувати персоналізований список кондитерських, які найбільше відповідають очікуванням конкретного користувача, приховуючи нерелевантні варіанти або розміщуючи їх у нижній частині видачі.

Після першого входу до веб-додатку користувач проходить коротке анкетування, у якому зазначає свої смакові уподобання, бажані категорії продукції (наприклад, торти, макарони, шоколадні набори), тип закладу (кав'ярня, пекарня, магазин), а також додаткові критерії – такі як бажаний район міста, середній ціновий рівень або наявність доставки. На основі цих даних система формує індивідуальний набір рекомендацій, який оновлюється динамічно залежно від змін у поведінці користувача або нових доступних закладів.

Користувач має можливість переглядати як рекомендовані магазини, так і повний каталог закладів, застосовуючи фільтри для уточнення запиту. Це дозволяє зберегти гнучкість у виборі та надати користувачеві контроль над процесом пошуку. Після вибору конкретного магазину, система надає можливість надіслати запит на бронювання або оформлення замовлення, що забезпечує зручну інтеграцію з бізнес-процесами закладу.

Інтерфейс додатку розроблений з урахуванням адаптивності до різних типів пристроїв – як настільних комп'ютерів, так і мобільних телефонів. Особлива увага приділена зручності навігації, логічній структурі елементів та інтуїтивному розміщенню функціональних блоків, що дозволяє користувачеві швидко орієнтуватися у системі навіть без попереднього досвіду.

Система може бути корисною не лише для індивідуальних користувачів, а й для корпоративних клієнтів – наприклад, компаній, які організують заходи, замовляють подарункові набори або шукають партнерські заклади для співпраці. Завдяки можливості швидкого підбору релевантних кондитерських магазинів та інтеграції з механізмом бронювання, система сприяє оптимізації часу пошуку, підвищенню точності вибору та покращенню комунікації між клієнтом і закладом.

1.3 Огляд існуючих рекомендаційних систем

На сьогоднішній день рекомендаційні системи стали невід'ємною частиною багатьох цифрових платформ, особливо тих, що орієнтовані на персоналізований контент або комерційні пропозиції. Їх застосування охоплює широкий спектр сфер – від електронної торгівлі до потокового відео, музичних сервісів, соціальних мереж і навіть професійних платформ. Кожна з таких систем має власну архітектуру, алгоритмічну основу та логіку формування рекомендацій, що залежить від типу даних, які вона обробляє, та цілей, які вона переслідує. [12]

1.4.1 Amazon

Одним із найвідоміших прикладів є система рекомендацій на платформі Amazon. Коли користувач переглядає товар або здійснює покупку, система фіксує ці дії, аналізує їх у контексті поведінки інших користувачів, які взаємодіяли з тими самими або схожими товарами, і формує список

рекомендацій. Наприклад, якщо користувач купує форму для випікання, йому можуть бути запропоновані інгредієнти, кулінарні книги або інші аксесуари для кондитерських виробів. Така система використовує колаборативну фільтрацію, яка базується на схожості поведінки між користувачами, а також контентну фільтрацію, що враховує характеристики самих товарів. [2]

1.4.2 Netflix

Інший приклад – Netflix, який формує рекомендації фільмів і серіалів на основі історії переглядів, оцінок, жанрових уподобань та поведінки користувачів зі схожими інтересами. Коли користувач переглядає кілька драматичних фільмів, система починає пропонувати нові драми, які мають високі рейтинги серед інших глядачів з подібними вподобаннями. Netflix активно використовує гібридні моделі, поєднуючи колаборативну фільтрацію з глибоким навчанням, що дозволяє враховувати не лише явні сигнали (оцінки, перегляди), а й неявні (тривалість перегляду, частота взаємодії). [3]

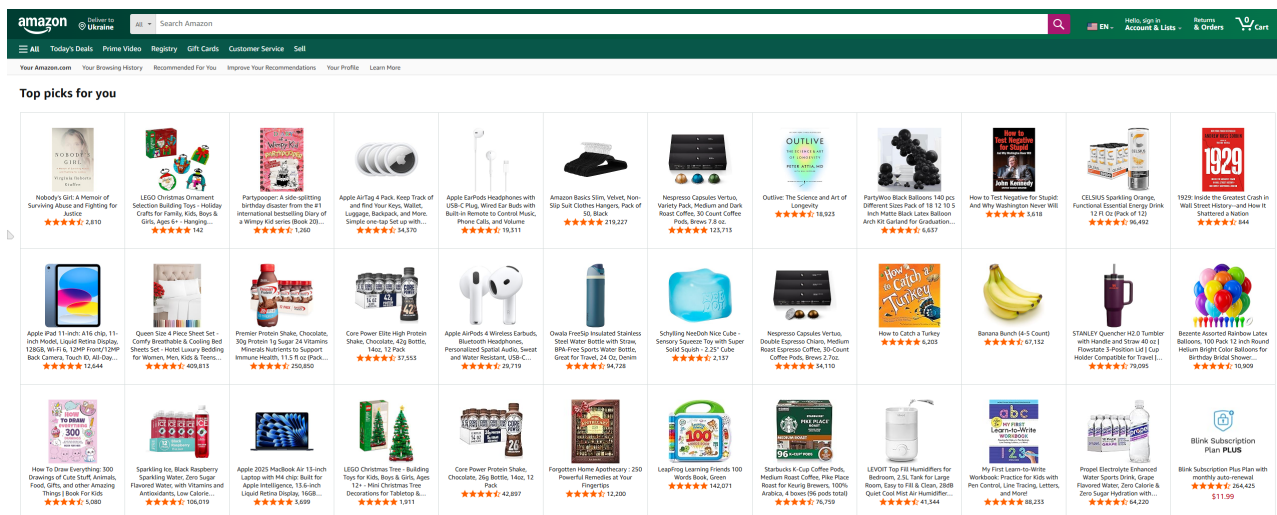


Рисунок 1.1 – Рекомендаційна система продуктів від Amazon

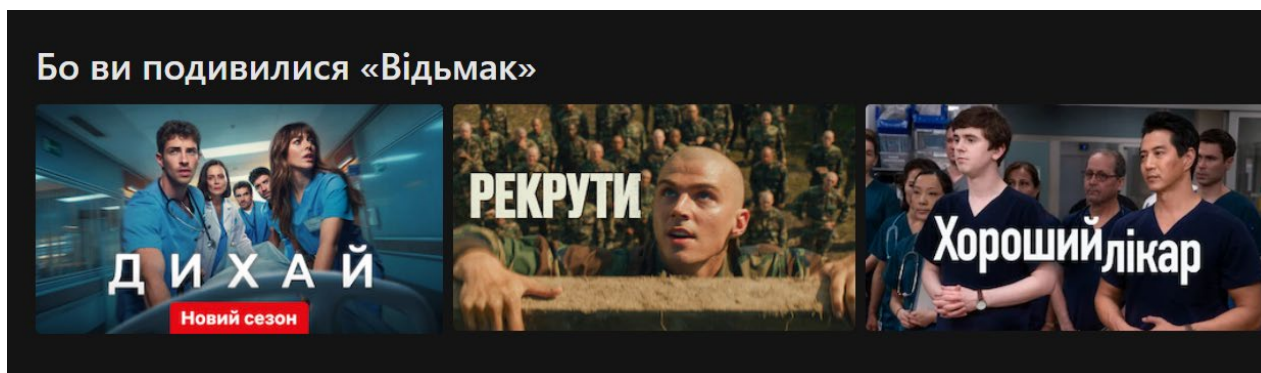


Рисунок 1.2 – Рекомендаційна система фільмів та сереалів від Netflix

1.4.3 Spotify

У сфері музики прикладом є Spotify. Коли користувач слухає певний жанр, наприклад Lo-Fi, система аналізує аудіо-фічі треків – такі як темп, тональність, структура – і пропонує схожі композиції. Крім того, враховується поведінка інших користувачів, які слухали ті самі треки, що дозволяє формувати плейлисти, які відповідають смаку конкретної аудиторії. У цьому випадку застосовується як контентна фільтрація, так і колаборативна, а також графові алгоритми для побудови зв'язків між треками. [4]

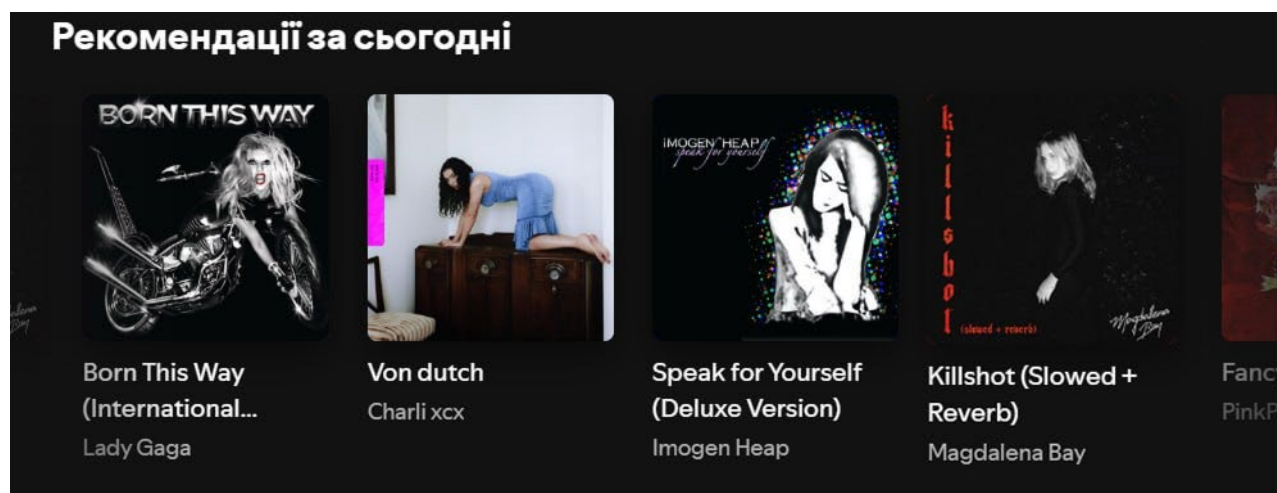


Рисунок 1.4 – Рекомендаційна система друзів від Spotify

1.4.4 YouTube

YouTube також є прикладом платформи з потужною рекомендаційною системою. Кожен перегляд, лайк, коментар або підписка фіксується і використовується для формування персоналізованої видачі на головній сторінці або в розділі «Up Next». Система враховує не лише тематику відео, а й тривалість перегляду, частоту взаємодії, а також загальну поведінку користувача. Наприклад, якщо користувач регулярно переглядає кулінарні відео, йому можуть бути запропоновані рецепти, огляди техніки для кухні або тематичні канали. YouTube використовує глибокі нейронні мережі для ранжування відео за ймовірністю перегляду, що дозволяє досягати високого рівня персоналізації. [5]

1.4.5 LinkedIn

У професійній сфері прикладом є LinkedIn, який рекомендує вакансії, контакти, курси та статті на основі профілю користувача. Система враховує професійний досвід, навички, місце роботи, а також поведінкові сигнали – наприклад, які вакансії переглядає користувач, з ким взаємодіє, які курси проходить. Тут застосовується демографічна фільтрація, контентний аналіз профілю та гібридні моделі, що дозволяють формувати точні рекомендації навіть для нових користувачів. [6]

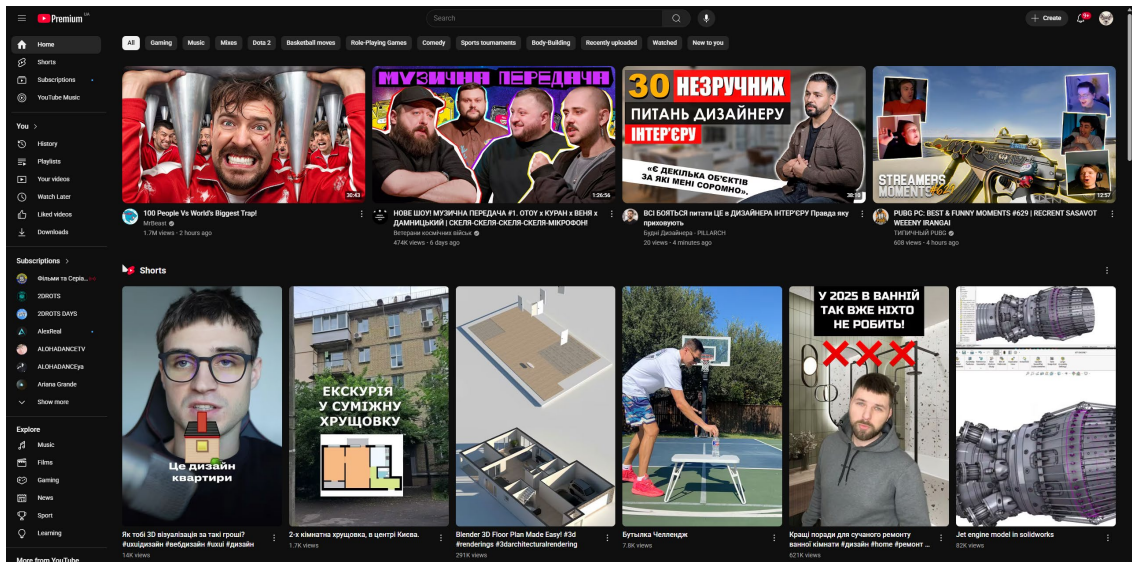


Рисунок 1.5 – Рекомендаційна система пісень від YouTube

1.5 Огляд методів фільтрації

У процесі побудови рекомендаційної системи важливу роль відіграє вибір методу фільтрації даних, що визначає логіку взаємодії між користувачами та об'єктами. Залежно від типу зв'язків (наприклад, «користувач–об'єкт» або «об'єкт–об'єкт»), застосовуються різні підходи до формування рекомендацій. У цьому розділі наведено короткий огляд основних методів фільтрації, які будуть детально розглянуті у наступних частинах роботи.

Колаборативна фільтрація

Цей метод базується на припущенні, що користувачі, які мали схожі вподобання в минулому, ймовірно, матимуть їх і в майбутньому. Система аналізує історію оцінок, виборів або взаємодій користувача з об'єктами (наприклад, кондитерськими магазинами), а також порівнює її з поведінкою інших користувачів. Якщо кілька осіб обирали однакові заклади, то новому користувачу можуть бути рекомендовані ті самі варіанти [19].

Колаборативна фільтрація може бути реалізована у двох формах:

- User-based – рекомендації формуються на основі схожості між користувачами.

- Item-based – система шукає схожі об’єкти, які оцінені однаково різними користувачами.

Контентно-орієнтована фільтрація

Цей підхід використовує властивості самих об’єктів для формування рекомендацій. Наприклад, якщо користувач обрав кондитерську, яка спеціалізується на ексклюзивних шоколадних виробках, система може запропонувати інші заклади з аналогічним асортиментом. Метод не потребує попередніх даних про інших користувачів, що робить його ефективним для нових клієнтів. [18]

Недоліки цього підходу:

- Менша точність на початкових етапах.
- Високі вимоги до якості та структури описових даних про об’єкти.

Фільтрація на основі знань

Метод базується на експертних знаннях про предметну область, логічних правилах або аналітичних моделях. Наприклад, система може враховувати, що користувач шукає заклад для святкування дитячого дня народження, і відповідно рекомендувати кондитерські, які пропонують тематичні торти, дитячі меню або послуги декору. [10]

Існують різновиди цього підходу:

- Case-based – рекомендації на основі схожих випадків.
- Demographic-based – врахування демографічних характеристик користувача.
- Utility-based – оцінка корисності об’єкта для конкретного запиту.

Гібридні методи

Гібридні системи поєднують кілька підходів для досягнення більшої точності та гнучкості. Наприклад, можна комбінувати контентну фільтрацію з колаборативною, або використовувати окремі модулі для різних типів користувачів. Такий підхід дозволяє враховувати як індивідуальні вподобання, так і загальні закономірності у виборі. [13]

Гібридизація може бути реалізована:

- Через паралельне прогнозування і об'єднання результатів.
- Через інтеграцію одного методу в інший.
- Через побудову єдиної моделі, що враховує всі типи зв'язків.

2 ДОСЛІДЖЕННЯ ІСНУЮЧИХ ПІДХОДІВ, МЕТОДІВ

2.1 Базові підходи при формуванні рекомендацій

Формування рекомендацій у сучасних інформаційних системах базується на обробці структурованих даних, які містять інформацію про користувачів, об'єкти вибору та контекст взаємодії. Кожна рекомендаційна система функціонує на основі алгоритмічного механізму, що дозволяє здійснювати фільтрацію доступних варіантів відповідно до заданих критеріїв. Такий механізм має на меті відібрати ті об'єкти, які з найбільшою ймовірністю відповідатимуть інтересам конкретного користувача.

У процесі побудови рекомендацій важливо враховувати не лише технічні характеристики об'єктів, а й поведінкові сигнали, історію вибору, географічні параметри та інші релевантні дані. Наприклад, у системі вибору кондитерських магазинів це можуть бути уподобання щодо типу продукції (торти, тістечка, шоколад), бажаний район міста, середній ціновий сегмент або наявність додаткових послуг (доставка, оформлення свят).

Вибір конкретного методу фільтрації залежить від специфіки предметної області, обсягу доступних даних та цілей, які ставить перед собою система. У деяких випадках доцільно застосовувати контентно-орієнтований підхід, коли система аналізує властивості самих об'єктів. В інших ситуаціях ефективнішим може бути колаборативний метод, що враховує схожість між користувачами. Іноді оптимальним рішенням є гібридна модель, яка поєднує кілька підходів для досягнення більшої точності.

Перед реалізацією алгоритму, що буде використовуватись у системі для підбору релевантних кондитерських магазинів, необхідно провести порівняльний аналіз існуючих методів фільтрації. Це дозволить обґрунтовано

обрати той підхід, який найкраще відповідає поставленій задачі та забезпечить високу якість рекомендацій. [12]

2.2 Колаборативна фільтрація

Колаборативна фільтрація — це різновид системи рекомендацій. Вона групує користувачів на основі схожої поведінки та пропонує нові елементи відповідно до характеристик групи.

Колаборативна фільтрація є методом пошуку інформації, який рекомендує елементи користувачам на основі того, як інші користувачі з подібними вподобаннями та поведінкою взаємодіяли з цими елементами. Іншими словами, алгоритми колаборативної фільтрації групують користувачів за їхньою поведінкою та використовують загальні характеристики групи для рекомендацій цільовому користувачеві. Колаборативні системи рекомендацій працюють за принципом, що схожі користувачі (з точки зору поведінки) мають подібні інтереси та смаки.

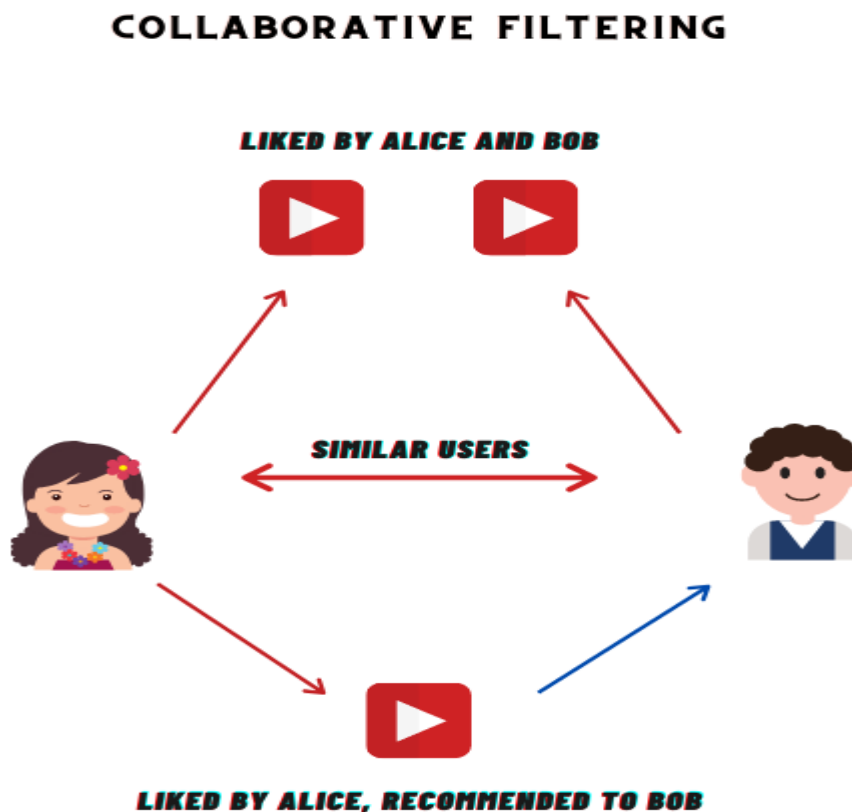


Рисунок 2.1 – Концепція колаборативної фільтрації.

Колаборативна фільтрація використовує матрицю для відображення поведінки користувачів щодо кожного елемента в системі. Система витягує значення з цієї матриці та відображає їх як точки даних у векторному просторі. Різні метрики потім вимірюють відстань між точками, щоб обчислити схожість між користувачами (user-user) та між елементами (item-item).[16]

На рисунку 2.1 зображено двох різних користувачів, їх інтереси та схожість смаків. Можемо бачити, що Alice та Bob мають схожі смаки, тому інтереси Alice рекомендується Bob та навпаки.

Саме по такому принципу і працює колаборативна фільтрація, на простому прикладі. Існує декілька підвидів даного методу фільтрації, детально розглянемо їх нижче.

2.2.1 Заснований на сусідстві

Методи, засновані на сусідстві (Neighborhood-based Collaborative Filtering), є одним із найпоширеніших підходів у рекомендаційних системах. Їхня основна ідея полягає у визначенні групи користувачів або елементів, які мають найбільшу схожість з активним користувачем чи цільовим елементом. Після цього прогноз робиться на основі оцінок цих «сусідів».

Для обчислення схожості між двома користувачами u та v часто використовується кореляція Пірсона:

$$sim(u, v) = \frac{\sum_{i \in I} (r_{u,i} - r^-_u)(r_{v,i} - r^-_v)}{\sqrt{\sum_{i \in I} (r_{u,i} - r^-_u)^2} \cdot \sqrt{\sum_{i \in I} (r_{v,i} - r^-_v)^2}}$$

де I – множина елементів, оцінених обома користувачами, $r_{u,i}$ — оцінка користувача u для елемента i , а r^-_u – середня оцінка користувача u .

Прогноз оцінки для користувача u щодо елемента j визначається як зважене середнє оцінок його сусідів:

$$\hat{r}_{u,j} = r_{-u} + \frac{\sum_{v \in N(u)} \text{sim}(u, v) \cdot (r_{v,j} - r_{-v})}{\sum_{v \in N(u)} [|\text{sim}(u, v)|]}$$

де $N(u)$ – множина найбільш схожих користувачів до u .

Приклад для кондитерських магазинів: якщо користувач А та користувач В часто відвідують одні й ті самі магазини та залишають схожі оцінки, то система може рекомендувати користувачу А новий магазин, який користувач В високо оцінив. Це дозволяє швидко знайти заклади з подібними смаками та рівнем обслуговування. [16]

2.2.2 Заснований на пам'яті

Методи, засновані на пам'яті (Memory-based Collaborative Filtering), працюють безпосередньо з матрицею «користувач-елемент», використовуючи історію взаємодій для обчислення прогнозів. Вони не потребують складних моделей, а спираються на статистичні залежності між оцінками.

Основна ідея полягає у використанні всієї матриці оцінок як «пам'яті» системи. Для кожного користувача або елемента обчислюється схожість з іншими, після чого прогноз робиться на основі агрегованих даних.

Формула прогнозу для користувача u щодо елемента j має вигляд:

$$\hat{r}_{u,j} = \frac{\sum_{v \in U} \text{sim}(u, v) \cdot r_{v,j}}{\sum_{v \in U} [|\text{sim}(u, v)|]}$$

де U – множина всіх користувачів, які оцінювали елемент j .

Приклад для кондитерських магазинів: якщо система має матрицю оцінок, де користувачі виставляють бали різним магазинам, то для нового користувача можна спрогнозувати його оцінку певного магазину, використовуючи середньозважені оцінки інших клієнтів. Наприклад, якщо більшість відвідувачів позитивно оцінили магазин «Ve-Sweet», то система може рекомендувати його новому користувачу. [19]

2.2.3 Заснований на моделі

Методи, засновані на моделі (Model-based Collaborative Filtering), будують математичну модель на основі даних про взаємодію користувачів з елементами. Найпоширеніший підхід – матрична факторизація.

Матриця оцінок R розкладається на добуток двох менших матриць:

$$R \approx P \cdot Q^T$$

де P – матриця користувачів, Q – матриця елементів, а k – кількість прихованих факторів.

Прогноз оцінки користувача u для елемента j :

$$\hat{r}_{u,j} = p_u \cdot q_j^T = \sum_{f=1}^k p_{u,f} \cdot q_{j,f}$$

Функція втрат для навчання моделі:

$$L = \sum_{(u,j) \in R} (r_{u,j} - \hat{r}_{u,j})^2 + \lambda (\|p_u\|^2 + \|q_j\|^2)$$

У системі вибору кондитерських магазинів модель може виявити приховані фактори – наприклад, «любов до шоколаду», «цінова чутливість», «сезонні покупки». Це дозволяє рекомендувати магазини навіть новим користувачам, виходячи з їхніх прихованих уподобань. [11]

2.2.4 Гібридний підхід в колаборативній фільтрації

Гібридні методи поєднують кілька підходів — наприклад, колаборативну фільтрацію та контентний аналіз. Це дозволяє підвищити точність та подолати недоліки окремих методів.

Комбінований прогноз можна записати так:

$$\hat{r}_{u,j} = \alpha \cdot \hat{r}_{u,j}^{CF} + (1 - \alpha) \cdot \hat{r}_{u,j}^{Content}$$

де α – коефіцієнт ваги, що визначає внесок кожного підходу.

Якщо користувач віддає перевагу магазинам із великим вибором шоколадних виробів, система врахує як його історію оцінок (колаборативна частина), так і характеристики магазинів (контентна частина), щоб запропонувати найбільш релевантний варіант.

2.3 Фільтрація на основі знань

Фільтрація на основі знань (Knowledge-based Filtering) ґрунтується на використанні експертних правил, логіки та знань про предметну область. На відміну від колаборативних чи контентних методів, які спираються на дані користувачів або характеристики елементів, системи на основі знань формують рекомендації на основі чітко визначених залежностей між потребами користувача та властивостями товарів.

У випадку кондитерських магазинів системи на основі знань може враховувати такі фактори:

- дієтичні обмеження (наприклад, безглютенові або низькокалорійні вироби);
- особливі події (свята, корпоративи, весілля);
- бюджет користувача;
- географічне розташування магазину.

Формально рекомендація на основі знань може описуватися, як функція відповідності:

$$Rec(u) = \{j \in Items \mid Constraints(u) \subseteq Features(j)\}$$

де $Constraints(u)$ – набір вимог користувача, а $Features(j)$ – набір характеристик магазину j .

Якщо користувач вказує, що шукає магазин із можливістю замовлення дієтичних десертів для корпоративу, система відфільтрує лише ті магазини, які мають відповідні опції. [1]

2.4 Фільтрація на основі популярності

Фільтрація на основі популярності (Popularity-based Filtering) є одним із найпростіших методів рекомендацій. Вона ґрунтується на припущенні, що

товари чи магазини, які користуються найбільшим попитом серед інших користувачів, будуть цікавими і новому клієнту.

У системі для кондитерських магазинів це може означати рекомендацію тих закладів, які мають найбільшу кількість позитивних відгуків або замовлень.

Формула для визначення популярності магазину j :

$$Pop(j) = \frac{Orders(j)}{TotalOrders}$$

де $Orders(j)$ – кількість замовлень у певному магазині j , а $TotalOrders$ – загальна кількість замовлень у системі.

Якщо магазин «Ve-Sweet» має найбільшу кількість замовлень у місті, система може рекомендувати його новим користувачам, як найпопулярніший варіант. [10]

2.5 Гібридна фільтрація

Гібридна фільтрація поєднує кілька методів рекомендацій (колаборативну, контентну, знаннєву, популярнісну) для досягнення більшої точності та подолання недоліків окремих підходів.

Основна ідея полягає у використанні різних джерел інформації: історії оцінок, характеристик магазинів, знань про користувача та статистики популярності. Це дозволяє створювати більш персоналізовані та гнучкі рекомендації. [14]

2.5.1 Гібридна фільтрація на основі зважування

У системах зважених рекомендацій важливим етапом є визначення кількох моделей, здатних адекватно інтерпретувати наявний набір даних. Суть підходу полягає у тому, що кожна модель формує власний прогноз, після чого результати комбінуються за допомогою статичних вагових коефіцієнтів. Ці ваги залишаються незмінними під час роботи системи та не коригуються на тестових даних.

Наприклад, можна інтегрувати модель контентної фільтрації та модель колаборативної фільтрації «об'єкт-об'єкт». Якщо для кожної з них встановити вагу у 50%, то остаточний прогноз сформується, як середнє значення двох результатів. Такий підхід забезпечує баланс між аналізом характеристик елементів та врахуванням поведінкових даних користувачів.

Основна перевага використання зваженого гібридного методу полягає у його простоті та лінійності. Розробник має можливість інтегрувати кілька різних моделей у єдиний механізм рекомендацій, що дозволяє більш гнучко підтримувати роботу системи з різними наборами даних. Це особливо актуально у випадках, коли окремі методи демонструють різну ефективність залежно від структури даних: комбінування результатів дає змогу компенсувати слабкі сторони кожної моделі та отримати більш стабільні рекомендації. [13]

2.5.2 Гібридна фільтрація на основі перемикачів

Метод фільтрації на основі перемикачів передбачає вибір єдиної рекомендаційної моделі залежно від конкретної ситуації. У цьому випадку система не комбінує результати кількох алгоритмів одночасно, а визначає, яка саме модель буде найбільш доречною для поточного користувача чи набору даних. Таким чином, кожна модель використовується для формування

рекомендацій на рівні окремого об'єкта, а остаточний результат залежить від попередньо встановлених умов.

Розробник системи повинен визначити чіткі критерії, за якими здійснюється вибір моделі. Це можуть бути характеристики профілю користувача (наприклад, кількість попередніх оцінок, історія взаємодії, наявність специфічних уподобань), або інші функціональні параметри, що описують контекст використання системи. Якщо даних про користувача недостатньо, система може переключитися на популярнісну модель, яка ґрунтується на загальних тенденціях. Якщо ж користувач має багату історію оцінок, доцільно застосувати колаборативну фільтрацію.

Гібридний підхід перемикає фактично вводить додатковий рівень у структуру рекомендаційної системи. Цей рівень виконує роль «менеджера моделей», який аналізує умови та визначає, яка саме модель повинна бути використана для формування рекомендацій у конкретний момент. Такий механізм дозволяє підвищити гнучкість системи та адаптувати її до різних сценаріїв роботи, зменшуючи ризик отримання нерелевантних результатів.

Для кондитерських магазинів: якщо новий користувач вперше заходить у систему і ще не має історії оцінок, то для нього застосовується популярнісна модель, яка рекомендує найпопулярніші магазини у місті. Якщо ж користувач уже кілька разів залишав оцінки та має сформований профіль, система перемикається на колаборативну фільтрацію, щоб запропонувати магазини, які відвідують люди зі схожими вподобаннями. [1]

2.5.3 Змішана гібридна фільтрація

Змішаний гібридний метод у рекомендаційних системах передбачає використання профілю користувача та його характеристик для формування кількох підмножин даних. На основі цих підмножин система генерує різні набори кандидатів, які подаються до окремих моделей рекомендацій. Кожна

модель формує власні передбачення, після чого результати об'єднуються у єдиний список рекомендацій.

Такий підхід дозволяє системі працювати з різними джерелами інформації одночасно, що забезпечує ширший спектр можливих рішень. Завдяки цьому користувач отримує не лише індивідуалізовані рекомендації, але й варіанти, які враховують загальні тенденції та особливості предметної області.

Особливістю змішаної гібридної фільтрації є здатність генерувати значну кількість рекомендацій одночасно. Система може адаптувати окремі набори даних до відповідних моделей, що підвищує продуктивність та точність результатів. Це робить метод ефективним у випадках, коли необхідно враховувати різні аспекти поведінки користувачів та властивості елементів.

Для кондитерських магазинів: якщо користувач має профіль із зазначеними уподобаннями (наприклад, любов до шоколадних виробів та середній рівень витрат), система може створити кілька наборів даних: один для колаборативної моделі, другий для контентної, третій для популярнісної. Кожна модель формує власний список магазинів-кандидатів, а змішаний гібридний підхід об'єднує ці результати, забезпечуючи користувача більш комплексними рекомендаціями. [14]

2.5.4 Гібридна фільтрація на основі комбінацій функцій

Цей підхід передбачає створення нової функції, яка інтегрує результати кількох моделей у єдиний механізм. Замість простого об'єднання або перемикання між алгоритмами, система формує спеціальну функцію, що враховує різні аспекти даних. Така функція може бути побудована на основі статистичних методів, машинного навчання або евристичних правил.

Для кондитерських магазинів: система може враховувати одночасно колаборативні оцінки користувачів, контентні характеристики магазинів (асортимент, ціни, розташування) та знаннєві правила (наприклад, сезонність

попиту). Комбінаційна функція інтегрує ці параметри, щоб сформувати більш точні рекомендації. [13]

2.5.5 Гібридна фільтрація на основі розширення функцій

Метод розширення функцій полягає у модифікації одного алгоритму шляхом додавання параметрів або результатів іншого. Це дозволяє підвищити точність базового методу, зберігаючи його структуру.

У колаборативну фільтрацію можна додати параметр «популярність магазину», що дозволить враховувати не лише схожість користувачів, а й загальні тенденції. У результаті система рекомендуватиме магазини, які одночасно відповідають індивідуальним вподобанням користувача та мають високий рівень відвідуваності. [13]

2.5.6 Гібридна фільтрація на основі каскаду

Каскадний підхід передбачає послідовне застосування кількох моделей. На першому етапі одна модель відбирає початковий набір кандидатів, а на наступних етапах інші моделі уточнюють та ранжують результати.

Для кондитерських магазинів: спочатку система може застосувати контентну фільтрацію, щоб відібрати магазини з певними характеристиками (наприклад, наявність шоколадних виробів). Потім колаборативна модель ранжує ці магазини відповідно до вподобань схожих користувачів. У фіналі популярніша модель може додатково скоригувати рейтинг, щоб врахувати загальні тенденції.

2.6 Оцінка якості рекомендаційних систем

Оцінка якості рекомендаційних систем є ключовим етапом їхнього проектування та впровадження. Вона дозволяє визначити, наскільки система

відповідає очікуванням користувачів, чи здатна вона забезпечити релевантні рекомендації та чи сприяє досягненню бізнес-цілей. У науковій та практичній літературі виділяють кілька підходів до оцінювання, які можна умовно поділити на кількісні метрики точності, якісні показники задоволеності користувачів та бізнес-орієнтовані індикатори ефективності. [15]

2.6.1 Метрики точності

Найбільш поширеним способом оцінки є використання метрик точності прогнозів. Вони показують, наскільки добре система передбачає оцінки користувачів або їхні майбутні дії. Серед основних метрик виділяють:

Mean Absolute Error (MAE):

$$MAE = \frac{1}{N} \sum_{(u,i) \in T} |r_{u,i} - \hat{r}_{u,i}|$$

де $r_{u,i}$ – реальна оцінка користувача u для елемента i , $\hat{r}_{u,i}$ – прогнозована оцінка, N – кількість тестових прикладів.

Root Mean Squared Error (RMSE):

$$RMSE = \sqrt{\frac{1}{N} \sum_{(u,i) \in T} (r_{u,i} - \hat{r}_{u,i})^2}$$

RMSE більш чутливий до великих похибок і дозволяє оцінити стабільність системи.

Precision та Recall: використовуються для оцінки якості списку рекомендацій.

$$Precision = \frac{(|Rel \cap Rec|)}{(|Rec|)}, Recall = \frac{(|Rel \cap Rec|)}{(|Rel|)}$$

де Rel – множина релевантних елементів, Rec – множина рекомендованих елементів.

У випадку кондитерських магазинів ці метрики можуть показати, наскільки система точно передбачає уподобання клієнтів (наприклад, чи дійсно рекомендований магазин відповідає їхнім смаковим інтересам).

2.6.2 Якісні показники

Окрім точності, важливим є рівень задоволеності користувачів. Для цього застосовуються опитування, інтерв'ю та аналіз поведінкових даних. Наприклад, можна вимірювати:

- середній час, витрачений користувачем на вибір магазину після отримання рекомендацій;
- кількість повторних відвідувань магазину;
- рівень довіри до системи (чи користувачі схильні слідувати рекомендаціям).

2.6.3 Бізнес-орієнтовані індикатори

Для комерційних систем важливим є вплив рекомендацій на фінансові показники. Серед ключових індикаторів:

- зростання середнього чеку (користувачі купують більше завдяки рекомендаціям);
- збільшення кількості замовлень;
- підвищення лояльності клієнтів (повторні покупки, позитивні відгуки).

У сфері кондитерських магазинів це може означати, що система не лише допомагає клієнтам швидше знайти потрібний заклад, але й стимулює їх до додаткових покупок (наприклад, вибір магазину з ексклюзивними тортами чи святковими наборами).

2.6.4 Комплексний підхід

Найбільш ефективним є комплексне оцінювання, яке поєднує кількісні та якісні показники. Наприклад, система може демонструвати високі значення Precision та Recall, але якщо користувачі не задоволені інтерфейсом або рекомендації здаються їм занадто очевидними, загальна ефективність буде низькою. Тому важливо враховувати як технічні метрики, так і людський фактор.

3 ПОСТАНОВКА ЗАДАЧІ

Завданням кваліфікаційної роботи магістра є дослідження та практичне застосування сучасних методів фільтрації, які можуть використовуватися у рекомендаційній системі для вибору кондитерських магазинів. Така система має забезпечувати користувачів персоналізованими рекомендаціями на основі їхніх індивідуальних уподобань, історії взаємодії та характеристик самих закладів.

Для досягнення поставленої мети необхідно вирішити такі підзадачі:

- здійснити аналіз найбільш поширених методів фільтрації, що застосовуються у рекомендаційних системах, зокрема колаборативної, контентної, знаннєвої та гібридної;
- визначити набір ключових критеріїв (наприклад: асортимент продукції, середній рівень цін, географічне розташування, наявність ексклюзивних виробів), які будуть враховуватися при формуванні рекомендацій для користувачів;
- обґрунтувати вибір оптимального методу або комбінації методів фільтрації, що забезпечують найбільш релевантні рекомендації кінцевому користувачу на основі його профілю та смакових уподобань;
- створити інтерфейс взаємодії у вигляді веб-сторінок та форм, що забезпечуватимуть зручність роботи як для кінцевого користувача, так і для адміністратора системи;
- реалізувати функціонал рекомендаційної системи, який надаватиме користувачу перелік магазинів, що найбільш відповідають його інтересам, а також забезпечить можливість відправки заявки на бронювання або попереднє замовлення продукції у вибраному закладі.

4 ВИБІР ОПТИМАЛЬНОГО МЕТОДУ ФІЛЬТРАЦІЇ

4.1 Визначення критеріїв для формування рекомендацій кондитерських магазинів

У процесі дослідження предметної області визначено, що найбільш релевантними критеріями для відбору кондитерських магазинів, які пропонуються кінцевому користувачу з урахуванням його індивідуальних смаків та потреб, є такі:

- Географічне розташування користувача. Система обмежує вибір лише тими магазинами, що знаходяться у потрібному регіоні, неподалік від геолокації користувача. Наприклад, мешканцю Харкова будуть запропоновані лише кондитерські заклади, розташовані в межах міста.

- Уподобання щодо асортименту. Критерій враховує пріоритетні категорії продукції, які користувач обирає найчастіше (торти, тістечка, еклери, шоколадні вироби тощо).

- Тип закладу. Додатково можна враховувати формат кондитерської: невелика локальна пекарня, спеціалізований магазин десертів, мережевий заклад чи преміум-кондитерська.

- Ціновий сегмент. Дає змогу фільтрувати заклади за рівнем вартості продукції (доступні, середньої ціни, дорогі, преміальні). Наприклад, якщо користувач орієнтується на ексклюзивні десерти високої вартості, система не буде пропонувати бюджетні варіанти.

Для впорядкування списку рекомендованих кондитерських магазинів можуть застосовуватися додаткові показники:

- Популярність закладу за кількістю переглядів або відгуків. Відображає рівень інтересу користувачів системи до конкретного магазину. Чим більше переглядів отримав заклад, тим вище його позиція у списку рекомендацій.

- Кількість замовлень або бронювань. Враховує кількість підтверджених замовлень від поточного користувача. Це показник індивідуальної

зацікавленості: чим він вищий, тим більш пріоритетною буде позиція магазину у рейтингу.

4.2 Визначення оптимального методу фільтрації

У процесі розробки рекомендаційної системи для вибору кондитерських магазинів для правильної реалізації системи вагому роль грає – визначення найбільш ефективного методу фільтрації, який дозволить забезпечити релевантність, точність та персоналізацію результатів. Для цього проведено аналіз всіх варіантів фільтрації у розділі 2, що застосовуються в сучасних рекомендаційних системах, зокрема контентного, колаборативного та гібридного методів.

Контентний метод фільтрації базується на порівнянні характеристик об'єктів з уподобаннями користувача. Він є ефективним у тих випадках, коли система має доступ до чітко структурованих даних про об'єкти (у випадку кондитерських магазинів) та анкетних даних користувача. Колаборативна фільтрація, навпаки, орієнтується на поведінкові патерни інших користувачів, що дозволяє виявляти приховані зв'язки між об'єктами, але потребує великої кількості історичних даних і має проблему «холодного старту».

Зважаючи на специфіку предметної області – доцільним є застосування методу, який дозволяє враховувати індивідуальні вподобання користувача, такі як:

- бажаний регіон розташування магазину;
- тип продукції;
- формат закладу;
- цінова категорія.

Контентний метод фільтрації дозволяє сформувати вибірку об'єктів, які повністю відповідають зазначеним критеріям. Такий підхід забезпечує високу точність рекомендацій, оскільки система виключає об'єкти, що не відповідають хоча б одному з параметрів. Крім того, цей метод не потребує попередньої

історії взаємодії користувача з системою, що дозволяє уникнути проблеми «холодного старту» – достатньо лише заповнити анкету при реєстрації.

Однак, незважаючи на переваги, контентний метод має суттєве обмеження – він не враховує популярність об'єктів. Це означає, що магазини з високим рівнем відвідуваності, позитивними відгуками або частими бронюваннями можуть опинитися в кінці списку рекомендацій, якщо не відповідають усім критеріям. Такий підхід не завжди є доцільним, оскільки популярність може бути важливим індикатором якості та довіри до закладу.

Для усунення цього недоліку запропоновано використання гібридного методу фільтрації, який поєднує переваги контентного підходу з додатковими механізмами зважування та сортування. Гібридна модель передбачає три послідовні етапи:

- Контентне фільтрування – відбір об'єктів, що відповідають заданим критеріям користувача.
- Зважування – оцінка об'єктів за додатковими параметрами, такими як кількість переглядів, рейтинг, кількість бронювань.
- Сортування за популярністю – ранжування об'єктів у списку рекомендацій за показниками загальної та суб'єктивної популярності.

Такий підхід дозволяє не лише забезпечити відповідність уподобанням користувача, але й врахувати соціальні сигнали, що підвищує довіру до системи та покращує користувацький досвід.

4.3 Порівняння обраного методу фільтрації з іншими

Для обґрунтування вибору гібридного методу фільтрації здійснено порівняння його характеристик із контентним та колаборативним підходами. Аналіз проводився за низкою критеріїв, що є релевантними для рекомендаційної системи кондитерських магазинів. Результат порівняння наведено в таблиці 4.1.

Таблиця 4.1 – Порівняння методів фільтрацій

Критерій оцінки	Контентний метод	Колаборативний метод	Гібридний метод
Врахування вподобань користувача (тип продукції, формат, регіон, ціна)	+	-	✓
Врахування популярності закладів (перегляди, бронювання, рейтинг)	-	-	+
Адаптація до змін у поведінці користувачів	-	+	+
Проблема «холодного старту»	-	+	-
Масштабованість системи	+	-	+
Потреба в анкетуванні користувача	+	+	+
Точність рекомендацій	Висока	Низька	Висока
Залежність від історії взаємодії	Немає	Висока	Помірна
Складність реалізації	Низька	Середня	Висока
Гнучкість у налаштуванні	Середня	Висока	Висока

Базуючись на даних з таблиці 4.1 можемо зробити висновок, що гібридний метод фільтрації є найбільш збалансованим і ефективним для побудови рекомендаційної системи у сфері кондитерських магазинів. Він поєднує точність контентного підходу з адаптивністю колаборативного, доповнюючи їх механізмами сортування за популярністю. Це дозволяє формувати персоналізовані, релевантні та соціально обґрунтовані рекомендації, що значно покращує користувацький досвід.

Єдиним недоліком є необхідність заповнення анкети користувача, що може потребувати додаткового часу, однак є критично важливим для забезпечення персоналізації.

Для того, щоб більш наочно зобразити принцип роботи обраного методу фільтрації, розглянемо його на приклади в таблицях 4.2 - 4.4.

Таблиця 4.2 – Кондитерські магазини

Назва закладу	Місто	Тип продукції	Формат	Цінова категорія	Перегляди	Бронювання
Lviv Handmade Chocolate	Харків	Шоколад, тістечка	Кав'ярня	Дорого	320	12
«Ведмедик»	Харків	Торти, печиво	Магазин	Прийнятно	180	8
La Crème	Харків	Французькі десерти	Кафе	Дорого	250	15
Кондитерська «Вероніка»	Львів	Торти, тістечка	Кафе	Прийнятно	400	20
SHOCO	Львів	Еклер, макарони	Кав'ярня	Дорого	370	18
Кондитерська «Цукерня»	Львів	Українські солодощі	Магазин	Дешево	210	10
Волконський	Київ	Торти, круасани	Кафе	Дорого	500	25
Пекарня «Пирогоманія»	Київ	Пироги, печиво	Магазин	Прийнятно	280	14
SHU Coffee Bar	Київ	Десерти, кава	Кав'ярня	Прийнятно	350	16
Buon Fratelli	Дніпро	Італійські десерти	Ресторан	Дорого	430	19
CheBakery	Харків	Торти, десерти	Кав'ярня	Прийнятно	250	15
Pakufuda Bakery	Харків	Торти, випічка	Кав'ярня	Прийнятно	140	10

Для наглядного прикладу роботи гібридного методу проведено експериментальне тестування на базі умовної вибірки кондитерських магазинів. змодельована ситуацію, в якій користувач має наступні вподобання:

- регіон: Харків;
- тип продукції: торти;
- формат закладу: магазин / кав'ярня;
- цінова категорія: прийнятно.

Виходячи з вподобань, система з методом гібридної фільтрації видала таку таблицю рекомендацій, які наведено в таблиці 4.3.

Таблиця 4.3 – Рекомендації кондитерських магазинів користувачу за допомогою гібридної фільтрації.

Назва закладу	Місто	Тип продукції	Формат	Цінова категорія	Перегляди	Бронювання
CheBakery	Харків	Торти, десерти	Кав'ярня	Прийнятно	250	15
«Ведмедик»	Харків	Торти, печиво	Магазин	Прийнятно	180	8
Rakufuda Bakery	Харків	Торти, випічка	Кав'ярня	Прийнятно	140	10

Наступним кроком робимо те саме, але замінюємо метод фільтрації з гібридного на контентний метод і отримуємо результат, який наведено в таблиці 4.4. Бачимо, що в цьому варіанті в нас немає сортування за популярністю.

Таблиця 4.4 – Рекомендації кондитерських магазинів користувачу за допомогою контентного методу.

Назва закладу	Місто	Тип продукції	Формат	Цінова категорія	Перегляди	Бронювання
CheBakery	Харків	Торти, десерти	Кав'ярня	Прийнятно	250	15
Rakufuda Bakery	Харків	Торти, випічка	Кав'ярня	Прийнятно	140	10
«Ведмедик»	Харків	Торти, печиво	Магазин	Прийнятно	180	8

Аналізуючи результати вибірок, наведених у таблицях 4.3 та 4.4, можна помітити, що обидва методи формують схожі списки закладів. Це пояснюється тим, що для відбору використовуються однакові критерії користувача: місто, тип продукції, формат закладу та цінова категорія. Таким чином, базовий набір об'єктів у вибірці є ідентичним. Проте ключова відмінність полягає у порядку відображення закладів у списку рекомендацій:

- Метод заснований на вмісті формує вибірку виключно за відповідністю критеріям користувача. Якщо заклад відповідає всім умовам: знаходиться у Харкові, пропонує торти, має формат кав'ярні або магазину та прийнятну цінову політику, він включається до списку. При цьому порядок розташування

об'єктів у списку не має додаткової логіки — заклади можуть відображатися довільно або за технічним порядком внесення до бази даних.

- Гібридний метод доповнює контентне фільтрування механізмом сортування за популярністю. Це означає, що після формування базової вибірки об'єкти ранжуються за спеціальним індексом популярності. Індекс розраховується на основі таких параметрів:

- кількість переглядів закладу у системі;
- кількість бронювань або покупок.

Таким чином, у гібридному методі перші позиції займають найбільш популярні кондитерські магазини, що підвищує довіру до рекомендацій та зручність для користувача.

5 РОЗРОБКА ВИМОГ ДО РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ

5.1 Розробка системних вимог до інформаційної системи

Рекомендаційна система для вибору кондитерських магазинів призначена для автоматизації процесу формування списку рекомендованих кондитерських, які з високою ймовірністю можуть зацікавити кінцевого користувача. У процесі підбору враховуються індивідуальні уподобання користувача, що дозволяє забезпечити персоналізований вибір кондитерських та підвищити ефективність взаємодії з системою.

Аналізуючи завдання створення такої системи, можна визначити основні системні вимоги до її реалізації:

- Серверна частина повинна розроблятися мовою програмування Python, що забезпечить гнучкість, масштабованість та можливість інтеграції алгоритмів фільтрації й ранжування.

- Клієнтська частина має розроблятися у вигляді веб-додатку, адаптованого для сучасних браузерів (Google Chrome, Opera, Firefox та інші). Особливу увагу слід приділити мобільній версії додатку, яка забезпечить зручність використання зі смартфонів.

- Кодова база клієнтської частини повинна реалізовуватися за допомогою React.js, що забезпечить ефективну взаємодію з Python-бекендом через REST API. Для стилізації інтерфейсу необхідно застосувати SCSS, що дозволить підтримувати структурованість та гнучкість у розробці CSS.

- У системі має бути реалізована можливість реєстрації та аутентифікації користувачів, зокрема через інтеграцію з Google OAuth API, що гарантує безпечний доступ до сервісу та захист персональних даних.

- У випадку виникнення помилок на стороні веб-сервера, серверна частина повинна автоматично перенаправляти користувача на спеціальну сторінку з детальною інформацією про проблему та можливими шляхами її вирішення.

5.2 Визначення функціональних вимог до системи

5.2.1 Розробка та опис діаграми IDF0

Розробка рекомендаційної системи для вибору кондитерських магазинів в себе декілька процесів, пов'язаних між собою однією системою. «Формування рекомендацій для вибору кондитерських магазинів» – глобальна функція в цій системі. У процесі формування рекомендацій беруть участь три суб'єкти: користувач, адміністратор та інформаційна система. Користувач є джерелом уподобань, адміністратор – джерелом внутрішніх правил сервісу, а інформаційна система – технічним механізмом реалізації логіки рекомендацій. Функція приймає в себе певні дані: інформація про користувача, база даних кондитерських магазинів, які потрібні для коректного відпрацювання ІС. Інформація про користувача включає загальні дані, а також індивідуальні уподобання, які визначають критерії відбору закладів. База даних магазинів містить структуровану інформацію про заклади, їхні характеристики, типи продукції, формат, регіон розташування, цінову політику та показники популярності. Результатом роботи системи є сформований список рекомендованих кондитерських магазинів, який передається користувачу. На рисунку 5.1 зображена концептуальна діаграма.

На першому рівні декомпозиції функція «Формування рекомендацій» розбивається на чотири послідовні процеси: авторизація користувача, збір і нормалізація даних, фільтрація закладів, а також фіналізація рекомендацій. Кожен із цих процесів має власні вхідні та вихідні дані, керуючі впливи та механізми реалізації, що дозволяє детально описати логіку роботи системи. На рисунку 5.2 зображений перший рівень декомпозиції концептуальної діаграми.

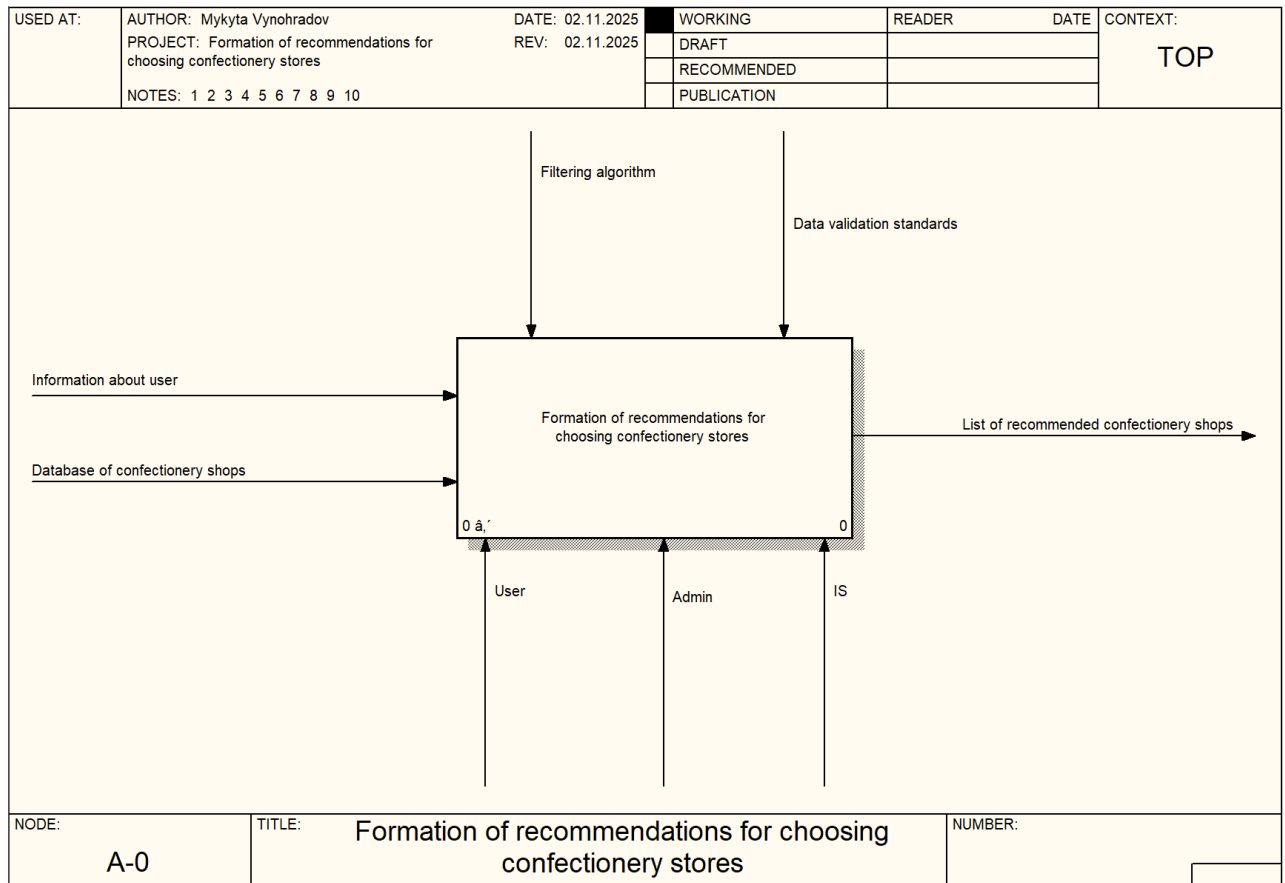


Рисунок 5.1 – IDEF0 Концептуальна діаграма

Перший процес – «Авторизація користувача» – передбачає перевірку автентичності особи, отримання базової інформації про користувача, заповнення анкети вподобань та створення профілю, який буде використано на наступних етапах. Вхідними даними є інформація про користувача, а вихідними – сформований профіль, який передається до блоку збору даних.

Другий процес – «Збір і нормалізація даних» – включає три підпроцеси: отримання профілю користувача, завантаження бази даних кондитерських магазинів та нормалізацію інформації. На цьому етапі система об'єднує дані з різних джерел, перевіряє їх на відповідність стандартам, усуває дублікати та формує єдину структуру, придатну для подальшої обробки. У процесі беруть участь адміністратор та інформаційна система, які забезпечують доступ до бази даних та контроль за якістю інформації. Вихідним результатом є валідні дані, що передаються до блоку фільтрації. Другий процес має декомпозицію на підпроцеси, які зображені на рисунку 5.3.

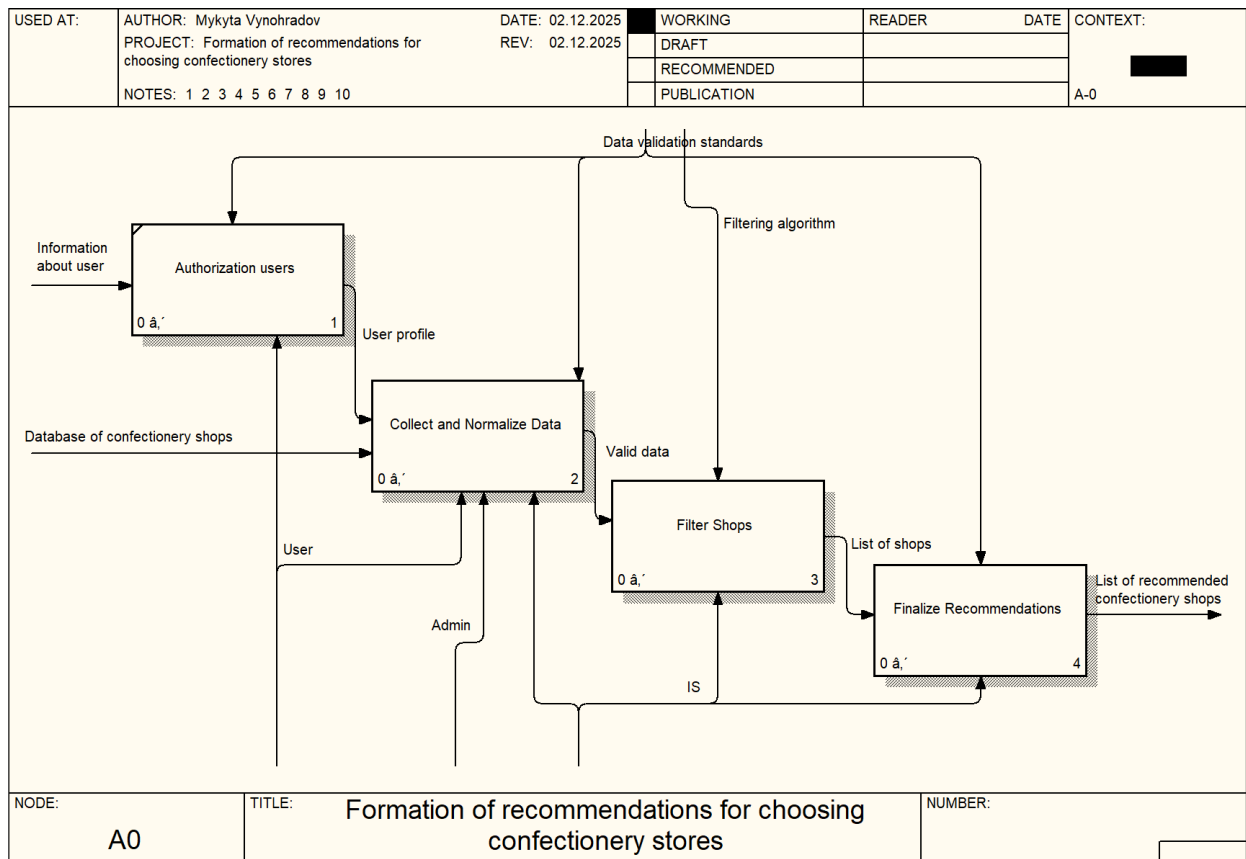


Рисунок 5.2 – IDEF0 Декомпозиція першого рівня

Процес «Збір і нормалізація даних» відповідає за отримання профілю користувача, завантаження даних про кондитерські магазини та приведення цієї інформації до валідного, узгодженого формату. На підпроцесі «Отримати профіль користувача» система приймає дані авторизованого користувача й формує сирий профіль, що містить регіон, уподобання та історію взаємодій. Вхідним потоком є активна користувацька сесія, керування здійснюється правилами авторизації, а механізмом є модуль автентифікації та база профілів. Результатом стає структурований профіль, який передається далі.

На «Завантаження бази даних магазинів» система отримує з бази даних закладів сирі записи про магазини (назва, місто, типи продукції, формат, ціни, показники активності) й готує їх до перевірки. Тут вхідними даними слугує сама база кондитерських магазинів, вихід – узгоджені сирі дані про заклади для наступного етапу.

На «Валідація та нормалізація даних» відбувається валідація й нормалізація як користувацьких, так і закладних даних відповідно до затверджених стандартів якості. У цьому підпроцесі вхідними є сирий профіль користувача і сирі дані закладів, керування забезпечується правилами валідації. Вихідним результатом є валідні дані – єдине узгоджене подання профілю та об'єктів, готове до фільтрації в процесі «Фільтрація закладів».

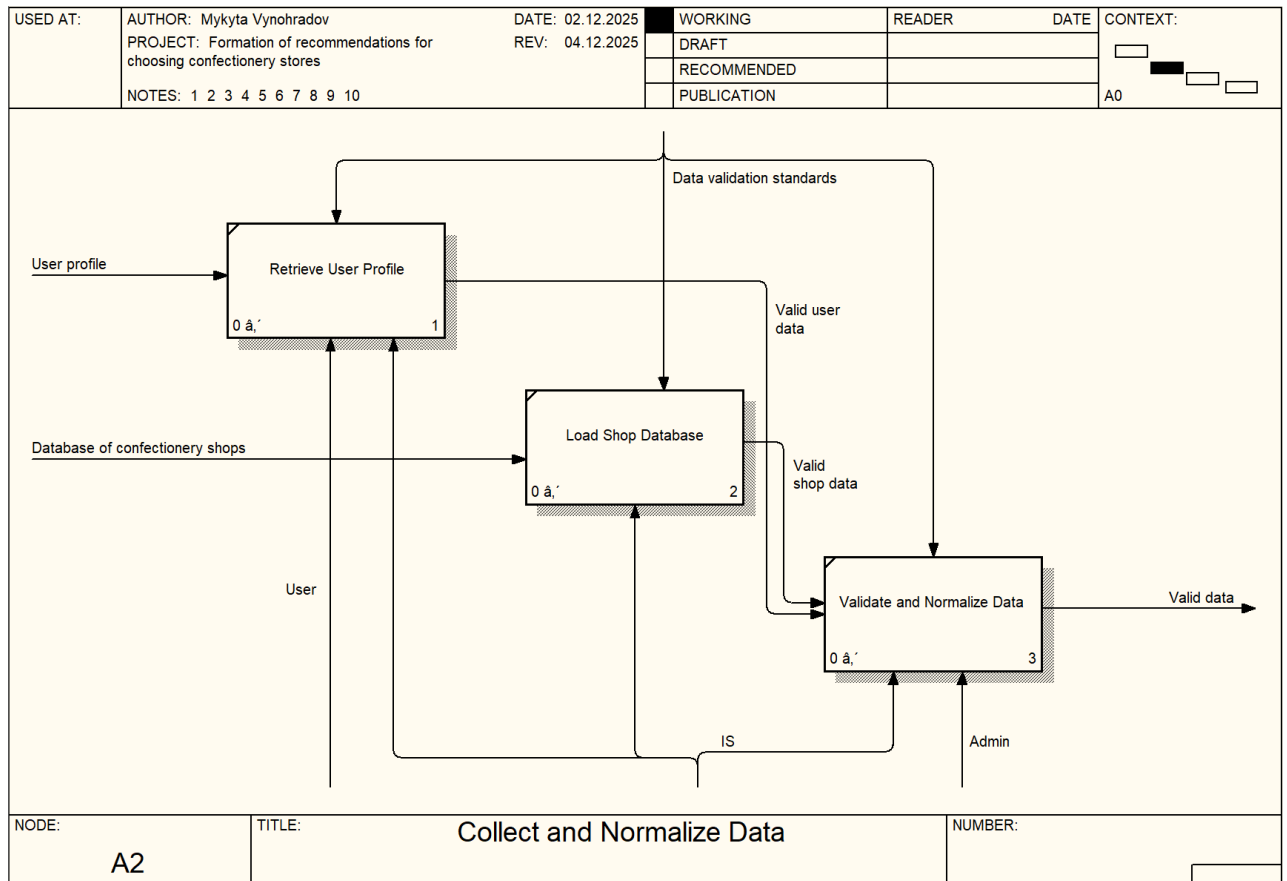


Рисунок 5.3 – IDEF0 Діаграма декомпозиції блоку «Збір і нормалізація даних»

Третій процес – «Фільтрація закладів» – реалізується у три етапи. Спочатку застосовується контентна фільтрація, яка відбирає заклади відповідно до уподобань користувача: регіон, тип продукції, формат закладу та цінова категорія. Далі застосовується фільтрація на основі зворотного зв'язку, яка дозволяє врахувати внутрішні правила сервісу, виключити неактивні або небажані заклади. На завершальному етапі здійснюється сортування за популярністю, що базується на кількості переглядів, бронювань та інших показниках активності. В результаті формується список закладів,

упорядкований за релевантністю та соціальною значущістю. Третій процес має декомпозицію на підпроцеси, які зображені на рисунку 5.4.

Процес «Фільтрація закладів» реалізує трирівневу фільтрацію: контентну, на основі службових правил та сортування за популярністю. На етапі «Застосувати фільтрацію на основі вмісту» система застосовує критерії користувача – регіон, тип продукції, формат закладу, цінову категорію – до валідних даних, сформованих у «Збір і нормалізація даних». Вхідним потоком є валідні дані (узгоджений профіль і об'єкти, уподобання користувача) керування забезпечується алгоритмом контентної фільтрації. Вихід цього етапу – фільтрований список закладів, що відповідають критеріям користувача.

На «Застосування фільтрації на основі правил» виконується виключення об'єктів, що порушують політику сервісу: неактивні, некоректні або позначені модерацією записи. Тут вхідними даними є список, отриманий на попередньому етапі. Вихід – уточнений список магазинів.

На «Застосування фільтрації на основі популярності» система змінює розташування заклади у списку рекомендацій на основі метрик популярності, включно з переглядами, бронюваннями або покупками, що дає змогу побудувати порядок «від найбільш популярних до менш популярних». Вхідними даними виступає уточнений список магазинів. Вихід цього етапу – впорядкований список закладів, який передається до процесу «Фіналізація рекомендацій».

Четвертий процес – «Фіналізація рекомендацій» – включає розрахунок індексу популярності, застосування вагових коефіцієнтів та формування остаточного списку. Індекс популярності обчислюється на основі статистичних даних, з урахуванням вагових коефіцієнтів, які визначають важливість окремих критеріїв. Після цього заклади сортуються, і формується у фінальний список, який передається користувачу через інтерфейс системи. У цьому процесі також бере участь інформаційна система, яка забезпечує технічну реалізацію передачі даних. Четвертий процес має декомпозицію на підпроцеси, які зображені на рисунку 5.5.

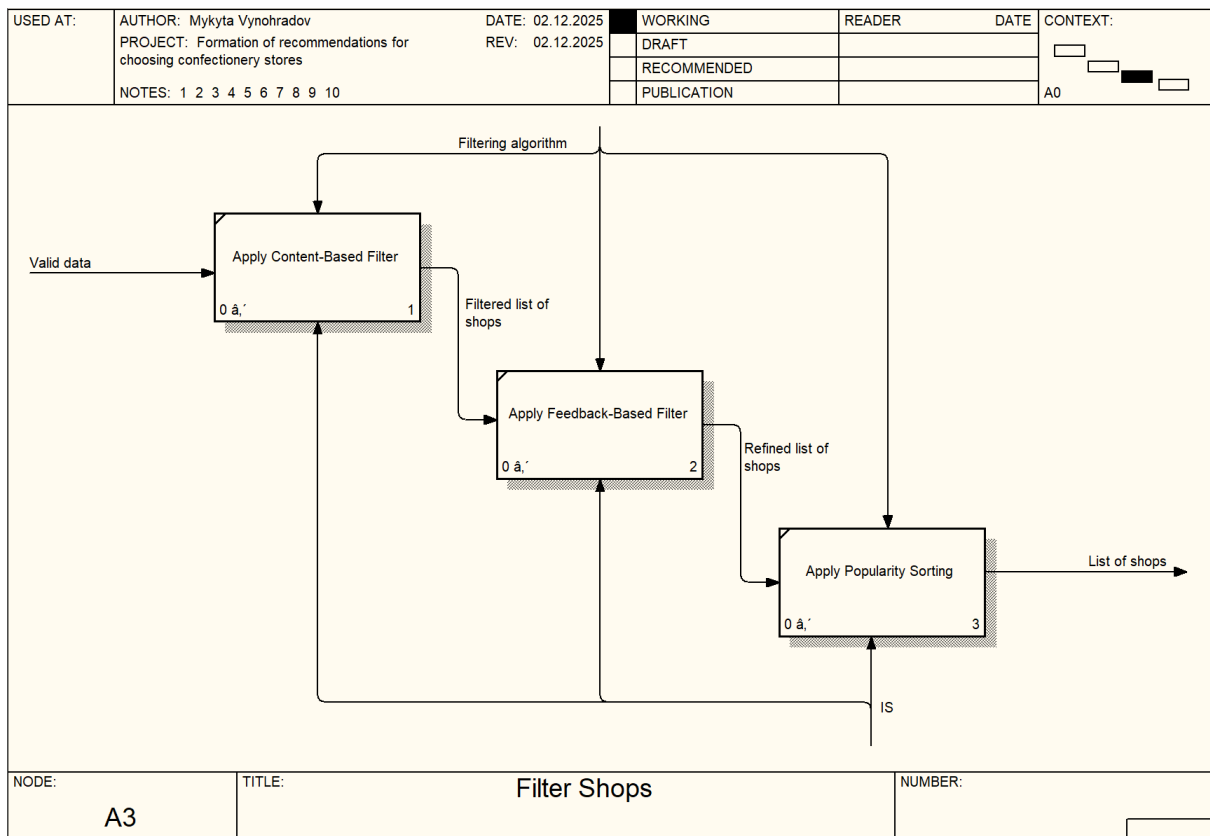


Рисунок 5.4 – IDEF0 Діаграма декомпозиції блоку «Фільтрація закладів»

Процес «Фіналізація рекомендацій» завершує формування рекомендацій, обчислюючи популярність, застосовуючи зважування та генеруючи фінальний список для подання користувачу. На підпроцесі «Розрахування індексу популярності» система перетворює сирі метрики активності на узагальнений індекс популярності для кожного закладу. Вхідними даними є відсортований список закладів і відповідні метрики (перегляди, покупки/бронювання). Результатом є дані індексу популярності для наступного етапу.

На «Застосування правил зважування» виконується зважування результатів з урахуванням пріоритетів – наприклад, підвищення ваги цінової категорії чи формату закладу залежно від налаштувань політики ранжування. Тут вхідні дані – обчислені індекси популярності. Виходом стає відсортований список з урахуванням зважувальних налаштувань.

На «Створення остаточного списку» система формує остаточний список рекомендованих кондитерських магазинів, застосовуючи алгоритм сортування й обмеження на кількість результатів, та готує його до подання через інтерфейс.

Вхідними даними є відсортований і зважений список. Вихід – фінальний список рекомендованих кондитерських магазинів, узгоджений із ІС та готовий до відображення користувачу.

Отже, побудована модель IDEF0 дозволяє чітко структурувати логіку роботи рекомендаційної системи, визначити всі необхідні компоненти та їхні взаємозв'язки. Вона є основою для подальшого програмного моделювання, тестування та впровадження системи у реальне середовище. Застосування нотації IDEF0 забезпечує формалізований підхід до опису функціональності, що відповідає вимогам інженерного проектування та академічного аналізу.

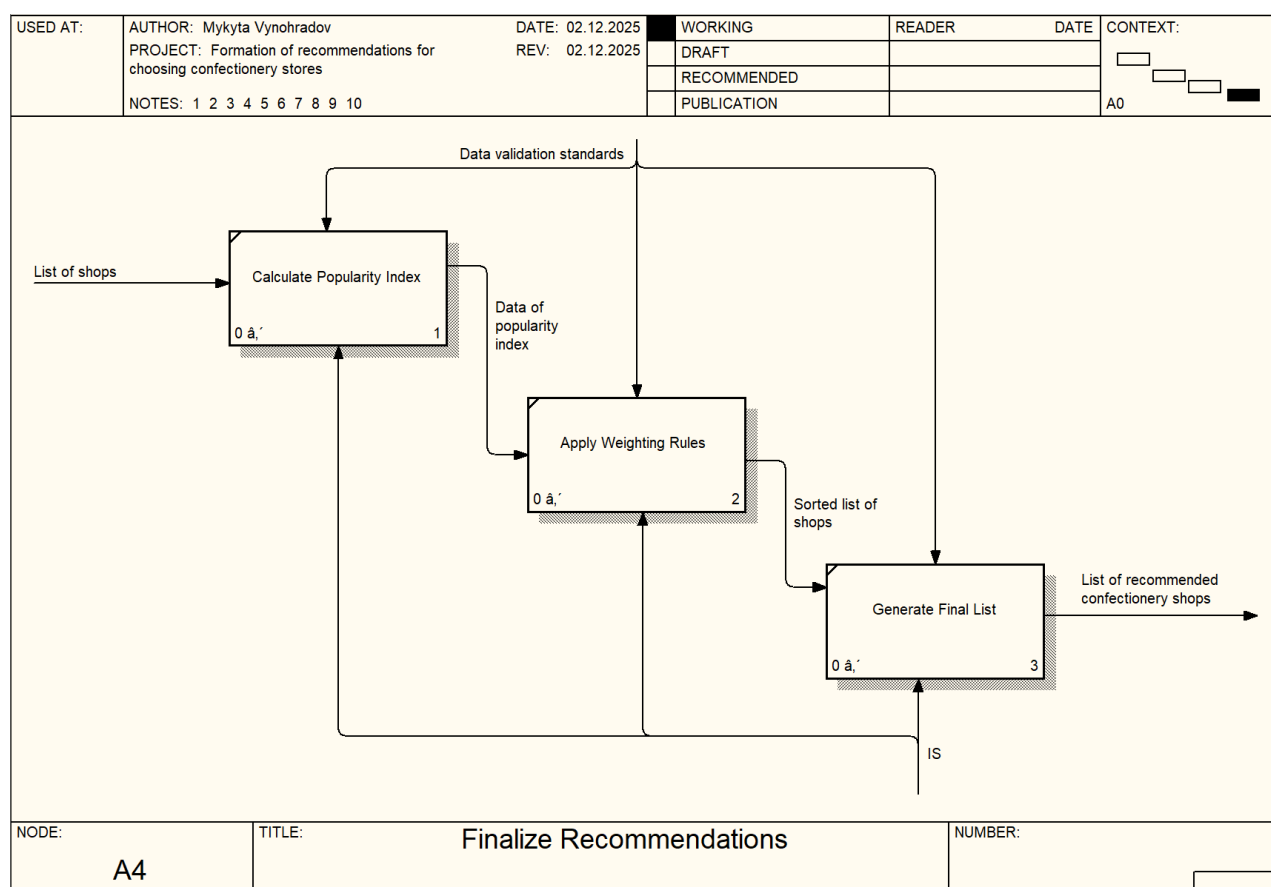


Рисунок 5.5 – IDEF0 Діаграма декомпозиції блоку «Фіналізація рекомендацій»

5.2.2 Розробка та опис діаграми UseCase діаграми

Діаграма прецедентів у нотації UML є формалізованим способом представлення взаємодії між користувачами системи (акторами) та її функціональними можливостями (прецедентами). Такий тип діаграми також називають діаграмою варіантів використання, оскільки вона демонструє, які дії можуть виконувати різні типи користувачів у межах системи, незалежно від конкретної реалізації програмної логіки. Основна мета побудови діаграми прецедентів полягає не в описі технічних деталей, а в узагальненому відображенні функціоналу, доступного для кожного актора.

На рисунку 5.6 представлено діаграму прецедентів для рекомендаційної системи вибору кондитерських магазинів. Вона охоплює основні сценарії взаємодії між користувачами та системою, зокрема для авторизованих користувачів, гостей та адміністратора.

У системі визначено три типи акторів:

– Авторизований користувач – це кінцевий користувач, який має доступ до персоналізованих функцій системи, зокрема реєстрації, редагування профілю, перегляду рекомендацій та взаємодії з фільтрами.

– Гість – користувач, який не проходив процедуру авторизації, але має доступ до базових функцій, таких як перегляд загального списку кондитерських магазинів та ознайомлення з їхньою детальною інформацією.

– Адміністратор – користувач із розширеними правами доступу, який відповідає за підтримку актуальності даних, конфігурацію системи та модерацію контенту.

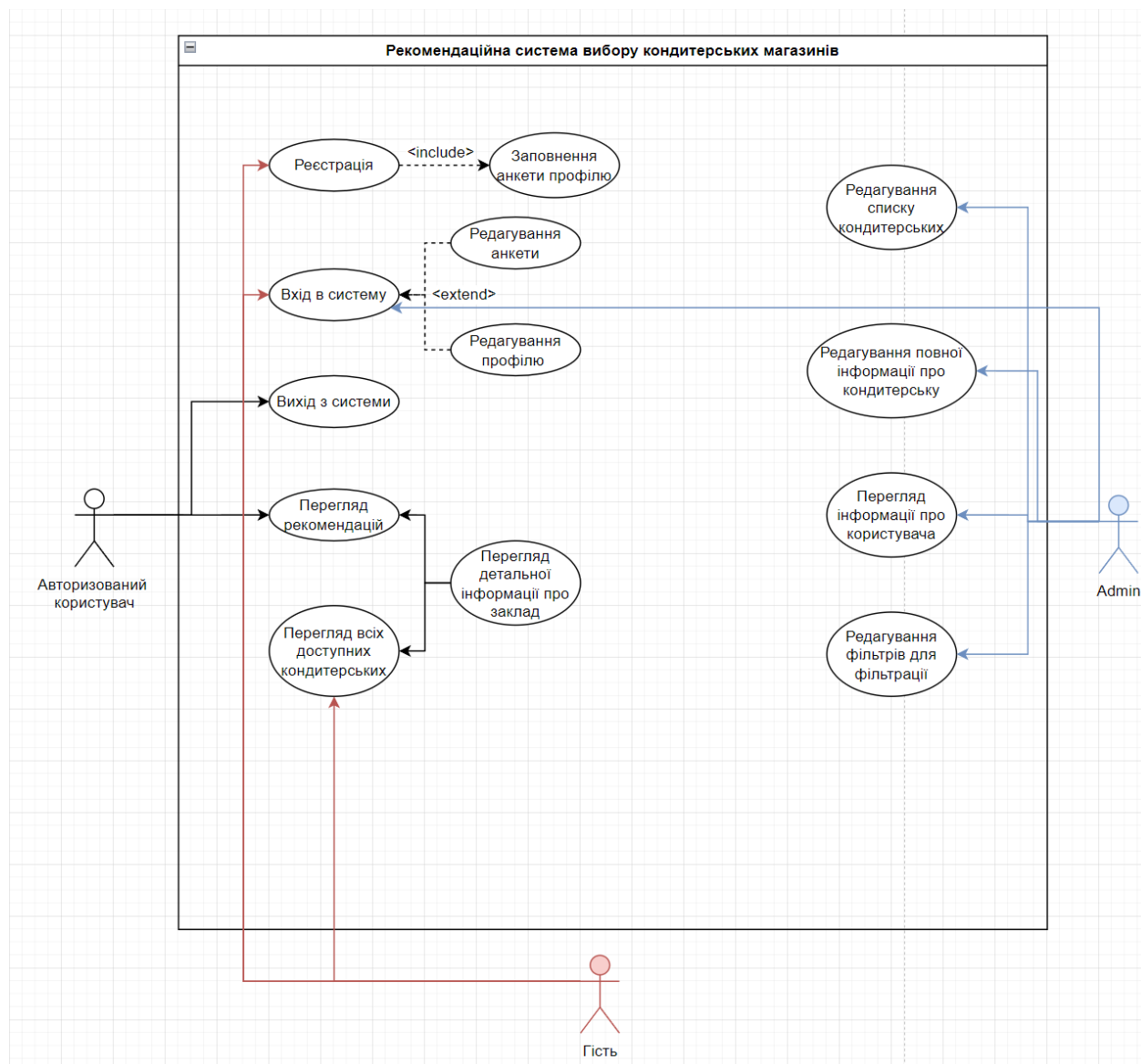


Рисунок 5.6 – UseCase діаграма

Для авторизованого користувача доступні наступні варіанти використання:

- Реєстрація, яка включає заповнення анкети профілю з уподобаннями щодо типу продукції, формату закладу, регіону та цінової категорії.
- Вхід у систему, що може розширюватися можливістю редагування профілю.
- Перегляд рекомендацій, сформованих на основі введених уподобань, з можливістю ознайомлення з детальною інформацією про кожен магазин.
- Перегляд усіх доступних кондитерських, незалежно від персоналізованих рекомендацій.

– Редагування фільтрів, що дозволяє користувачу самостійно налаштувати критерії пошуку магазинів.

Гість має доступ до таких прецедентів, як перегляд загального списку кондитерських магазинів та перегляд детальної інформації про кожен заклад, але не може змінювати фільтри чи зберігати персональні налаштування.

Адміністратор системи має доступ до розширених функцій:

– Редагування списку кондитерських магазинів, включаючи додавання, оновлення або видалення записів.

– Редагування повної інформації про кожен магазин (назва, тип продукції, формат, регіон, ціни, популярність).

– Перегляд інформації про користувачів, з метою аналітики або модерації.

– Налаштування фільтрів, типів продукції та форматів закладів, що використовуються в анкетах та при фільтрації.

– Редагування запитань анкети, яка заповнюється користувачем під час реєстрації.

Таким чином, діаграма прецедентів демонструє повний спектр функціональних можливостей системи для кожного типу користувача, що дозволяє чітко визначити межі доступу, сценарії використання та логіку взаємодії з рекомендаційною системою для вибору кондитерських магазинів.

6 ОПИС ПРИЙНЯТИХ ПРОЕКТНИХ РІШЕНЬ

У процесі розробки рекомендаційної системи для вибору кондитерських магазинів прийнято низку технічних рішень, які забезпечують ефективну реалізацію функціоналу, відповідність сучасним стандартам веб-розробки та зручність масштабування.

6.1 Обґрунтування мови програмування для реалізації серверної частини

Для реалізації серверної частини рекомендаційної системи обрано мову програмування Python, яка на сьогодні є однією з найпопулярніших і найзручніших мов для створення веб-сервісів, систем обробки даних та реалізації алгоритмів персоналізованих рекомендацій.

Згідно з актуальними статистичними даними, які надані на рисунку 6.1, Python займає друге місце за популярністю серед мов програмування, поступаючись лише TypeScript всього на 1%, і має 17.0% частки серед розробників. Такий високий показник свідчить про стабільну підтримку спільноти, наявність великої кількості бібліотек, документації та готових рішень, що значно спрощує розробку та підтримку системи.

У контексті побудови рекомендаційної системи для вибору кондитерських магазинів Python має низку переваг:

- Гнучкість у реалізації логіки фільтрації та ранжування. Завдяки бібліотекам Pandas, NumPy, Scikit-learn та іншим, Python дозволяє ефективно працювати з масивами даних, обчислювати індекси популярності, застосовувати вагові коефіцієнти та реалізовувати алгоритми рекомендацій.

- Зручна інтеграція з базами даних. Python має повну підтримку PostgreSQL через ORM-бібліотеки, такі як SQLAlchemy або Django ORM, що дозволяє легко працювати з реляційними структурами, необхідними для зберігання інформації про магазини, користувачів та їхні уподобання.

- Швидка розробка REST API. Фреймворки Flask і FastAPI дозволяють створювати легкі та масштабовані серверні додатки, які забезпечують обмін даними з клієнтською частиною через HTTP-запити.

- Підтримка OAuth-автентифікації. Python має готові рішення для інтеграції з Google OAuth API, що дозволяє реалізувати безпечну авторизацію користувачів.

У порівнянні з іншими мовами:

- JavaScript (13.4%) активно використовується для фронтенду, але на серверній частині має обмеження в обробці складних аналітичних задач.

- Java (11.8%) є потужною мовою для корпоративних систем, але має складніший синтаксис і довший цикл розробки.

- PHP (6.8%) традиційно застосовується для веб-сайтів, але менш придатний для реалізації рекомендаційних алгоритмів.

Отже, виходячи з інформації вище, робимо висновок, що Python є оптимальним вибором для серверної частини системи, оскільки поєднує простоту, гнучкість, високу популярність серед розробників та потужну екосистему для реалізації інтелектуальної логіки рекомендацій. Його використання дозволяє ефективно реалізувати всі функціональні вимоги системи, забезпечити надійність, масштабованість та зручність підтримки проєкту в довгостроковій перспективі.

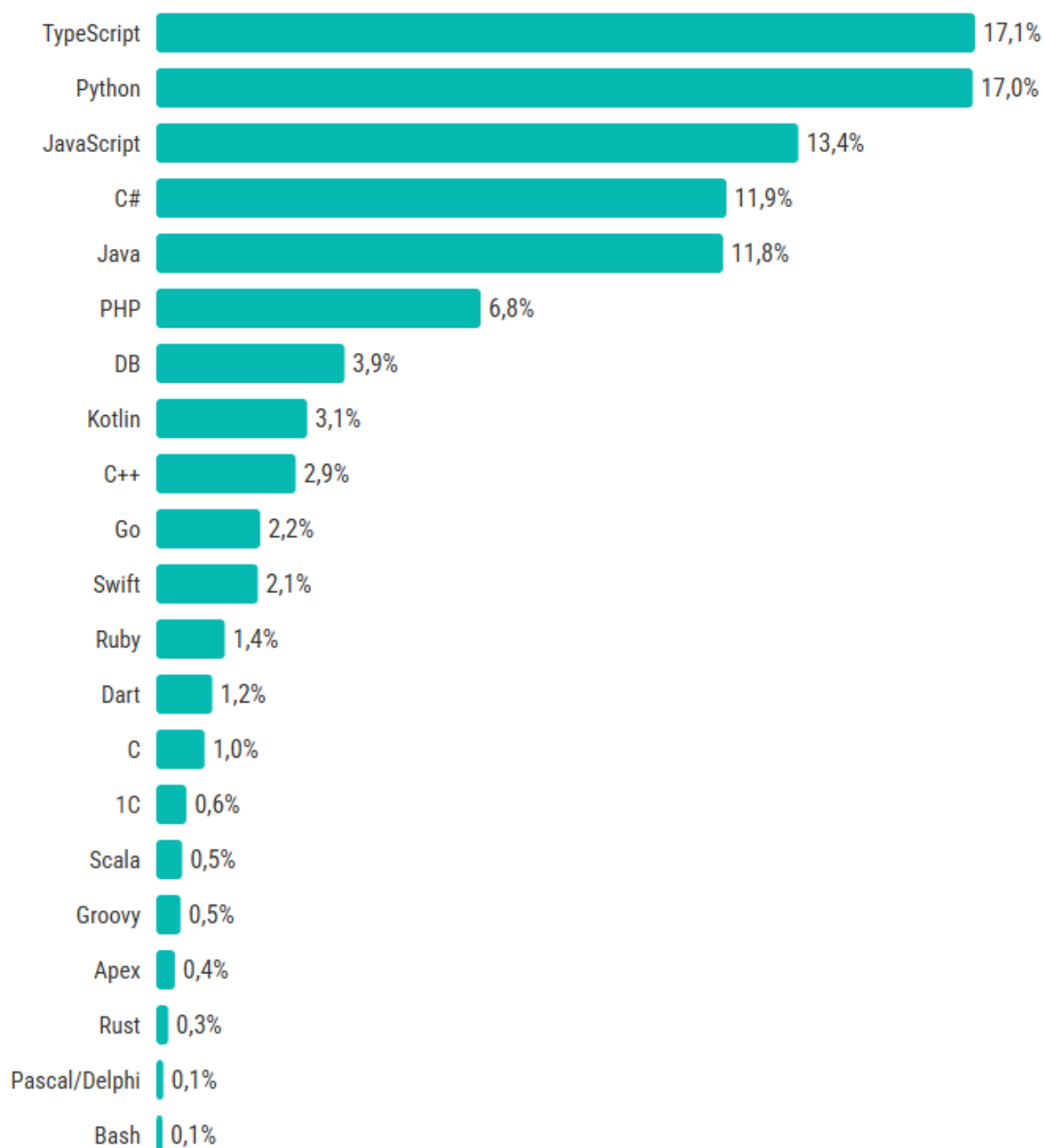


Рисунок 6.1 – Рейтинг мов програмування 2025

6.2 Обґрунтування інструментів для розробки клієнтської частини

Клієнтська частина рекомендаційної системи для вибору кондитерських магазинів реалізується у вигляді адаптивного веб-додатку, орієнтованого на мобільні пристрої. Для її створення обрано JavaScript як основну мову програмування, а також React.js – сучасну бібліотеку для побудови інтерфейсів користувача. Такий вибір зумовлений потребою у динамічному,

інтерактивному та масштабованому інтерфейсі, який здатен ефективно взаємодіяти з серверною частиною, реалізованою на Python.

JavaScript є стандартом для розробки клієнтських веб-додатків і має найвищу популярність серед мов фронтенду. Згідно з актуальними статистичними даними, JavaScript займає третє місце за популярністю (13.4%), поступаючись лише TypeScript і Python. Його універсальність, підтримка усіма сучасними браузерами та активна екосистема роблять його незамінним для реалізації інтерфейсної логіки.

React.js, як бібліотека JavaScript, дозволяє створювати компонентно-орієнтовані інтерфейси, які легко масштабуються, повторно використовуються та швидко оновлюються без перезавантаження сторінки. Це особливо важливо для рекомендаційної системи, де користувач постійно взаємодіє з фільтрами, переглядає списки магазинів та отримує персоналізовані результати.

Переваги використання JavaScript + React.js у рамках даної системи:

- Ідеальна сумісність із Python-бекендом. Через REST API або GraphQL клієнтська частина на JavaScript легко обмінюється даними з сервером, написаним на Python + Django.

- Висока продуктивність. React.js використовує віртуальний DOM, що дозволяє швидко оновлювати інтерфейс без зайвих перерендерів.

- Гнучкість у побудові UI. Компонентна архітектура дозволяє легко реалізувати окремі блоки: форма реєстрації, фільтри, карта, список рекомендацій, детальна інформація про магазин.

- Широка підтримка мобільної адаптації. Завдяки SCSS та медіа-запитам можна створити зручний інтерфейс для смартфонів, що відповідає вимогам сучасного UX.

- Активна спільнота та екосистема. Велика кількість готових бібліотек, таких як React Router, Axios, Material UI, дозволяє прискорити розробку.

Порівняння з альтернативами:

– jQuery: хоча й простий у використанні, не підтримує компонентну архітектуру, важко масштабувати, не оптимізований для Single Page Application.

– Vue.js: має простіший синтаксис, але менш поширений у великих проєктах, слабша інтеграція з корпоративними інструментами.

– Angular: потужний фреймворк, але має складну структуру, високий поріг входу та надлишкову складність для проєктів середнього масштабу.

React.js чудово поєднується з Python-бекендом через стандартні протоколи HTTP-запитів. Наприклад, бекенд на Django може надавати JSON-відповіді, які легко обробляються на фронтенді через Axios або Fetch API. Така архітектура дозволяє чітко розділити логіку обробки даних на сервері та логіку відображення на клієнті, що відповідає принципам сучасної веб-розробки.

6.3 Обґрунтування вибору СУБД

Система управління базами даних (СУБД) — це програмно-лінгвістичне середовище, що забезпечує створення, зберігання, обробку та управління даними в інформаційних системах. Вона виконує критично важливі функції, серед яких:

- управління даними в зовнішній пам'яті;
- фіксація змін, резервне копіювання та відновлення після збоїв;
- оптимізація доступу до даних в оперативній пам'яті з використанням кешування;
- підтримка мов баз даних – мови визначення даних (DDL) та мови маніпулювання даними (DML).

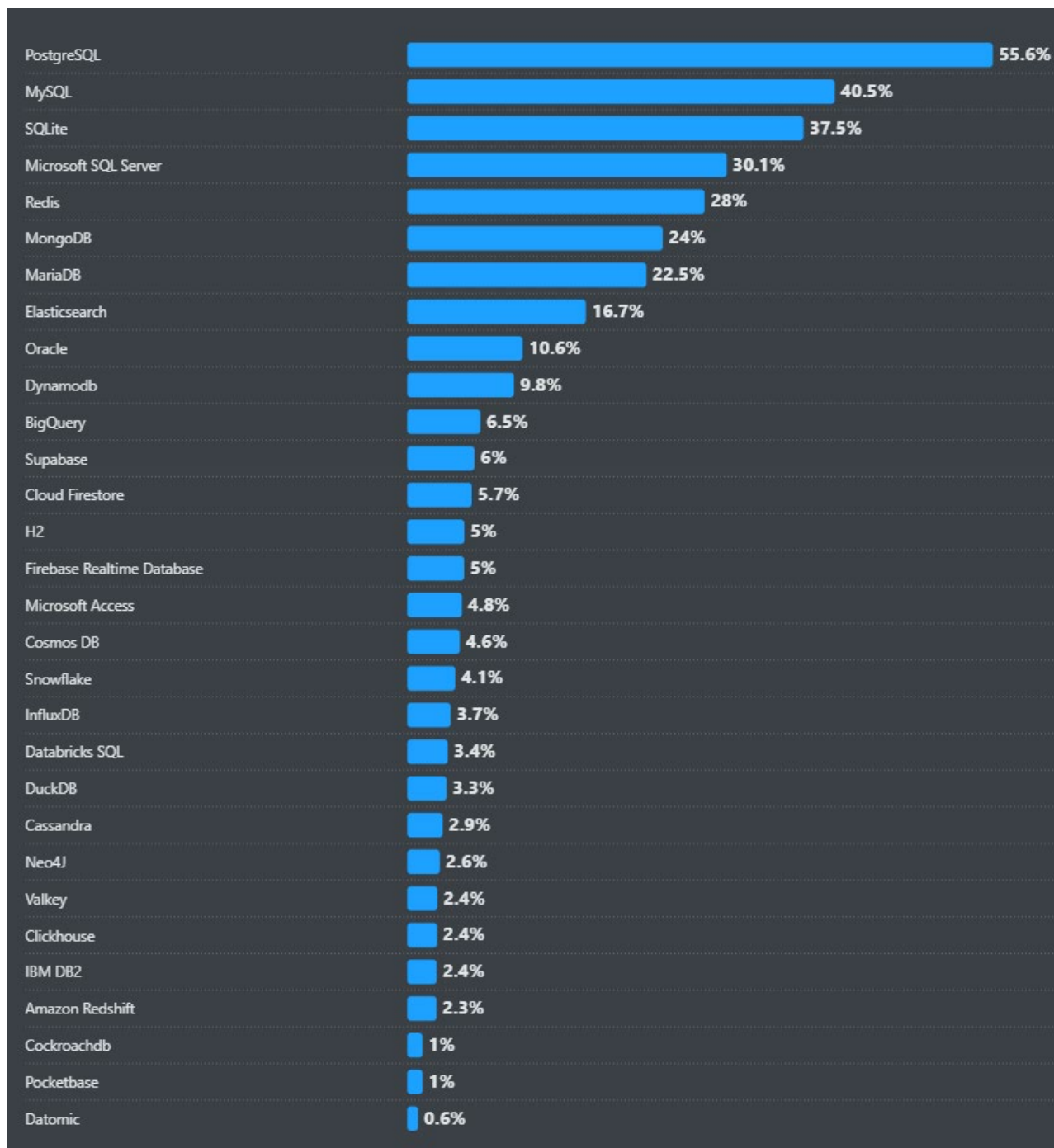


Рисунок 6.2 – Рейтинг СУБД 2025

Для реалізації рекомендаційної системи вибору кондитерських магазинів обрано реляційну СУБД PostgreSQL, яка є однією з найпотужніших і найнадійніших систем з відкритим кодом. Згідно з актуальними статистичними даними, PostgreSQL займає перше місце за популярністю серед СУБД, охоплюючи 55.6% серед розробників, що значно перевищує показники MySQL (40.5%), SQLite (37.5%) та MongoDB (24%). Такий рівень поширеності свідчить

про стабільну підтримку, активну спільноту, регулярні оновлення та високу довіру серед професіоналів.

Переваги PostgreSQL у контексті даної системи:

– Завдяки бібліотекам `psycopg2`, `SQLAlchemy` та `Django ORM`, PostgreSQL легко інтегрується з серверною частиною, що реалізована на Python.

– PostgreSQL дозволяє реалізовувати багаторівневу фільтрацію, сортування, агрегацію та обчислення індексів популярності, що є основою логіки рекомендацій.

– Система підтримує ACID-властивості, що гарантує цілісність даних навіть у разі збоїв.

– PostgreSQL підтримує JSON, масиви, геодані (через PostGIS), що дозволяє зберігати як структуровану, так і напівструктуровану інформацію про кондитерські.

– Система легко адаптується до зростання обсягу даних, що важливо при розширенні бази кондитерських магазинів або збільшенні кількості користувачів.

Порівняння з альтернативами:

– MySQL – популярна СУБД, але має обмежену підтримку складних типів даних і менш гнучку систему розширень.

– SQLite – легка та проста, але не придатна для багатокористувацьких систем із високим навантаженням.

– MongoDB – NoSQL-рішення, добре працює з неструктурованими даними, але не забезпечує реляційні зв'язки, що критично для зв'язку між користувачами, магазинами та їхніми атрибутами.

6.4 Моделювання даних інформаційної системи

Для проведення моделювання даних рекомендаційної системи для вибору кондитерських магазинів використано програмний засіб MySQL Workbench. На рисунку 6.3 представлена фізична модель даних веб-додатку, яка відображає всі сутності предметної області та зв'язки між ними.

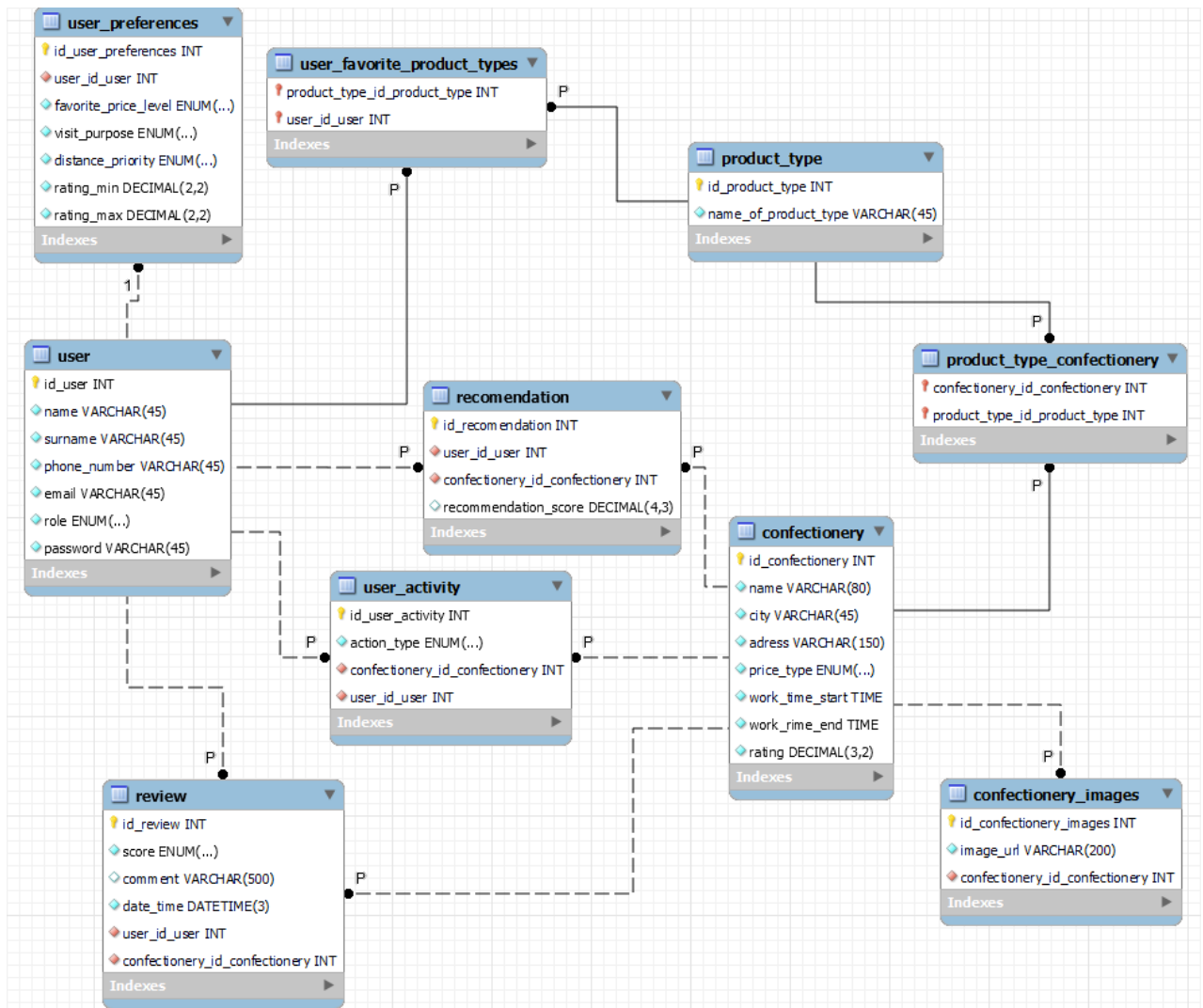


Рисунок 6.3 – Фізична модель даних

Архітектура бази даних побудована відповідно до принципів третьої нормальної форми, що забезпечує відсутність надмірності, логічну цілісність і масштабованість системи.

Фізична модель бази даних рекомендаційної системи розроблена з урахуванням вимог до збереження персональних даних користувачів, їх анкетних характеристик, поведінкової активності, відгуків, інформації про кондитерські заклади та результатів роботи рекомендаційного алгоритму. Повний перелік сутностей, їх атрибутів, типів даних та призначення наведено в таблиці 6.1.

Центральною сутністю фізичної моделі є таблиця `user`, яка використовується для збереження даних про зареєстрованих користувачів системи. Вона містить унікальний ідентифікатор користувача, особисті дані, а також службову інформацію, необхідну для авторизації та розмежування прав доступу. Усі інші сутності, що відображають активність користувача або результати рекомендацій, мають зовнішній ключ на цю таблицю, що забезпечує логічну цілісність даних.

Анкетні дані користувачів зберігаються в таблиці `user_preferences`, структура якої також наведена у таблиці 6.1. У цій таблиці фіксуються такі параметри, як бажаний рівень цін, мета відвідування закладу, пріоритет відстані, а також допустимий діапазон рейтингу закладів. Дані з цієї таблиці використовуються системою як контентні ознаки профілю користувача під час формування первинних рекомендацій.

Для реалізації можливості вибору кількох улюблених типів продукції між сутностями `user` та `product_type` введена проміжна таблиця `user_favorite_product_types`, яка фізично реалізує зв'язок багато-до-багатьох. Завдяки цьому кожен користувач може мати кілька улюблених категорій продукції, а один тип продукції може бути обраний багатьма користувачами. Інформація про самі типи продукції зберігається в довідниковій таблиці `product_type`.

Інформація про кондитерські заклади зберігається у таблиці `confectionery`, яка містить назву закладу, місто, адресу розташування, тип цінової політики, графік роботи та середній рейтинг. Дана таблиця є ключовою, оскільки саме її об'єкти є кінцевою ціллю рекомендаційної системи. Для збереження

мультимедійного контенту використовується таблиця `confectionery_images`, яка забезпечує можливість прив'язки кількох зображень до одного закладу.

Зв'язок між закладами та типами продукції реалізується за допомогою таблиці `product_type_confectionery`, що дозволяє одному закладу мати у своєму асортименті кілька категорій продукції. Така структура є необхідною для коректної роботи рекомендаційного алгоритму, який враховує відповідність улюблених типів продукції користувача фактичному асортименту закладу.

Для реалізації колаборативної складової рекомендаційної системи використовується таблиця `review`, у якій зберігаються відгуки користувачів про заклади. Вона містить числову оцінку, текстовий коментар, дату та час створення відгуку, а також зовнішні ключі на таблиці `user` та `confectionery`. Дані цієї таблиці використовуються для побудови матриці «користувач – заклад» та визначення подібності між користувачами.

Поведінкова складова рекомендаційної системи реалізується через таблицю `user_activity`, в якій фіксуються неявні дії користувачів, зокрема перегляди, кліки та додавання закладів до обраного. Вона містить зовнішні ключі на таблиці `user` і `confectionery` та тип дії. Дані з цієї таблиці використовуються для аналізу поведінкових патернів користувачів та уточнення рекомендацій.

Результати роботи рекомендаційного алгоритму зберігаються в таблиці `recommendation`, що містить зв'язок між конкретним користувачем та закладом, а також числовий показник `recommendation_score`, який відображає рівень релевантності рекомендації. Саме ця таблиця використовується для формування персонального списку рекомендованих закладів у користувацькому інтерфейсі системи.

Усі зв'язки між сутностями реалізовані з використанням зовнішніх ключів із визначеними обмеженнями посилальної цілісності. Повний перелік таких обмежень із зазначенням таблиць-джерел, таблиць-призначень та правил підтримки цілісності при операціях оновлення та видалення наведено в таблиці 6.2.

Обмеження типу ON UPDATE CASCADE та ON DELETE CASCADE застосовуються для таблиць, що безпосередньо залежать від користувачів або закладів, зокрема для відгуків, активності користувачів, зображень закладів та рекомендацій. Це гарантує автоматичне видалення або оновлення залежних записів у разі зміни або видалення головного об'єкта. Для довідникових таблиць, таких як product_type, використано обмеження типу RESTRICT, що унеможливорює видалення записів, які вже використовуються в інших таблицях.

Таблиця 6.1 – Сутності фізичної моделі даних

№	Назва сутності	Назва атрибуту	Тип даних	Призначення
1	user	id_user	INT	Первинний ключ користувача
		name	VARCHAR(45)	Ім'я користувача
		surname	VARCHAR(45)	Прізвище користувача
		phone_number	VARCHAR(45)	Номер телефону
		email	VARCHAR(45)	Email користувача
		role	ENUM	Роль користувача
		password	VARCHAR(45)	Пароль
2	user_preferences	id_user_preferences	INT	Первинний ключ
		user_id_user	INT	Зовнішній ключ на user
		favorite_price_level	ENUM	Цінові вподобання
		visit_purpose	ENUM	Мета візиту
		distance_priority	ENUM	Пріоритет відстані
		rating_min	DECIMAL(2,2)	Мінімальний рейтинг
		rating_max	DECIMAL(2,2)	Максимальний рейтинг
3	product_type	id_product_type	INT	Первинний ключ
		name_of_product_type	VARCHAR(45)	Назва типу
4	user_favorite_product_types	user_id_user	INT	FK на user
		product_type_id_product_type	INT	FK на product_type
5	confectionery	id_confectionery	INT	Первинний ключ
		name	VARCHAR(80)	Назва закладу
		city	VARCHAR(45)	Місто

		adress	VARCHAR(150)	Адреса
--	--	--------	--------------	--------

Продовження таблиці 6.1

№	Назва сутності	Назва атрибуту	Тип даних	Призначення
		price_type	ENUM	Цінова категорія
		work_time_start	TIME	Початок роботи
		work_time_end	TIME	Кінець роботи
		rating	DECIMAL(3,2)	Середній рейтинг
6	confectionery_images	id_confectionery_images	INT	Первинний ключ
		image_url	VARCHAR(200)	Посилання на зображення
		confectionery_id_confectionery	INT	FK на confectionery
7	review	id_review	INT	Первинний ключ
		score	ENUM	Оцінка
		comment	VARCHAR(500)	Текст відгуку
		date time	DATETIME	Час створення
		user id user	INT	FK на user
		confectionery_id_confectionery	INT	FK на confectionery
8	user_activity	id_user_activity	INT	Первинний ключ
		action_type	ENUM	Тип дії
		user_id user	INT	FK на user
		confectionery_id_confectionery	INT	FK на confectionery
9	recommendation	id_recomendation	INT	Первинний ключ
		user id user	INT	FK на user
		confectionery_id_confectionery	INT	FK на confectionery
		recommendation_score	DECIMAL(4,3)	Вага рекомендації

Таблиця 6.2 – Сутності фізичної моделі даних

Назва сутності	Поле FK	Назва сутності 2	Поле PK	ON UPDATE	ON DELETE
user_preferences	user_id_user	user	id_user	CASCADE	CASCADE
user_favorite_product_types	user_id_user	user	id_user	CASCADE	CASCADE
user_favorite_product_types	product_type_id_product_type	product_type	id_product_type	RESTRICT	RESTRICT
product_type_confectionery	confectionery_id_confectionery	confectionery	id_confectionery	RESTRICT	RESTRICT

Продовження таблиці 6.1

Назва сутності	Поле FK	Назва сутності 2	Поле PK	ON UPDATE	ON DELETE
product_type_confectionery	product_type_id_product_type	product_type	id_product_type	RESTRICT	RESTRICT
confectionery_images	confectionery_id_confectionery	confectionery	id_confectionery	CASCADE	CASCADE
review	user_id_user	user	id_user	CASCADE	CASCADE
review	confectionery_id_confectionery	confectionery	id_confectionery	CASCADE	CASCADE
user_activity	user_id_user	user	id_user	CASCADE	CASCADE
user_activity	confectionery_id_confectionery	confectionery	id_confectionery	CASCADE	CASCADE
recommendation	user_id_user	user	id_user	CASCADE	CASCADE
recommendation	confectionery_id_confectionery	confectionery	id_confectionery	CASCADE	CASCADE

6.5 Розробка гібридного методу фільтрування рекомендацій

У процесі розробки рекомендаційної системи для підбору кондитерських закладів реалізовано гібридний метод фільтрування, який поєднує в собі переваги контентної фільтрації, колаборативної фільтрації та поведінкового аналізу користувача. Такий підхід дозволяє значно підвищити точність рекомендацій, забезпечити персоналізацію результатів та усунути проблему дефіциту даних, що є характерною для систем із класичними методами фільтрації. [20], [26], [29]. Гібридний метод використовується як основний механізм формування персоналізованих рекомендацій у розробленій інформаційній системі та безпосередньо пов'язаний із фізичною моделлю бази даних, зокрема з таблицями user, user_preferences, user_favorite_product_types, review, user_activity, confectionery та recommendation.

Процес формування рекомендацій починається з перевірки факту авторизації користувача в системі. Якщо користувач не є авторизованим, система не має доступу до його персональних даних та історії активності, тому рекомендації у такому випадку формуються на основі загальної популярності

закладів. Для цього використовуються агреговані показники рейтингу та кількості відгуків.

У випадку, коли користувач успішно пройшов авторизацію, система перевіряє, чи заповнено анкету вподобань, дані якої зберігаються у таблиці `user_preferences`. Якщо анкету не заповнено, користувачу пропонується пройти опитування для формування базового профілю інтересів, після чого система переходить до контентної фільтрації.

Наступним етапом є перевірка, чи має система достатній обсяг інформації про поведінкову активність користувача у таблиці `user_activity`, а також оцінки у таблиці `review`. Якщо таких даних недостатньо, рекомендації формуються переважно на основі контентної фільтрації. У протилежному випадку активується повноцінний гібридний механізм, що поєднує всі три складові.

Контентна фільтрація базується на аналізі явних вподобань користувача, що зберігаються у таблицях `user_preferences` та `user_favorite_product_types`. До таких параметрів належать бажаний рівень цін (`favorite_price_level`), мета візиту (`visit_purpose`), пріоритет розташування (`distance_priority`), допустимий діапазон рейтингу (`rating_min`, `rating_max`), а також вибрані користувачем типи продукції.

На основі цих даних здійснюється початковий відбір закладів із таблиці `confectionery`. Наприклад, якщо користувач обрав середній рівень цін, тип продукції «торти» та мету візиту «подарунок», система відбирає лише ті заклади, які відповідають цим умовам та мають відповідний асортимент. Таким чином, контентна фільтрація формує першу множину кандидатів для рекомендацій.

Колаборативна фільтрація ґрунтується на аналізі оцінок та відгуків інших користувачів, що зберігаються у таблиці `review`. На основі цих даних формується матриця взаємодії «користувач – заклад», яка використовується для визначення ступеня подібності між користувачами. Якщо два користувачі мають схожі оцінки для певних закладів, система вважає їхні інтереси подібними. Відповідно, заклади, які отримали високі оцінки від схожих

користувачів, можуть рекомендуватися поточному користувачеві навіть у разі відсутності його власного досвіду взаємодії з такими закладами. Цей механізм дозволяє формувати персоналізовані рекомендації на основі досвіду спільноти користувачів.

Поведінкова складова реалізується на основі аналізу неявних дій користувача, що фіксуються у таблиці `user_activity`. До таких дій належать перегляди закладів, кліки по елементах інтерфейсу та додавання закладів до обраного. На відміну від відгуків, які є явною формою зворотного зв'язку, поведінкові дані дозволяють виявляти приховані інтереси користувача. Наприклад, багаторазові перегляди сторінки закладу або часте додавання до обраного свідчать про високий рівень зацікавленості.

Для різних типів дій використовуються різні вагові коефіцієнти, що дозволяє визначити ступінь впливу кожної дії на фінальний рейтинг рекомендації.

На фінальному етапі всі три складові – контентна, колаборативна та поведінкова фільтрація – інтегруються у єдиний числовий показник `recommendation_score`. Для кожного потенційного закладу обчислюється зважена сума показників відповідності анкетним даним, результатів аналізу подібності користувачів та рівня активної взаємодії. Отримані значення зберігаються у таблиці `recommendation`, де фіксується зв'язок між користувачем та закладом. Після цього результати додатково сортуються за універсальним коефіцієнтом популярності, який формується на основі рейтингу та кількості відгуків.

Фінальний відсортований список передається на клієнтську сторону інформаційної системи для відображення у вигляді персоналізованих рекомендацій користувачеві.

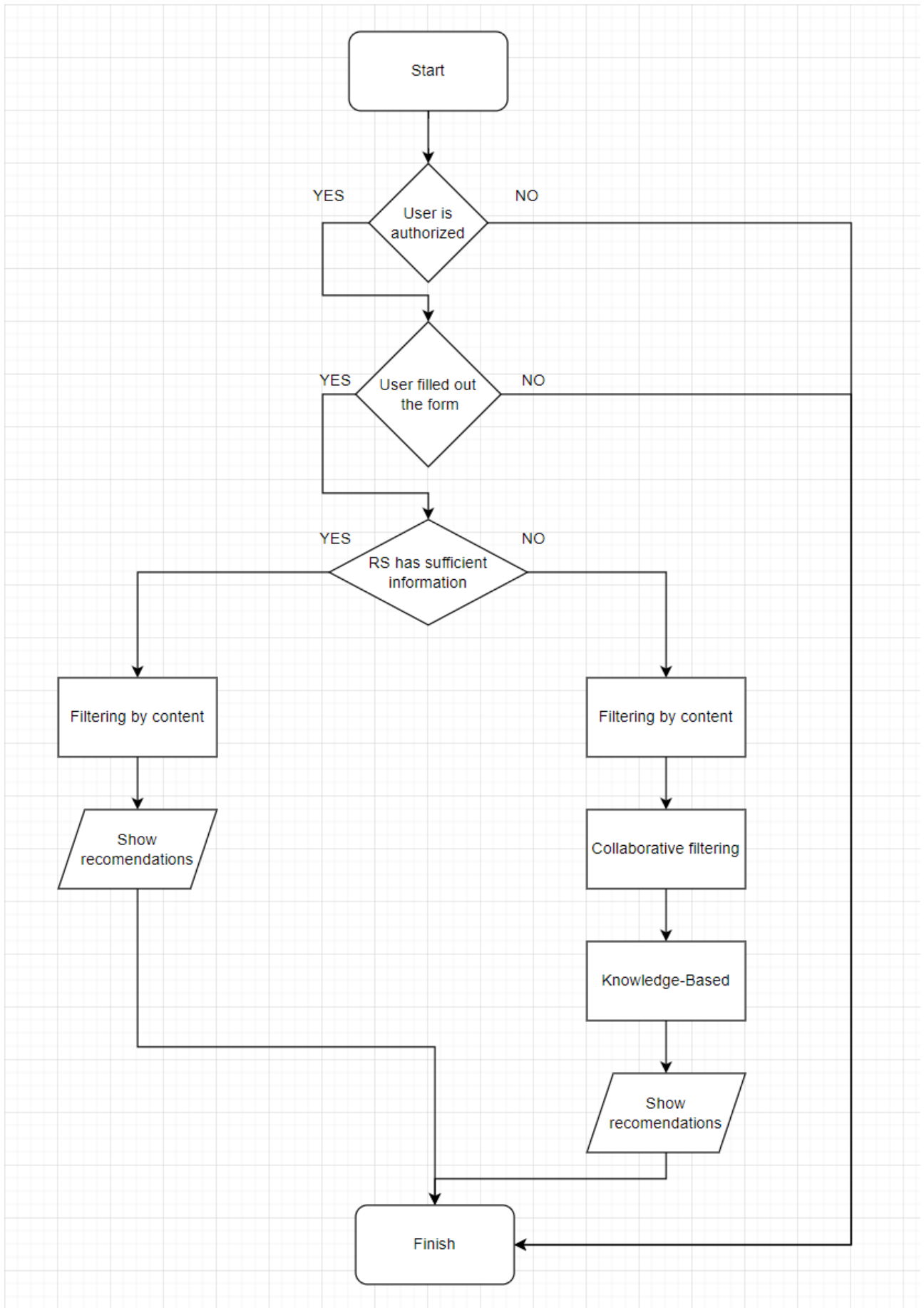


Рисунок 6.8 – Блок схема алгоритму процесу підбору рекомендацій

6.6 Розробка інтерфейсу клієнтської частини

На даному етапі створено макети ключових сторінок за допомогою сервісу Figma та виконано їх впровадження до рекомендаційної системи використовуючи Python + Django, HTML, React та CSS. При створенні макетів враховано всі системні та функціональні вимоги.

На рисунку 6.9 зображено домашню сторінку рекомендаційної системи для неавторизованого користувача. На ній розміщена назва та декілька кнопок, дві з яких призначенні для реєстрації та авторизації, а третя для пошуку за назвою.

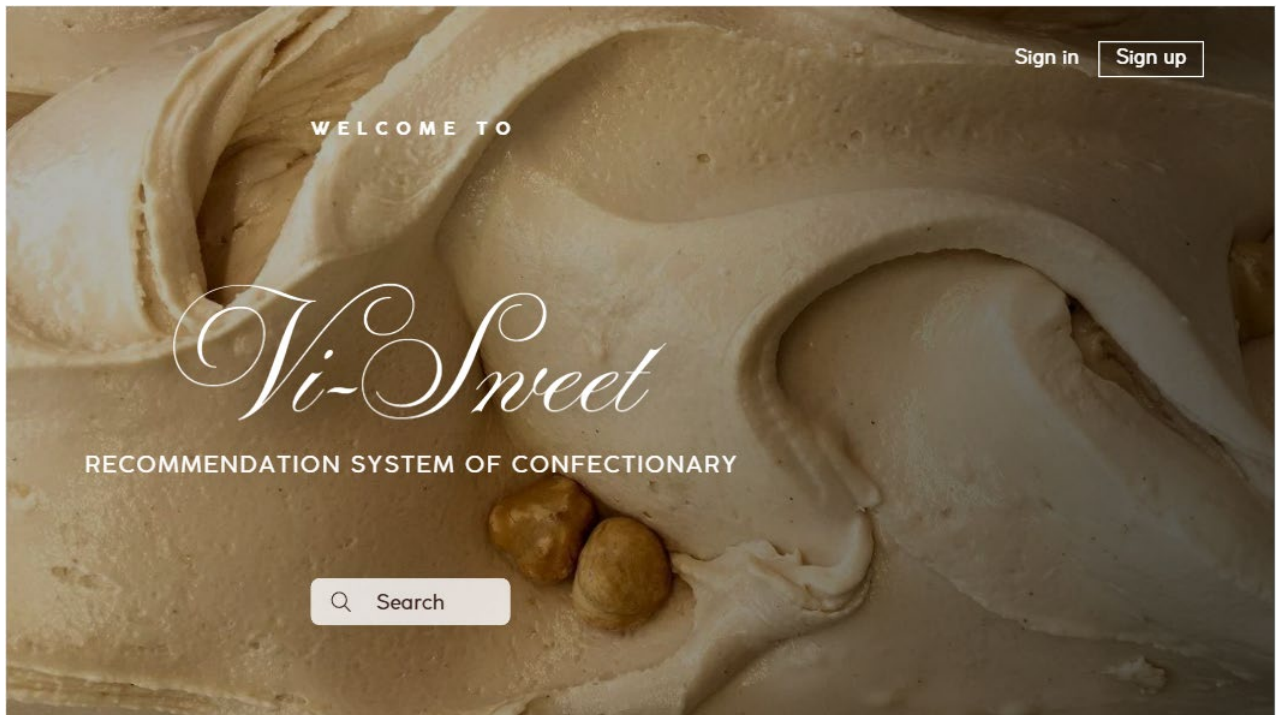


Рисунок 6.9 – Домашня сторінка рекомендаційної системи

При натисканні на кнопку авторизації, користувач потрапляє на сторінку авторизації, де має можливість авторизуватися за допомогою email та паролю, або за допомогою Google. Ця сторінка продемонстрована на рисунку 6.10.

При натисканні на кнопку реєстрації, користувач потрапляє на сторінку реєстрації, де йому треба заповнити таку інформацію про себе, як: ім'я, прізвище, email, пароль та номер телефону, або ж обрати реєстрацію за допомогою Google. Ця сторінка наведена на рисунку 6.11.

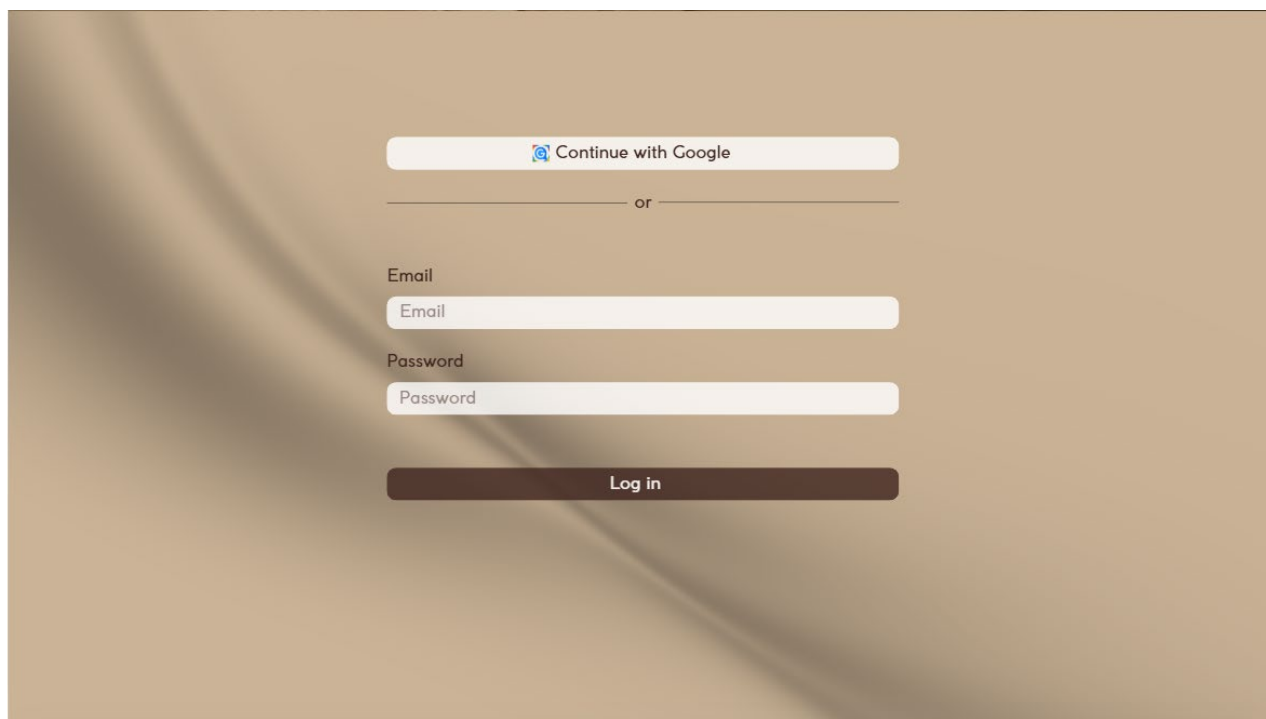


Рисунок 6.10 – Сторінка авторизації

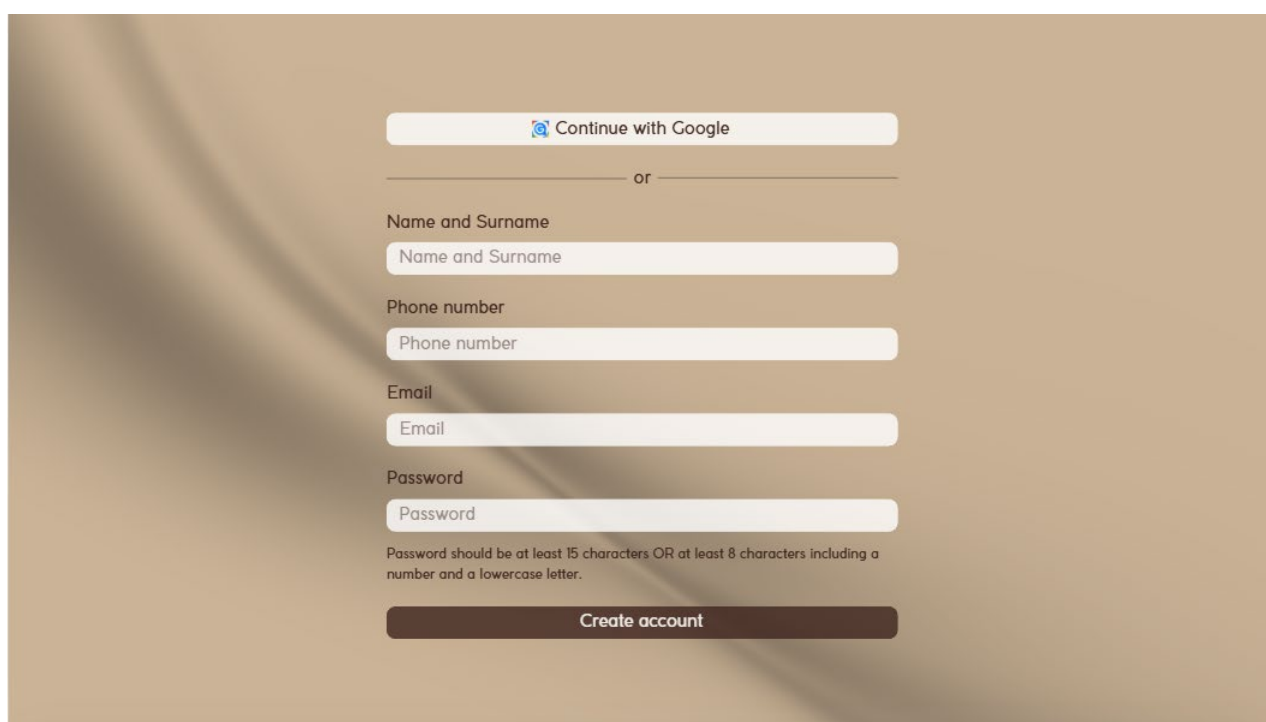


Рисунок 6.11 – Сторінка реєстрації

Після реєстрації, користувач повинен пройти анкетування, для отримання коректних послуг від рекомендаційної системи. Переходячи на сторінку анкетування, за умови, що це перше проходження анкети, користувач бачить

перед собою питання, на які, він має обрати 1 або декілька варіантів відповіді, які для нього є прийнятні. Перше питання до користувача, це його улюбленні типи десертів, наприклад, торт чи тістечко, шоколад чи пряник, а при натисканні на вид кондитерського виробу, користувач також може вибрати підвид виробу. Наступним ділом, користувачу треба обрати прийнятний діапазон цін, які його задовільнять, наприклад від 100 грн до 300 грн, або від 2000 грн до 5000 грн. Далі користувачу треба обрати, яку ціль відвідування він має, наприклад, день народження, просто повсякденне вживання і тд. Четверте питання стосується пріоритетного розташування кондитерської, наприклад, в межах 2 км, в межах міста, чи в межах області. І останнє, не менш важливе питання порівняно з іншими, це мінімальний та максимально бажаний рейтинг кондитерської. Після відповіді на всі питання, користувач натискає кнопку, згенерувати рекомендації, та його перекидає до рекомендацій кондитерських, які підпадають, під його бажані критерії.

Please tick the items that apply to you

1) How did you hear about this site?*

- From friends
- From a newsletter
- From advertising

2) Cakes*

- Sponge Cake
- Chocolate Cake
- Honey Cake
- Napoleon Cake
- Cheesecake
- Tiramisu
- Carrot Cake
- Red Velvet Cake
- Mousse Cake
- Mirror Glaze Cake
- Tiered Cake
- Wedding Cake

3) Ice Cream & Frozen Desserts

- Ice Cream
- Gelato
- Sorbet

4) Purpose of visit to the confectionery store

- Everyday visit
- For a holiday
- Custom order
- Gift options

5) In what range should the establishment be located?

- Within a range of 2 km
- Within a range of 5 km
- Within a range of 10 km
- Within the city
- Within the region
- Within the country

Рисунок 6.12 – Сторінка анкетування

У випадку, якщо користувач потрапляє на сторінку анкетування не в перший раз, то він має змогу змінити вже заповнену анкету, під поточні критерії, та отримати новий список рекомендованих кондитерських магазинів. Сторінка анкетування та її вигляд продемонстровано на рисунку 6.12.

Також користувач має можливість, на окремо виділений для цього сторінці, подивитися весь перелік кондитерських магазинів, без прив'язки до своїх відповідей в анкеті. Сторінка з всім кондитерськими магазинами наведена на рисунку 6.13.



Рисунок 6.13 – Сторінка з всім кондитерськими магазинами

У користувача повинна бути змога подивитися повну інформацію кондитерського магазину, а саме, адресу, фотографії, контакти, відгуки та перелік продукції, які продає певний магазин, це можливо за допомогою натискання на певну кондитерську і користувач потрапляє на персоналізовану сторінку. Її вигляд наведений на рисунку 6.14.



Рисунок 6.14 – Сторінка певного кондитерського магазину

Кожен з користувачів має можливість переглянути та змінити свої данні, які він вписав на етапі реєстрації, а саме: ім'я, прізвище, email, пароль та номер телефону, так само як і змінити відповіді на питання в анкетуванні, або просто переглянути свої відповіді. Це все користувач може зробити та переглянути на сторінці користувача, яка зображена на рисунку 6.15.

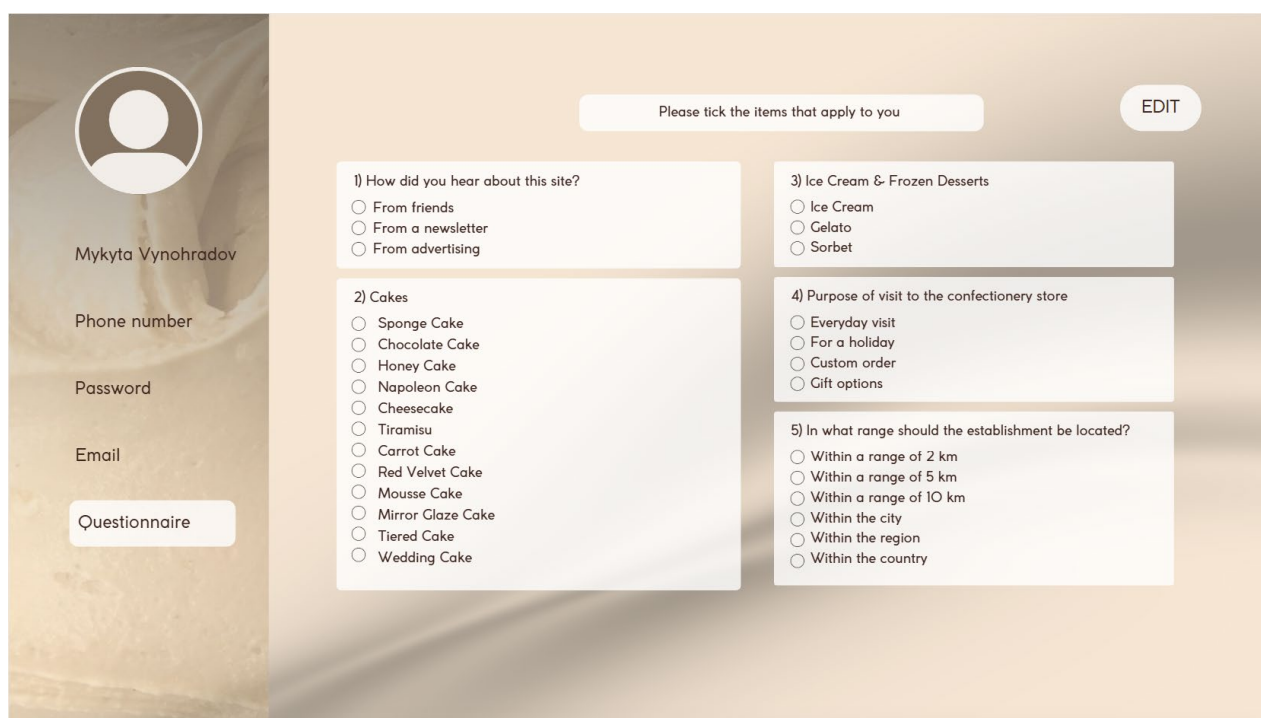


Рисунок 6.15 – Профіль користувача

6.7 Тестування розробленого програмного забезпечення

Для перевірки коректності функціонування розробленої рекомендаційної системи та її відповідності встановленим вимогам застосовано метод ручного тестування. Сутність даного підходу полягає у безпосередньому виконанні тестових сценаріїв користувачем без використання автоматизованих засобів. У процесі тестування перевіряючий імітує реальну поведінку користувача, взаємодіючи з функціональними можливостями системи з метою виявлення можливих помилок у роботі програмного забезпечення.

Ручне тестування дозволило перевірити логіку роботи основних модулів системи, зокрема модулів авторизації, введення анкетних даних, формування рекомендацій, обробки відгуків та аналізу користувацької активності. У межах професійного підходу до тестування також було використано підготовлені тестові сценарії, що описують послідовність дій користувача та очікуваний результат виконання кожної операції.

Для проведення тестування обрано такі основні функціональні можливості системи:

- визначення уподобань користувача;
- відображення списку рекомендованих кондитерських магазинів;
- відображення детальної інформації про кондитерський магазин.

Перша функціональна можливість системи – визначення уподобань користувача.

Очікувана поведінка: Після реєстрації та першого входу до системи користувач повинен автоматично потрапити на сторінку анкетування, де він має заповнити форму для визначення своїх уподобань щодо вибору кондитерських магазинів. Метою цього анкетування є отримання системою необхідних параметрів для формування персоналізованих рекомендацій. Форма повинна містити можливість вибору улюблених типів десертів та їх підвидів, встановлення бажаного діапазону цін, визначення мети відвідування закладу,

пріоритетного розташування та мінімально/максимально прийнятного рейтингу.

Передбачаються такі варіанти використання:

- Користувач уже проходив анкетування раніше. У такому разі система не повинна автоматично перенаправляти його на форму визначення уподобань. Користувачу має відобразитися стандартний інтерфейс, а доступ до анкети можливий через сторінку профілю з можливістю редагування попередніх відповідей.

- Користувач намагається надіслати незаповнену або неповну форму. У разі відсутності відповідей на обов'язкові питання система повинна відобразити повідомлення про помилку із зазначенням, які саме дані потрібно доповнити. Відправлення форми в такому випадку є недоступним.

- Користувач успішно заповнив анкету. Система повинна зберегти дані у базі даних (`user_preferences`, `user_favorite_product_types`) та автоматично згенерувати рекомендаційний список кондитерських магазинів, що відповідають введеним критеріям. Після цього користувача перенаправляє на сторінку результатів рекомендацій.

Очікувана швидкодія:

- 1–2 секунди для завантаження анкети;
- 1–3 секунди для генерації рекомендацій (залежить від стабільності інтернет-з'єднання та кількості доступних закладів).

Зауваження: Функціонал повинен коректно працювати як при першому, так і при повторному проходженні анкети, забезпечуючи відображення збережених відповідей та можливість їх змінювати.

Друга функціональна можливість системи – відображення списку рекомендованих кондитерських магазинів.

Очікувана поведінка: Після успішного проходження анкетування або після зміни попередніх відповідей у профілі користувача система повинна автоматично сформувати та відобразити список рекомендованих кондитерських магазинів. Список має формуватися на основі заданих користувачем критеріїв,

а також результатів роботи гібридного рекомендаційного алгоритму. На сторінці рекомендацій користувачу повинні відображатися назва закладу, коротка інформація про нього, рейтинг, цінова категорія та фото.

Можливі наступні випадки:

- Користувач успішно заповнив анкету. У цьому випадку система повинна відобразити сформований персоналізований список рекомендованих кондитерських магазинів, відсортований за рівнем релевантності.

- Користувач не проходив анкетування. У такому випадку система повинна відобразити список популярних кондитерських магазинів, сформований на основі середнього рейтингу та кількості відгуків.

- За заданими критеріями не знайдено жодного закладу. У цьому випадку система повинна відобразити відповідне інформаційне повідомлення з рекомендацією змінити параметри анкетування.

- Користувач повторно змінює анкетні дані. Система повинна автоматично оновити список рекомендацій відповідно до нових параметрів користувача.

Очікувана швидкодія: 1–3 секунди (залежно від кількості записів у базі даних та стабільності інтернет-з'єднання).

Зауваження: Список рекомендованих закладів повинен оновлюватися динамічно без потреби повторної авторизації користувача.

Третя функціональна можливість системи – відображення детальної інформації про кондитерський магазин.

Очікувана поведінка: При виборі користувачем конкретного кондитерського магазину зі списку рекомендацій або зі списку всіх доступних закладів система повинна відобразити персоналізовану сторінку детальної інформації про обраний магазин. На цій сторінці користувачу повинні бути доступні повні відомості про заклад, зокрема адреса розташування, контактні дані, фотогалерея, перелік типів продукції, середній рейтинг та відгуки інших користувачів.

Можливі наступні випадки:

- Користувач переходить на сторінку магазину зі списку рекомендацій. У цьому випадку система повинна відобразити повну інформацію про обраний кондитерський магазин та зафіксувати дану дію як поведінкову активність користувача.

- Користувач переходить на сторінку магазину зі сторінки всіх закладів. Система повинна відобразити аналогічну інформацію без урахування персоналізованого контексту.

- Інформація про магазин тимчасово недоступна. У такому випадку система повинна відобразити відповідне повідомлення про помилку або відсутність даних.

- Користувач переглядає відгуки інших користувачів. Система повинна забезпечити коректне відображення списку відгуків із зазначенням оцінки, тексту коментаря та дати публікації.

Очікувана швидкодія: 1–2 секунди для завантаження основної інформації; до 3 секунд для повного завантаження фотогалереї та відгуків.

Зауваження: Перехід на сторінку магазину повинен реєструватися в системі як елемент поведінкової активності користувача з метою подальшого уточнення рекомендацій.

ВИСНОВОК

У ході виконання магістерської кваліфікаційної роботи розглянуто та проаналізовано предметну область розробки рекомендаційних систем у сфері кондитерських закладів. Досліджено сучасний стан розвитку рекомендаційних технологій, основні підходи до їх побудови, а також особливості застосування таких систем у сфері кондитерських магазинів.

У процесі виконання роботи досягнуто поставленої мети, а саме – розроблено веб-орієнтовану рекомендаційну систему для підбору кондитерських магазинів на основі гібридного методу фільтрації. Для досягнення цієї мети послідовно виконано аналіз предметної області, проведено огляд існуючих програмних аналогів та визначено їх переваги й недоліки.

Розглянуто основні підходи до побудови рекомендаційних систем, зокрема контентну, колаборативну, знаннєву та поведінкову фільтрацію. На основі проведеного аналізу обґрунтовано доцільність використання гібридного підходу, який поєднує переваги кількох методів і дозволяє підвищити точність та стабільність рекомендацій.

У межах кваліфікаційної роботи сформульовано функціональні та нефункціональні вимоги до інформаційної системи. Для формалізації логіки функціонування системи побудовано діаграми IDEF0, що дозволили описати основні процеси на високому рівні абстракції. На основі вимог виконано проектування архітектури системи та логічної і фізичної моделей бази даних, які забезпечують повну підтримку анкетування користувача, збереження його активності, відгуків та результатів рекомендацій.

Розроблено веб-додаток рекомендаційної системи, який забезпечує можливість реєстрації та авторизації користувачів, у тому числі з використанням Google Auth API, проходження анкетування для визначення індивідуальних уподобань, формування персоналізованого списку рекомендованих кондитерських магазинів, перегляд детальної інформації про заклади та керування власним профілем.

Розроблена система забезпечує персоналізований підбір закладів, враховує анкетні дані користувачів, їх поведінкову активність та відгуки. Це відповідає сучасним підходам до побудови інтелектуальних рекомендаційних сервісів у сфері електронної комерції та HoReCa [21], [26].

Особливу увагу в роботі приділено реалізації гібридного алгоритму формування рекомендацій, який базується на поєднанні анкетних даних користувачів, аналізу їх поведінкової активності та відгуків інших користувачів. Це дозволяє забезпечити більш точний та адаптивний підбір закладів відповідно до поточних потреб користувачів.

Для перевірки працездатності розробленої системи та відповідності її функціонування поставленим вимогам виконано ручне тестування основних функціональних можливостей, зокрема процесів реєстрації, авторизації, анкетування, відображення рекомендацій, перегляду детальної інформації про заклади. Результати тестування підтвердили коректність роботи системи та її придатність до практичного використання.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Ricci F., Rokach L., Shapira B. Посібник з рекомендаційних систем. 3-тє вид. Нью-Йорк: Springer, 2021. 1015 с.
2. Liu X. та ін. Enhancing user intent capture in session-based recommendation with attribute patterns. Amazon Science, 2023. URL: <https://www.amazon.science/publications/enhancing-user-intent-capture> (дата звернення: 27.11.2025).
3. Netflix Technology Blog. Foundation Model for Personalized Recommendation. 21.03.2025. URL: <https://netflixtechblog.com/foundation-model-recommendation> (дата звернення: 27.11.2025).
4. Spotify Engineering. Lessons learned from Algorithmic Impact Assessments in practice. 29.09.2022. URL: <https://engineering.atspotify.com/2022/09/algorithmic-impact-assessments/> (дата звернення: 27.11.2025).
5. Liu A. та ін. How to Train Your YouTube Recommender to Avoid Unwanted Content Using Dislike Signals. ICWSM, 2024. URL: <https://arxiv.org/abs/2404.01245> (дата звернення: 27.11.2025).
6. LinkedIn Engineering. Building a Large-Scale Recommendation System: People You May Know. 06.02.2024. URL: <https://engineering.linkedin.com/blog/2024/people-you-may-know> (дата звернення: 27.11.2025).
7. Wang Z., Pan L., Lim E. A survey on point-of-interest recommendations leveraging heterogeneous data (2021–2023). Financial Innovation, 2025. URL: <https://doi.org/10.1186/s40854-025-00729-z> (дата звернення: 27.11.2025).
8. Lim J., Kang S., Kang J. POI Recommendation Scheme Based on User Activity Level and Location Characteristics. Applied Sciences, vol. 14, no. 23, 2024. URL: <https://doi.org/10.3390/app142311726> (дата звернення: 27.11.2025).
9. Zeng J., Zhao Y., Li J. LGSA: A next POI prediction method by using local and global interest. Expert Systems with Applications, 2023. URL: <https://doi.org/10.1016/j.eswa.2023.121164> (дата звернення: 27.11.2025).

10. Alfaifi Y., Garg M., Al-Mharmah H. Recommender Systems Applications: Data Sources and Challenges. *Information*, vol. 15, no. 10, 2024. URL: <https://www.mdpi.com/2078-2489/15/10/660> (дата звернення: 27.11.2025).
11. Li Y. та ін. Recent Developments in Recommender Systems: A Survey. arXiv preprint, arXiv:2306.12680, 2023. URL: <https://arxiv.org/abs/2306.12680> (дата звернення: 27.11.2025).
12. Zhou H. та ін. A Comprehensive Survey of Recommender Systems. *Applied Sciences*, vol. 13, no. 20, 2023. URL: <https://www.mdpi.com/2076-3417/13/20/11378> (дата звернення: 27.11.2025).
13. Bodduluri K. Exploring the Landscape of Hybrid Recommendation Systems. DIVA Portal Report, 2024. URL: <https://www.diva-portal.org/smash/get/diva2:1843233/FULLTEXT01.pdf> (дата звернення: 27.11.2025).
14. Butmeh H. та ін. Hybrid attribute-based recommender system for personalized learning. *Frontiers in Computer Science*, 2024. URL: <https://www.frontiersin.org/articles/10.3389/fcomp.2024.1404391> (дата звернення: 27.11.2025).
15. Rossiiev O. D. та ін. A comprehensive survey on reinforcement learning-based recommender systems. *CEUR Workshop Proceedings*, vol. 3917, 2025. URL: <https://ceur-ws.org/Vol-3917/paper62.pdf> (дата звернення: 27.11.2025).
16. Aljunid M. F., Manjaiah M. D. H., Kazim M. A collaborative filtering recommender systems: Survey. *Neurocomputing*, 2025. URL: <https://doi.org/10.1016/j.neucom.2024.128718> (дата звернення: 27.11.2025).
17. Zhao X. та ін. Embedding in Recommender Systems: A Survey. arXiv, arXiv:2310.18608, 2023. URL: <https://arxiv.org/abs/2310.18608> (дата звернення: 27.11.2025).
18. IBM Research. What is content-based filtering? IBM Blog, 2024. URL: <https://www.ibm.com/think/topics/content-based-filtering> (дата звернення: 27.11.2025).

19. Recommendation systems techniques based on model-based methods: an overview. ММС, vol. 11, no. 4, 2024. URL: <https://science.lpnu.ua/mmc/all-volumes-and-issues/volume-11-number-4-2024/recommendation-systems-techniques-based> (дата звернення: 27.11.2025).

20. Bodduluri K. C., Palma F., Kurti A., Jusufi I., Löwenadler H. Exploring the Landscape of Hybrid Recommendation Systems in E-Commerce: A Systematic Literature Review. IEEE Access, 2024.

21. Sami A., et al. A deep learning based hybrid recommendation model for Internet Users (HRS-IU-DL). Scientific Reports, 2024.

22. Sabiri B., et al. Hybrid Quality-Based Recommender Systems. Visual Computing for Industry, Biomedicine, and Art, 2025.

23. Elahi M., et al. Hybrid recommendation by incorporating the sentiment of reviews. Information Sciences, 2023.

24. Biswas P. K., et al. A hybrid recommender system combining ALS and deep learning to overcome cold-start problem. Expert Systems with Applications, 2022.

25. Maruyama M. H., et al. Hybrid Recommender System for Educational Resources. SCITEPRESS, 2023.

26. Valencia-Arias A., et al. Artificial intelligence and recommender systems in e-commerce: a bibliometric study. Journal of Innovation & Knowledge, 2024.

27. Chelvam N. L. N. M. S., et al. Hybrid-Based Recommender System Based on Electronic Reviews. Journal of Informatics and Virtual Education, 2025.

28. Kowald D., et al. Reviews in recommender systems: Recent advances and future challenges. Frontiers in Artificial Intelligence, 2024.

29. Gheewala S., et al. Deep learning in modern recommender systems: An in-depth survey. Neural Computing & Applications, 2025.

30. Степанюк О. В., та ін. Інформаційна рекомендаційна система на основі застосування асоціативних правил у бізнесі. Економічний вісник, 2024. (Українське джерело)

31. Hazar M. J., et al. A Recommendation System Involving a Hybrid Approach of Reviews and Latent Factor Modeling. *Journal of the Brazilian Computer Society*, 2023.

32. Tholib A., et al. An Intelligent Recommendation System Utilizing a Hybrid Deep Learning and Probabilistic Matrix Factorization Approach (HD-PMF). *Engineering, Technology & Applied Science Research*, 2025.

33. Kozulia M. M., Sushko V. V. Hybrid filtering system for rating prediction. *MicroCAD*, 2021. (Українське наукове середовище)

34. Chelvam N. L. N. M. S. Hybrid Recommendation Based on Electronic Reviews and User Behavior. *International Journal of Intelligent Systems*, 2025.

35. Тітов С. В., Тітова О. В., Чорна О. С. Метод знаходження апроксимацій приблизних множин з використанням систем числення. *Системи обробки інформації*. 2023. № 2 (173). С. 58-62. <https://doi.org/10.30748/soi.2023.173.07>

36. Ситніков Д.Е., Ситнікова П.Е., Тітов С.В., Тітова О.В. Фільтрація результуючого набору асоціативних правил з точки зору оцінки цікавості / Д.Е. Ситніков, П.Е. Ситнікова, С.В. Тітов, О.В. Тітова // *Системи обробки інформації* № 1(164) 2021, стр. 83-88. <https://doi.org/10.30748/soi.2021.164.09>

37. Методи рекомендаційних систем для вибору кондитерських магазинів / Виноградов М.Ю., Тітов С.В. // *Інформаційні технології і автоматизація – 2025* / *Матеріали XVIII міжнародної науково-практичної конференції*. Одеса, 30-31 жовтня 2025 р. - Одеса, Видавництво ОНТУ, 2025 р. – Ст. 681-682.