004.89:004.415.2:641.55

ПРОЄКТУВАННЯ СИСТЕМИ ПЕРСОНАЛЬНИХ РЕКОМЕНДАЦІЙ РЕЦЕПТІВ

Гратілова К.О.[1]; Тройніна А.С., к.т.н., доц.[2]
[1,2] Національний університет «Одеська політехніка»
[1,2] Україна, Одеса
[1] 8042288@stud.op.edu.ua; [2] atroinina@op.edu.ua

Анотація. *Спроєктовано вебзастосунок персональних рекомендацій рецептів на основі збережених рецептів, наявних продуктів та історії оцінок. Подано вимоги, аналіз ризиків, архітектуру, структуру бази даних, компоненти та підхід до реалізації й оцінювання моделі машинного навчання. Результати можуть бути використані для створення кулінарних застосунків.*
Ключові слова: *рекомендаційна система; машинне навчання; персоналізація; вебзастосунок; Firebase.*

Вступна частина. Питання «що приготувати сьогодні?» щодня постає перед мільйонами людей: знайдений рецепт нерідко вимагає інгредієнтів, яких удома немає, а інформація про приготовані й вподобані страви зазвичай не фіксується. Рекомендації у сфері харчування виділилися в окремий напрям досліджень – food recommender systems [1]. Їхня специфіка полягає в складній структурі рецепта та залежності оцінки від наявних продуктів.

Традиційно застосовують колаборативну фільтрацію, фільтрацію за змістом та гібридні методи [2]. Для рецептів колаборативна фільтрація стикається з розрідженістю даних, адже щоб оцінити страву, її потрібно приготувати; проблемою є також «холодний старт» і різні написання однакових інгредієнтів. Аналіз сервісів Samsung Food, Supercook та Allrecipes показав, що вони розв’язують лише окремі частини задачі: жоден не поєднує збережені рецепти, наявні продукти та історію оцінок для створення рекомендацій.

Мета роботи. Проєктування вебзастосунку персональних рекомендацій рецептів для вибору страв за вподобаннями користувача, наявними продуктами та дієтичними обмеженнями шляхом визначення вимог, аналізу ризиків, розроблення архітектури, структури бази даних і компонентів, включно з модулем машинного навчання та методикою оцінювання його якості.

Основна частина. Система підтримує дві ролі: гість та авторизований користувач. Функціональні вимоги охоплюють реєстрацію, авторизацію, редагування особистого кабінету, додавання та видалення рецептів з обраних, керування доступними інгредієнтами, перегляд рекомендованих рецептів та оцінювання приготованої страви. Нефункціональні вимоги задано вимірюваними показниками: відповідь на запит рекомендацій до 2 с для 95 % запитів за активності до 200 користувачів; доступність даних не менше 99,9 %; Precision@10 $\geq$ 0,6 та Recall@10 $\geq$ 0,4 для 80 % активних користувачів; виключення рецептів з алергенами у 100 % випадків; перенавчання моделі за падіння якості нижче порогів.

Під час аналізу було виявлено ризик отримання нерелевантних або небезпечних рекомендацій. Він має кілька аспектів: різні написання однакових інгредієнтів у рецептах і переліку продуктів; «холодний старт» і нестача оцінок для навчання моделі; помилкові правила доступу до Firestore; можливість рекомендації страви з алергеном. Для його зниження передбачено словник синонімів із нормалізацією назв, використання популярності

рецептів і вказаних кухонь для нових користувачів, кешування результатів, окремий шар доступу до Firebase та жорстке виключення алергенів і дієт до формування списку [3].

Систему побудовано за багаторівневою клієнт-серверною архітектурою (рис. 1): клієнтський вебзастосунок, серверне середовище (REST API та рекомендаційний модуль) і хмарні сервіси Firebase [4]. Це забезпечує централізоване зберігання даних, винесення ресурсомістких обчислень на сервер і оновлення моделі без змін на стороні користувачів. Автентифікацію виконує Firebase Authentication, дані зберігаються в Cloud Firestore, а бізнес-логіка звертається до бази виключно через шар Repository, що ізолює залежність від постачальника сховища.

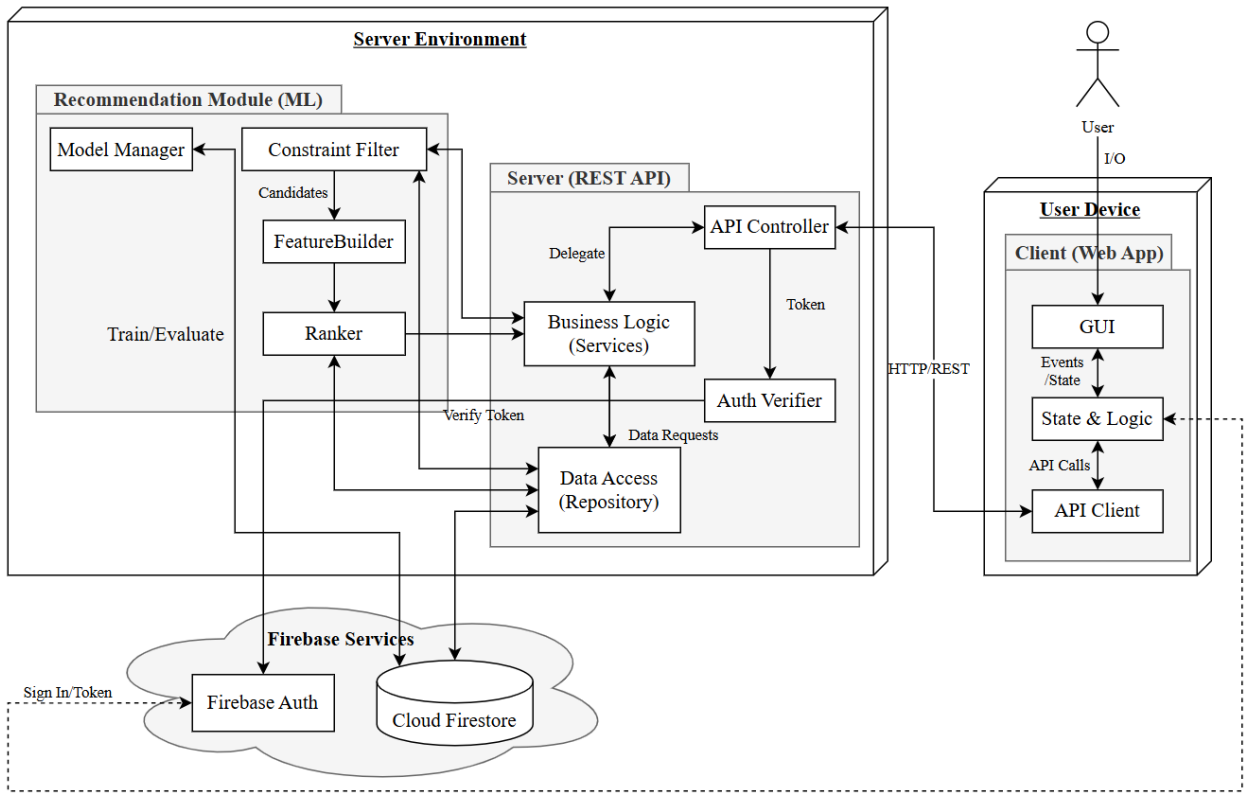


Рисунок 1 – Архітектурна модель системи

Рекомендаційний модуль працює як конвеєр: Constraint Filter відсіює рецепти з алергенами та несумісні з дієтами; Feature Builder будує вектор вподобань; допоміжний компонент Pantry Matcher обчислює частку наявних інгредієнтів; Ranker впорядковує список за загальною оцінкою, а Model Manager відповідає за навчання й оцінювання моделі. Результат кешується й позначається застарілим після зміни оцінок, обраних чи продуктів; якщо модуль недоступний, показується останній збережений список або популярні рецепти.

Структура бази даних включає наступні ключові сутності: Rating (оцінки 1–5 як основне джерело навчальних даних), User_Ingredient (наявні продукти), Recipe_Ingredient (склад рецепта), Ingredient_Synonym (нормалізація назв), довідники Allergen, Diet і Cuisine, Recommendation (кеш результатів) та Model_Version (версії моделі з метриками).

Рекомендаційний модуль реалізується мовою Python із бібліотекою scikit-learn. Згідно з оглядом [5], більшість систем рекомендації їжі спирається на фільтрацію за змістом і машинне навчання; такий підхід обрано й тут через розрідженість оцінок. Рецепт подається вектором ознак: інгредієнти, кухня й дієти кодуються бінарно, теги – через TF-IDF (міру важливості слова в тексті), час приготування та складність масштабуються до діапазону від 0

до 1. Вектор-профіль користувача будується з оцінених ним рецептів: чим вища оцінка, тим більший їхній вплив, а обрані дають слабший. Відповідність смакам (taste_score) – це косинусна схожість вектора рецепта з профілем, а загальна оцінка – зважена сума taste_score і частки наявних інгредієнтів (ingredient_match) із вагами, підібраними на перевірочній вибірці. Для нових користувачів використовуються вказані кухні та популярність рецептів; нова версія моделі стає активною, лише якщо не гірша за поточну, а перенавчання запускається за падіння якості або за розкладом.

Якість оцінюється офлайн на відкладеній вибірці, що відповідає поширеній практиці [5]. Основні метрики: Precision@10 і Recall@10, NDCG@10 (враховує позицію рецепта у списку). Предметні метрики: частка рецептів у десятці з наявними інгредієнтами від 60 % (ціль – не менше 70 % рецептів) та частка порушень алергенів і дієт (ціль – 0 %). Результати порівнюють із рекомендаціями за популярністю та за збігом із продуктами. Оцінки користувача за датою ділять на навчальну, перевірочну й тестову вибірки (70/15/15); активними вважаються користувачі, у яких тестова вибірка містить не менше 15 релевантних рецептів (оцінка 4–5), що робить досяжним Precision@10 $\geq$ 0,6, а тих, хто має менше 10 оцінок, оцінюють окремо як «холодний старт».

Висновки. Спроєктовано вебзастосунок персональних рекомендацій рецептів, який поєднує збережені рецепти, наявні продукти та історію оцінок. Визначено функції системи й вимірювані нефункціональні вимоги, виявлено ризик нерелевантних або небезпечних рекомендацій та заходи щодо його зниження, запропоновано клієнт-серверну архітектуру з окремим рекомендаційним модулем, структуру бази даних і компоненти. Описано підхід до реалізації моделі, показники якості та вибірки для тестування. Отримані рішення, зокрема досягнення цільових значень, буде перевірено під час створення прототипу.

Використання інструментів ШІ. Під час підготовки публікації використано мовну модель Claude Sonnet 5.5 (Anthropic) для покращення академічного викладу у вступній, основній частинах і висновках та перекладу англомовної анотації (розділ «Abstract»).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] Wu, Y., Kang, L., Guo, Q., & Zhao, L. (2025). A review of recipe recommendation methods. *IEEE Access, 13*, 92483–92494. https://www.researchgate.net/publication/391974878

[2] Ibrahim, O. A. S., Younis, E. M. G., Mohamed, E. A., & Ismail, W. N. (2025). Revisiting recommender systems: An investigative survey. *Neural Computing and Applications, 37*(4), 2145–2173. https://doi.org/10.1007/s00521-024-10828-5

[3] Fu, Y., Cai, L., Wu, R., Kang, Y., & Zhao, Y. (2026). Nutrient-aware personalized meal recommendation using structured food knowledge and constraint verification. *Foods*, *15*(10), Article 1647. https://doi.org/10.3390/foods15101647

[4] Google. (n.d.). *Firebase documentation*. Retrieved November 7, 2026, from https://firebase.google.com/docs

[5] Bondevik, J. N., Bennin, K. E., Babur, Ö., & Ersch, C. (2024). A systematic review on food recommender systems. *Expert Systems with Applications*, *238*(Part E), Article 122166. https://www.sciencedirect.com/science/article/pii/S0957417423026684

Hratilova K.O., Troinina A.S.

Design of a personal recipe recommendation system

Abstract. *A web application for personalized recipe recommendations has been designed based on saved recipes, available ingredients, and rating history. Requirements, risk analysis, architecture, database structure, components and an approach to implementing and evaluating the machine learning model are described. The results can be used to develop cooking applications.*

Keywords: *recommendation system; machine learning; personalization; web application; Firebase.*