

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Харківський національний університет радіоелектроніки
Факультет Комп'ютерних наук
Кафедра Програмної інженерії

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

другий (магістерський)
(рівень вищої освіти)

Дослідження алгоритмів рекомендаційних систем для
здорового харчування

Виконав:
студент 2 курсу групи ІІЗМ-21-1
Чмуж Г. С.
(прізвище, ініціали)

Спеціальність 121 – Інженерія програмного
забезпечення

Тип програми Освітньо-наукова

Керівник доц. Голян Н. В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. Кафедри _____

З.В. Дудар

2023 р.

Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____
Кафедра _____ Програмної інженерії _____
Рівень вищої освіти _____ другий (магістерський) _____
Спеціальність _____ 121 – Інженерія програмного забезпечення _____
(код і повна назва)
Тип програми _____ освітньо-наукова програма _____
Освітня програма _____ Інженерія програмного забезпечення _____

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

«___» _____ 2023 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студента _____ Чмужу Геннадію Сергійовичу _____
(прізвище, ім'я, по-батькові)

1. Тема роботи «Дослідження алгоритмів рекомендаційних систем для
здорового харчування» _____

затверджена наказом університету від « 29 » 03.2023 р. № 302Ст

2. Термін подання студентом роботи до екзаменаційної комісії
«19» травня 2023 р.

3. Вихідні дані до роботи: електронні ресурси за обраною тематикою,
рекомендаційні алгоритми для здорового харчування, методичні вказівки до
виконання кваліфікаційної роботи магістра, пояснювальна записка.

4. Перелік питань, що потрібно опрацювати в роботі: аналіз предметної
області, постановка задачі, огляд існуючих рекомендаційних алгоритмів, опис
розробленої рекомендаційної системи для здорового харчування.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз предметної галузі	25.02.2023	виконано
2	Постановка задачі	10.03.2023	виконано
3	Проведення дослідження	01.04.2023	виконано
4	Підготовка пояснювальної записки	01.05.2023	виконано
5	Підготовка презентації та доповіді	08.05.2023	виконано
6	Попередній захист	11.05.2023	виконано
7	Перевірка на академічний плагіат	11.05.2023	виконано
8	Нормоконтроль	13.05.2023	виконано
9	Рецензування	16.05.2023	виконано
10	Знесення диплома в електронний архів	16.05.2023	виконано
11	Допуск до захисту у зав. Кафедри	22.05.2023	виконано

Дата видачі завдання 29 березня 2023 р.

Студент _____

(підпис)

Керівник роботи _____

(підпис)

доц. Голян Н. В.

(посада, прізвище, ініціали)

РЕФЕРАТ / ABSTRACT

Кваліфікаційна робота магістра містить: 70 с., 16 рис., 9 табл, 10 джерел, 5 додатків.

АЛГОРИТМ, ДОПУСТИМІ НОРМИ, ЗДОРОВ'Я, ЇЖА,
РЕКОМЕНДАЦІЙНА СИСТЕМА, РЕКОМЕНДАЦІЇ, ТРАВЛЕННЯ,
ХАРЧУВАННЯ.

Об'єктом дослідження є рекомендаційні системи для підтримки здорового харчування користувачів.

Предметом дослідження є рекомендаційні алгоритми, які застосовуються в системах здорового харчування.

Метою даної роботи є аналіз, дослідження та вдосконалення рекомендаційних алгоритмів в системах здорового харчування.

Методи дослідження, що були застосовані: методи порівняння, синтез та аналіз, математичні методи.

Результатом виконаної роботи є створення вдосконаленого рекомендаційного алгоритму для систем здорового харчування, який зможе розширити можливості існуючих алгоритмів.

ALGORITHM, DIGESTION, FOOD, HEALTH, NUTRITION, PERMISSIBLE
NORMS, RECOMMENDATION, RECOMMENDATIONS SYSTEM.

The object of the study is recommendation systems to support healthy nutrition of users.

The subject of research is recommendation algorithms used in healthy food systems.

The purpose of this work is the analysis, research and improvement of recommendation algorithms in healthy food systems.

Research methods that were applied: methods of comparison, analysis and synthesis, mathematical methods.

The result of the performed work is the creation of an improved recommendation algorithm for healthy food systems, which will be able to expand the capabilities of existing algorithms.

Умови публікації пояснювальної записки

Я,

Чмуж Геннадій Сергійович

(прізвище, ім'я, по батькові)

студент(ка) групи ІІЗМ-21-1 здобувач вищої освіти на другому (магістерському) рівні

кафедра програмної інженерії,

(повна назва кафедри)

заявляю: моя кваліфікаційна робота на тему

Дослідження алгоритмів рекомендаційних систем для здорового харчування,

(назва роботи)

що буде представлена до ЕК для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу EIArKhNURE. Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлений (а) з діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

ЗМІСТ

Вступ.....	7
1 Аналіз предметної галузі.....	10
1.1 Огляд стану розв'язання проблеми.....	10
1.2 Виявлення проблем та актуалізація рішень	17
1.3 Постановка задачі.....	18
2 Проведення дослідження.....	20
2.1 Актуальність та мета.....	20
2.2 Методи дослідження.....	21
2.3 Методологія теоретичних досліджень, математична модель.....	23
2.4 Методологія та проведення експерименту	26
2.5 Вибір рекомендаційного алгоритму.....	28
2.6 Проведення лінійної згортки	29
2.7 Аналіз програмного забезпечення за допомогою планування експерименту	31
3 Аналіз результатів досліджень	37
3.1 Аналіз	37
3.2 Формування моделі рекомендаційної системи	39
4 Розробка програмної системи	42
4.1 Архітектура та проектування.....	42
4.2 Програмні рішення.....	45
5 Можливості використання отриманих результатів	49
Висновки	51
Перелік джерел посилання.....	53
Додаток А.....	54
Додаток Б	55
Додаток В.....	56
Додаток Г	64
Додаток Д.....	65

ВСТУП

Від якості харчування залежить загальний стан нашого організму. Майже всі чули, що «неправильна» їжа – фастфуди, жирне, смажене – стає причиною надмірної маси тіла. Але у такого раціону насправді набагато більше наслідків для здоров'я, ніж здається на перший погляд. Крім того, важливо не тільки їсти якісну їжу, але й дотримуватись певного графіка її прийому, щоб ми отримували поживні речовини постійно протягом дня. Причин неправильних харчових звичок може бути кілька. Постійний поспіх, стреси, ненормований робочий графік тощо. Але найчастіше неправильне харчування закладається ще у дитинстві. Ми запозичуємо харчові звички ще від батьків, а з віком навіть не замислюємося про те, наскільки взагалі корисне звичне для нас меню. Хоча всім відомо, що неправильне харчування може вплинути на наш організм, але тема шкоди неправильного харчування ні для кого не нова. Особливо вона актуальна в наш час, коли навколо повно так званого фаст-фуду, а також різних напівфабрикатів, переповнених шкідливими речовинами. Звичайно, вся ця їжа може виглядати дуже апетитно і запах у неї може бути дуже спокусливим, проте для шлунка та й для всього організму в цілому вона може бути небезпечною. Неправильне харчування переповнює наш організм токсинами. Воно навантажує його зайвою роботою. Серцю доводиться витратити більше зусиль у тому, щоб качати кров; дихальна система також отримує додаткове навантаження. Такі зміни починають відбуватися не відразу – часом до цього має пройти чимало часу. Однак виявлятися вони починають поступово, і тому спочатку можуть бути непомітні, зате потім вже може бути пізно повернути колишні сили організму [1]. На допомогу приходять різні програмні системи та мобільні додатки, які можуть контролювати раціон користувача та запобігати виникненню даних проблем. Але чи можна їх назвати достатньо точними та надійними?

Харчування впливає на кожний орган тіла людини, тож дотримання правильного раціону є запорукою здоров'я. Неправильне харчування навпаки призводить до різноманітних та тяжких проблем, які складно помітити на початку та ще складніше позбавитись від них, коли вони вже розвинулись [2]. Наявність

різних рекомендаційних систем значно зменшує кількість та серйозність проблем, які виникають через харчування.

Для того щоб людині контролювати свій раціон та різні норми з правилами, їй необхідно знати всі ці норми, яких достатньо багато, а також вираховувати для себе кожен раз перед прийомом їжі. Звичайно що потрібно перенести всю цю логіку до електронного пристрою, який буде швидко робити це все сам та без помилок. Але проблема в тому, що існує занадто багато чинників, які потрібно врахувати, для того щоб не нашкодити здоров'ю людині [1]. В основі таких обчислень лежить алгоритм, який приймає на вхід всі ці чинники та визначає, як людині краще харчуватися.

Метою даного дослідження є розробка програмного алгоритму, який дозволить враховувати найрізноманітніші фактори стосовно будь-якої людини або продукту харчування та надасть надійні рекомендації для цієї людини щодо її здорового харчування. Для того, щоб досягти мету, необхідно виконати наступні задачі:

- проаналізувати програмні системи та додатки, які вже існують на ринку для вирішення проблем, пов'язаних зі здоровим харчуванням, а також розглянути, яку інформацію вони використовують в своїх алгоритмах, для того, щоб допомогти користувачу;
- розробити алгоритм, який буде компенсувати недоліки існуючих алгоритмів, а також матиме унікальну особливість, яка буде відсутня в інших;
- порівняти розроблений алгоритм з існуючими.

Об'єктом дослідження є рекомендаційні системи для підтримки здорового харчування користувачів.

Предметом дослідження є рекомендаційні алгоритми, які застосовуються в системах здорового харчування.

Для досягнення поставленої у даній роботі мети був використаний наступний перелік методів дослідження:

- методи порівняння;

- методи синтезу та аналізу;
- математичні методи.

Методи синтезу та аналізу необхідні, щоб проаналізувати властивості, а також особливості рекомендаційних алгоритмів, що вже існують, а також розробки алгоритму рекомендацій щодо здорового вживання продуктів харчування. Для виявлення найбільш ефективних серед всіх алгоритмів були використані методи порівняння.

Відмінність одержаних результатів від аналогічно відомих раніше полягає в наявності нових додаткових, раніше не використовуваних, величин, які роблять рекомендації більш точними. Ступенем новизни є удосконалення та отримання подальшого розвитку.

Практичне значення одержаних результатів полягає в незалежності алгоритму від специфічної технології або мови програмування, тому його можливість реалізації дає найбільш гнучкий спектр використання. Він може бути прописаний на будь-якій мові програмування, включаючи його часткову оптимізацію у відповідності з обраною технологією, та бути запущений на будь-якому пристрою, починаючи від серверу та фронтенду до смартфона або смарт годинника. Таким чином даний алгоритм можна застосувати в будь-якій рекомендаційній системі для здорового харчування.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Огляд стану розв'язання проблеми

Задоволення голоду один із найважливіших інстинктів організму, необхідний для збереження життя. По-перше, всі клітини та тканини нашого організму формуються з тієї їжі, яку ми їмо. По-друге, їжа є джерелом енергії, яка потрібна на функціонування організму. Отже, від того, що ми їмо, в якій кількості, коли і яким чином, залежить наше здоров'я.

Здорове харчування дає можливість стабілізувати вагу без насильницьких обмежень, допомагає позбутися захворювань і запобігти їх розвитку, сприяє відновленню інтелектуальної та фізичної енергії. Здорове харчування – важлива складова здорового способу життя [2].

Правильне харчування передбачає, що в організм регулярно, у необхідній кількості та оптимальних співвідношеннях повинні надходити багато поживних речовин – білки, вуглеводи, жири, вода, мінеральні речовини та вітаміни. Недолік чи надлишок деяких поживних елементів стають причинами спочатку тимчасових незручностей, а потім і ризику розвитку різних захворювань. Щоб правильно харчуватися, потрібне не лише бажання, а й знання. Їжа потрібна для життя. Будь-яка їжа – це не просто задоволення чи ритуал. Будь-яка з'їдена страва постачає в організм енергію та речовини для підтримки здоров'я [2].

Існує безліч правил щодо прийому їжі. Співвідношення білків, жирів та вуглеводів за день має приблизно дорівнювати 1:1:4. Запам'ятати правильні пропорції можна за допомогою простого прийому: уявіть тарілку, розділену на три однакові частини. Дві з них займають вуглеводи, а третю порівну ділять білки та жири. Одні й ті самі продукти на столі – це нудно і загрожує недобором важливих поживних компонентів. Отримати необхідний набір вітамінів, мікроелементів та мінеральних речовин можна лише з довгого списку продуктів, більшість з яких цілком доступні: овочі, фрукти, м'ясо, риба, горіхи, молочні продукти, бобові, хліб, макарони із твердих сортів пшениці, ягоди та зелень. Як не парадоксально, щоб не набирати вагу, треба їсти частіше. Таким чином ви не відчуватимете почуття голоду і зможете контролювати якість і кількість їжі, що

приймається. Вуглеводи життєво важливі. Це основне джерело енергії, оскільки організм використовує для потреб саме глюкозу. Вуглеводи легко перетравлюються і швидко надходять у кров. При нестачі вуглеводів тіло починає черпати енергію з білків, які у м'язах, тим самим позбавляючи організм будівельного матеріалу для життєво важливих гормонів і антитіл. Вуглеводи прості та повільні. У розумінні обивателя вуглеводна їжа – це щось солодке. Але продукти, що містять дуже багато цукру – спосіб забезпечення організму енергією, якої в результаті вистачає дуже ненадовго, і швидко виникає відчуття голоду. Тому краще їсти складні або, як їх ще називають, повільні вуглеводи, які засвоюватимуться поступово: овочі, фрукти, продукти з цільного зерна, макаронні вироби з твердих сортів пшениці та крупи. Менше цукру. Надмірний цукор у харчуванні – ще одна причина зайвої ваги [1].

Цей перелік рекомендацій можна продовжувати ще довго. На допомогу приходять різноманітні програмні додатки. У мережі є дуже багато різних програм для здоров'я та фітнесу: безкоштовні, платні, корисні та не дуже. Через це досить складно зрозуміти, з чого краще розпочати і які з них справді варті уваги. Перш ніж щось завантажувати, важливо визначитись із цілями. Є додатки, які допомагають стати сильнішими, а такі, що дозволяють бути більш підтягнутим. Деякі надають можливість стежити за калоріями чи графіком харчування або ставлять за мету схуднути.

Також варто виділити декілька додатків, які за штрих кодом надають повний перелік складу даного продукту. Недоліком таких додатків є те, що вони мають невелику базу, де зберігаються занесена інформація про продукти, тому користувачам доводиться власноруч вносити інформацію про продукти, яких ще не має в базі. Одним з таких додатків є Open Food Facts. Воно дозволяє сканувати штрих-коди харчових продуктів і отримувати дані про їх склад, харчові добавки та харчову цінність. На сьогоднішній день у додатку вже більше 700 000 продуктів, багато з яких додаються користувачами програми (див.рис.1.1).

Також програма дозволяє визначати:

- вуглецевий слід (викиди CO₂) та інструкції з утилізації упаковки;

- поживні речовини, жири, насичені жирні кислоти, вуглеводи, цукру, клітковина, білок і сіль та натрій;
- алергени, а також приналежність продукту: біологічні, безглютенові, веганські, вегетаріанські, халяльні, кошерні і т. п.;
- на винах та пиві – вміст алкоголю.



Рисунок 1.1 – Додаток «Open Food Facts»

Ще є додатки, які можуть виділяти знаки на упаковках та пояснювати користувачу, за що він відповідає. Так Ecolabel Guide – це програма розроблена експертами-екологами на підставі підходу Програми ООН з навколишнього середовища (UNEP). Рекомендовані в додатку екомаркування виділяють якісну продукцію, що відповідає найсуворішим екологічним стандартам, які існують сьогодні. Будь-який знак на упаковці може бути відсканованим та перевіреном, що він означає і чи заслуговує на довіру. Цей додаток дозволяє обирати продукти, які справді відповідають екостандартам. Посібник з екомаркування розроблено, щоб допомогти споживачам зробити екологічний вибір продуктів на основі інформації, що стоїть за маркуванням, і розпізнати екомаркування з інших марок.

Інструмент розпізнавання дозволяє сфотографувати етикетку на упаковці товару та одразу відповідає на запит (див.рис.1.2).

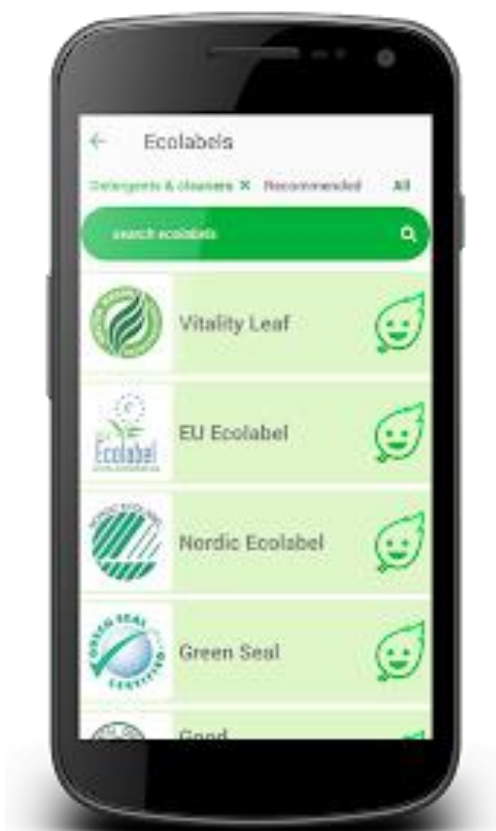


Рисунок 1.2 – Додаток «Ecolabel Guide»

Є додатки довідники. Одним з них є «Е код – харчові добавки». Це довідник харчових добавок із кодом «Е», що працює без інтернету. Каталог містить опис груп добавок, а також виявлені у них шкідливі побічні ефекти для здоров'я людини. У довіднику є повний опис відомих барвників, консервантів, антиоксидантів, стабілізаторів, емульгаторів, підсилювачів смаку, антифлавінів та піногасників. У цьому додатку подано повну таблицю Е-кодів харчових добавок, їх назву, опис, походження, сферу застосування та небезпеку, яку вони містять для людини. За допомогою зручного пошуку можна швидко знайти цікаві добавки як за кодом, так і за назвою. Програма представляє собою оффлайн довідник харчових добавок з кодом «Е». Для класифікації харчових добавок в країнах Євросоюзу розроблена система нумерації (діє з 1953 року). Каталог харчових добавок містить опис групи, а також шкідливі побічні ефекти (якщо такі виявлені) для здоров'я. Відмінною особливістю подібних програм є швидкий з

використанням тільки цифрових кнопок і підказкою можливих місць (див.рис.1.3).

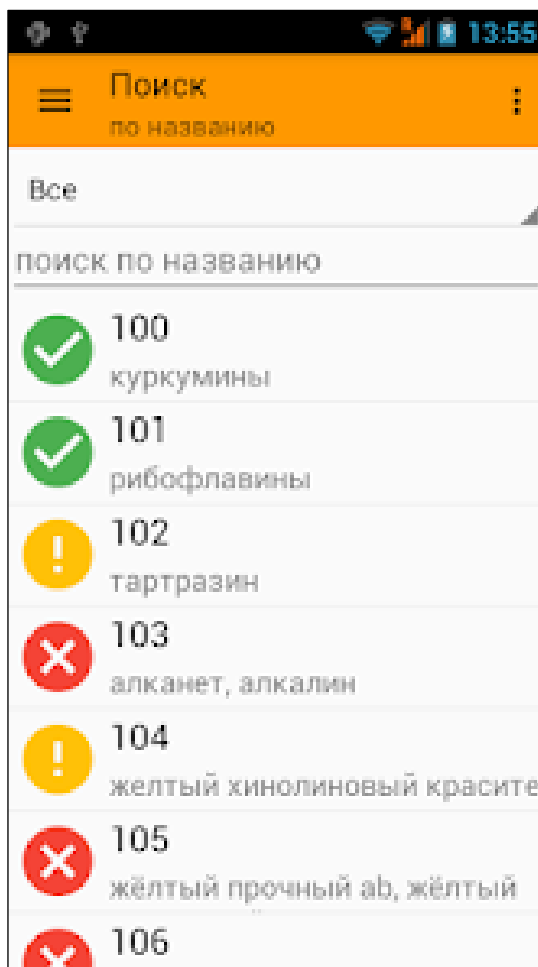


Рисунок 1.3 – Додаток «Е код – харчові добавки»

Наступним прикладом програмного забезпечення в даній області може стати meez. Цей інструмент приготування рецептів, якому довіряють понад 12 000 професіоналів у сфері харчування та напоїв, розуміє внутрішні операції. Він надає можливість отримувати точні лазерні витрати на харчові продукти, автоматично враховувати врожайність, одиниці вимірювання та втрати при підготовці. Також він прискорює адаптацію співробітників і допомагає зберегти послідовність, перетворивши свої рецепти на інтерактивні навчальні матеріали. Це програмне забезпечення дозволяє готувати точну кількість їжі, яка потрібна користувачам, за допомогою автоматичного масштабування та перетворення рецептів. Крім того, можливо синхронізувати всю команду кухарів, маючи центральне місце для всього вмісту ваших рецептів.

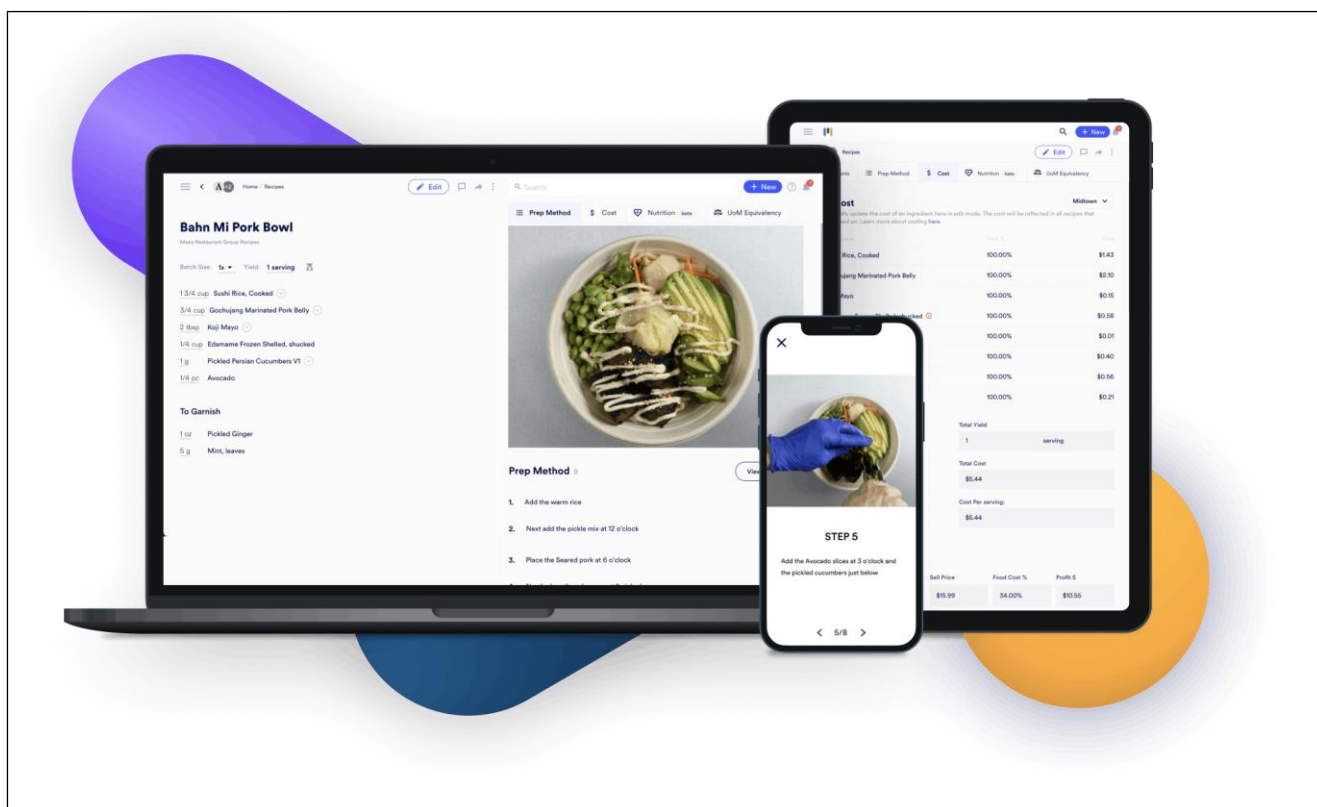


Рисунок 1.4 – Програмне забезпечення «meez»

Ще одним програмним забезпеченням для вирішення проблем з харчуванням може бути «DietMaster Pro». Маючи понад 25 років роботи, DietMaster Pro пропонує найкращу в своєму класі технологію професійного харчування з безмежними можливостями, а також приватні білі етикетки та можливості інтеграції. Їх програмне забезпечення пропонує настільне рішення для Windows і Mac, Інтернет і мобільні рішення. Воно дозволяє роздрукувати плани харчування за лічені хвилини. Надає можливість отримати доступ до їх додаткової бібліотеки готових шаблонів планів харчування, розроблених лікарями та зареєстрованими дієтологами, або розробити власні плани за допомогою їх простого інтерфейсу. Найкраще підходить для повної лінійки професійних консультаційних продуктів щодо харчування, розроблених відповідно до потреб медичної практики, здоров'я та фітнесу, оздоровлення та освітніх ринків.

Професійна серія програмного забезпечення DietMaster пропонує настільне рішення для систем Mac або Windows або веб- та мобільне рішення, яке можна легко додати на ваш веб-сайт за допомогою опцій білої мітки.



Рисунок 1.5 – Програмне забезпечення «DietMaster Pro»

Є програмне забезпечення, яке дозволяє керувати навіть не власним раціоном, а інших людей. Так My PT Hub – це комплексне програмне забезпечення для особистих тренерів, тренерів, власників тренажерних залів і дієтологів, щоб керувати своїми клієнтами, створюючи гнучкі програми тренувань і харчування, одночасно відстежуючи їхній прогрес і досягнення. Тут можливо почати роботу з їх безкоштовної 30-денної пробної версії. Там наявні такі нові маркетингові функції:

- прискорення свого бізнесу за допомогою Google Ads у своєму обліковому записі My PT Hub;
- демонстрація свого бізнесу за допомогою MySite, вашої власної віртуальної вітрини.

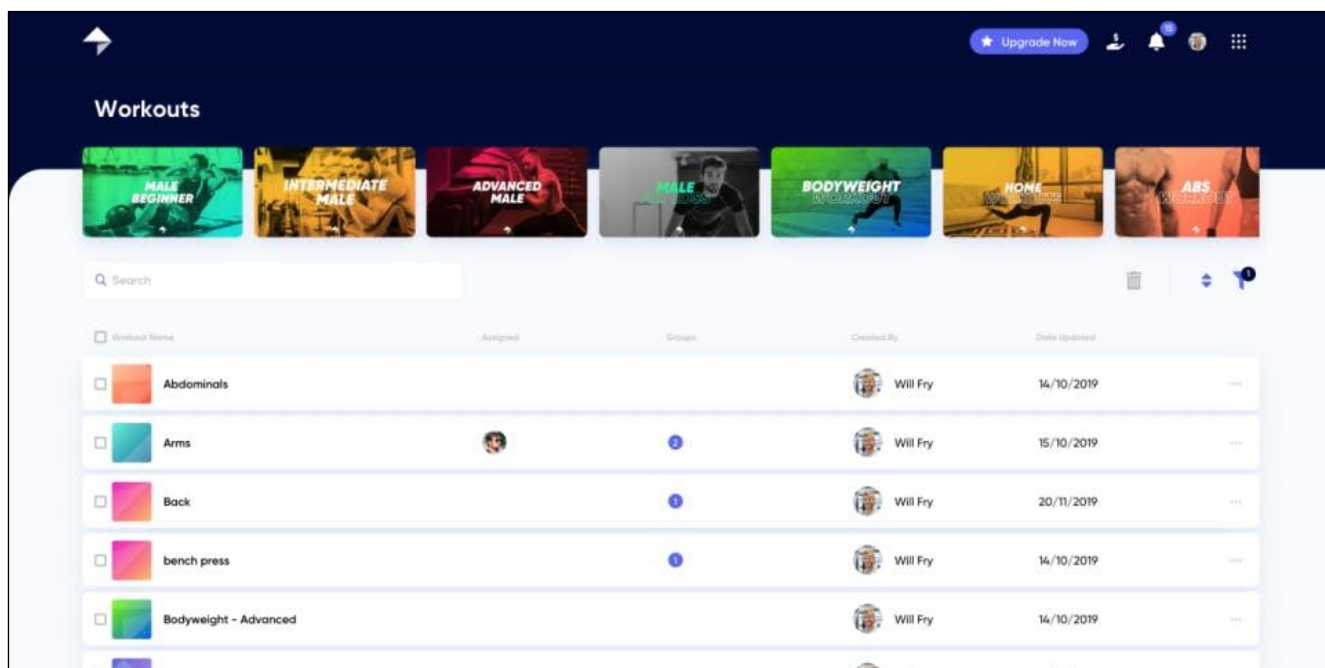


Рисунок 1.6 – Програмне забезпечення «My PT Hub»

Приклади існуючих програмних систем можна розглядати й надалі, але в цілому основи, на яких вони базуються, повторюються, але вже в різних контекстах.

1.2 Виявлення проблем та актуалізація рішень

Насправді існує величезна кількість програмного забезпечення, яке може допомогти з досягненням деякого здорового балансу під час харчування, і всі вони дають певні рекомендації, які є зрозумілими та не абстрактними. Однак використовуються прості базові принципи, і важко сказати, що точність таких рекомендацій досить гарна.

Індивідуальні рекомендації для здорового харчування – достатньо складне завдання, оскільки багато параметрів потрібно враховувати, щоб вибрати оптимальний алгоритм. Тому необхідно враховувати функціональні можливості алгоритмів і вибирати найбільш ефективні. Кожна рекомендаційна система працює не лише з інформацією про їжу та людину, але й з джерелами, звідки ця інформація може бути отримана, що є достатньо важливим критерієм.

Найбільшою проблемою додатків для харчування є вузька сфера їхньої спрямованості. У той час як кілька додатків містять додаткові поради щодо способу життя щодо фізичних вправ, стресу та уважності, більшість зосереджується виключно на їжі, яку ви їсте, особливо ретельно перевіряючи споживання калорій. Помилково ототожнювати обмеження калорійності зі здоровим харчуванням. Ми не можемо однаково ставитися до всіх типів калорій і повинні рахувати окремі поживні речовини окремо, обмежуючи вуглеводи та цукор, додаючи більше білка.

Непросто створити ідеальну рекомендаційну систему в такій складній області, але ті рішення, які вже існують, можна значно покращити за допомогою використання більш складних принципів, але які будуть давати при цьому більш точні результати.

1.3 Постановка задачі

Під час виконання даної роботи необхідно створити концепт алгоритму рекомендацій щодо здорового харчування, який може надавати користувачам персоналізовані поради щодо досягнення збалансованої та поживної їжі. Алгоритм повинен враховувати дієтичні обмеження і вподобання та надавати рекомендації, адаптовані до його унікальних потреб.

Необхідно обрати алгоритм, який найкраще підходить для вирішення проблеми: існує кілька рекомендаційних алгоритмів, які можна використовувати, як-от спільне фільтрування, фільтрування на основі вмісту та гібридне фільтрування.

Алгоритм повинен:

- враховувати харчові звички та вподобання користувача;
- враховувати інформацію про харчові продукти;
- враховувати фізіологічні особливості користувача;
- швидко надавати рекомендації;
- заповнювати відсутні значення в отриманих даних;

- враховувати додаткову інформацію про продукти, яка не використовується в інших програмних забезпеченнях;
- отримувати інформацію про харчовий продукт за допомогою камери;
- враховувати попередні прийоми їжі під час надання рекомендацій.

Алгоритм повинен мати модульність, для того, щоб його можна було запровадити (після перевірки та оптимізації) в будь-яку існуючу систему, а також створювати на його основі нові програмні рішення.

Необхідно також розглянути нутриціологію, бо це важлива область дослідження для створення системи рекомендацій, пов'язаної зі здоровим харчуванням, оскільки вона забезпечує наукову основу для розуміння харчового вмісту різних продуктів і їх впливу на організм.

2 ПРОВЕДЕННЯ ДОСЛІДЖЕННЯ

2.1 Актуальність та мета

З їжі наш організм отримує необхідні поживні речовини та енергію. Потреба в них залежить від віку, статі, способу життя, стану здоров'я та багатьох інших факторів.

У харчуванні важлива не тільки енергія, що отримується, але й те, щоб вона надходила від різних поживних речовин і пропорція цих речовин була збалансованою. Кожна поживна речовина відіграє в організмі свою важливу роль, і різні групи продуктів містять різні основні речовини, вітаміни та мінерали.

Сьогодні багато хто намагається, наприклад, скорочувати кількість споживаних вуглеводів, що може стати причиною дефіциту деяких важливих поживних речовин.

Поширена помилка, що все смачне чи шкідливе, або призводить до ожиріння. Насправді все не так погано: їсти можна все, але у правильних кількостях і з правильною частотою.

Організм повинен отримувати з їжі як достатню для життєдіяльності кількість енергії, і всі необхідні поживні речовини. Незбалансоване, одноманітне харчування призводить до дефіциту поживних речовин, що різним чином відбивається на організмі і може призвести до серйозних захворювань.

Один із наслідків незбалансованого харчування – проблеми з масою тіла: якщо організм недоотримує з їжею потрібних йому речовин, він змушує нас їсти більше.

Правильне, здорове харчування не означає нудне меню та відмова від смачних страв. Якраз навпаки. Воно має на увазі збалансованість, якої можна досягти, якщо їсти різні продукти і стільки, скільки цього потребує організм. Так він зможе отримувати необхідні поживні речовини, щоб ми могли бути здоровими та енергійними. Тому ніяка їжа не є здоровою чи нездоровою, корисною чи шкідливою сама по собі. Все залежить від розміру порцій та частоти прийому їжі. Їсти можна все, але у правильних кількостях.

Найчастіше причиною одноманітного харчування стають рутини та звички. Рацион потрібно час від часу переглядати, щоб весь час немає одне й те саме. Харчування може бути одноманітним і тому, що людина свідомо вирішила не їсти певних продуктів або вони їй не подобаються. У такому випадку потрібно уважно стежити, щоб необхідні поживні речовини все ж таки надходили з інших джерел.

Наразі неможливо власноруч контролювати склад кожного продукту та вираховувати його допустиму порцію для вживання. На прилавках постійно з'являються нові продукти, більшість з яких націлена на здорове харчування. Але все залежить від міри. Також не існує програмних рішень, які б впоралися з поставленою задачею. Оскільки ще не існує жодного додатку, який би враховував кожен складову продукту разом з особливостями певної людини, та надавав би рекомендації щодо допустимої норми вживання, то дослідження можна вважати актуальним та перспективним.

Таким чином, метою дослідження є запровадження програмної системи, що дозволить на основі особливостей людини швидко надати їй рекомендації щодо потенційної шкоди та допустимої кількості продукту для вживання.

Об'єктом дослідження є здорове та збалансоване харчування продуктами для яких відомий склад з упаковки.

Предметом дослідження є методи виміру допустимої норми речовин в продуктах харчування та формування рекомендацій для певної людини щодо допустимої кількості складових речовин.

2.2 Методи дослідження

Наукове дослідження – комплексне завдання, суть якого полягає у пошуку шляхів вирішення певної проблеми, розробки методів, які дозволять вирішити поставлені питання, та вибору того методу, який найбільше підходить та найкраще справляється з завданням. Проте для того, щоб провести саме дослідження, також існують певні методи його виконання, які дозволяють тим чи іншим способами отримати певні дані для аналізу. Це процес дослідження певного об'єкта (предмета або явища) за допомогою наукових методів, що має на

меті встановлення закономірностей його виникнення, розвитку і перетворення в інтересах раціонального використання у практичній діяльності людей.

Існує певна кількість методів проведення наукового дослідження. Їх вибір залежить від суті поставленої задачі, її характеру та, звичайно, доступних можливостей щодо проведення тих чи інших дій, які передбачає обраний метод відповідно.

Перш за все у ході виконання роботи за основу були взяті емпіричні методи досліджень. Оскільки у науці основними формами емпіричного дослідження є спостереження і експеримент, то вони були використані як основні.

Було почато зі спостереження. Перед тим як вживати той чи інший продукт, звертають увагу на наступні складові будь-якої етикетки.

- термін придатності. Він повинен бути добре надрукований на упаковці. Є магазини, де досі проводять різні махінації з термінами, особливо у випадку з молочними продуктами та іншими товарами, що швидко псуються. Тому необхідно звіряти дату виробництва та закінчення терміну придатності;
- склад продукту. Обов'язково повинні бути зазначені покращувачі смаку, БАДи, ароматизатори, барвники, покращувачі, інші Е, ГМО, продуктів із включенням невластивих їм компонентів білкової природи (наприклад, соя у м'ясних продуктах). Також обов'язково повинні окремо вказуватись потенційно алергенні інгредієнти: арахіс, аспартам (штучний підсолоджувач), гірчиця, діоксид сірки (сульфіт), злаки, що містять глютен або безпосередньо сам глютен, кунжут, люпин, моллюски, ракоподібні, риба, молоко (лактоза), горіхи, селера, соя, яйця та продукти їх переробки.

Ефективність аналізу етикеток продуктів харчування та генерації рекомендацій для користувача тісно залежить від фізіологічних особливостей користувача та якісного складу речовин, наявних в певному продукті. У даній роботі розглядається метод надання рекомендацій щодо допустимої норми, за

яким користувач після вводу своєї фізіологічних даних, наводить камеру свого смартфона на продукт та отримує рекомендації.

Для реалізації мети дослідження буде використано експериментальний метод. Для його виконання необхідно формалізувати проблему, тобто привести її до певної математичної проблеми.

Таким чином, план проведення наукових досліджень має такий вигляд:

- формалізація проблеми дослідження;
- виявлення факторів впливу на ефективність надання рекомендацій користувачеві на основі складу продукту;
- створення експериментальних варіантів реалізації ПЗ для аналізу упаковок товарів та генерації рекомендацій щодо їх вживання;
- проведення експерименту;
- аналіз результатів;
- висновки.

2.3 Методологія теоретичних досліджень, математична модель

Варто розуміти, що надання рекомендацій допустимої норми для різних користувачів відрізняється, оскільки є фактори, які можуть впливати на якість алгоритму: вік, наявність хвороб та образ життя, тощо. Такі фактори можна оцінити кількісно, але є й такі, які оцінити важко, наприклад: наскільки добре розвинений обмін речовин або неконтрольоване вживання інших продуктів. Тож, слід розділити дослідження ефективності методу надання рекомендацій на 2 частини.

Перша частина – це оцінка ефективності з огляду на те, як саме подаються рекомендації з боку програмного забезпечення. Можна виділити такі фактори впливу на ефективність:

- тип ПЗ (мобільний застосунок, веб-застосунок, на даному етапі можна не розглядати інші варіанти реалізації, оскільки наявні є найбільш популярними на даний момент);
- кількість поживних речовин, що входять до складу продукту;

- різноманітність поживних речовин;
- наявність деталізованого опису складу в продукті харчування;
- регулярність дотримання рекомендацій.

Друга частина оцінює ефективність з точки зору користувача.

Фактори впливу такі:

- вік користувача;
- оцінка фізіологічного стану;
- чи веде користувач здоровий образ життя.

Обидві частини можна розглянути у якості регресійних моделей, оскільки має на вході певний набір факторів, та на виході показник ефективності. Даний показник ефективності є оцінкою стану здоров'я користувача.

Для експерименту буде визначено кількість днів, що надається на використання рекомендацій – $N_{\text{зар}}$.

Спочатку розглянемо ефективність саме ПЗ з різними підходами до реалізації способу надання рекомендацій. Позначимо показник ефективності, як Y_s (software). У даній частині цей показник розглядається як середнє значення оцінок, отриманих після медичного огляду групою користувачів. У таблиці 2.1 опишемо детальніше розглянуті фактори впливу.

Таблиця 2.1 – Фактори впливу для програмного забезпечення

Фактор впливу	Опис	Можливі значення
X_{s1}	Цільова платформа для ПЗ. Розглянемо 2 типи: мобільний застосунок та веб-застосунок	1 – мобільний -1 – веб
X_{s2}	Кількість речовин, що входить до складу продукту	[5, 30]
X_{s3}	Різноманітність поживних речовин, що входять до складу.	[1, 10]

Кінець таблиці 2.1

Фактор впливу	Опис	Можливі значення
X_{s4}	Наявність деталізованого складу речовин на упаковці	1 – наявний -1 – відсутній
X_{s5}	Регулярність дотримання рекомендацій. Розглядається, як відношення кількості дотриманих рекомендацій до загальної експериментальної кількості днів, наданих на використання рекомендацій	$N_{\text{факт}} / N_{\text{заг}}$

Матимемо загальне представлення даної математичної моделі у вигляді рівняння регресії (формула 2.1):

$$Y_s = k_{s0} + k_{s1}X_{s1} + k_{s2}X_{s2} + k_{s3}X_{s3} + k_{s4}X_{s4} + k_{s5}X_{s5} \quad (2.1)$$

Коефіцієнти $k_{s0}..k_{s5}$ надаватимуть уявлення про вплив відповідних факторів $X_{s1}..X_{s5}$ на середню оцінку стану здоров'я користувачів, отриману за експериментальний результуючий дотриманий набір рекомендацій.

Далі розглянемо ефективність рекомендацій з точки зору вибірки користувачів. Позначимо показник ефективності, як Y_u (user). У даній частині цей показник розглядається як оцінка стану здоров'я користувача. У таблиці 2.2 було описано розглянуті фактори впливу.

Таблиця 2.2 – Фактори впливу для користувачів

Фактор впливу	Опис	Можливі значення
X_{u1}	Вік користувача. Залежить від сформованої групи користувачів	$[age_{\min}, age_{\max}]$

Кінець таблиці 2.2

Фактор впливу	Опис	Можливі значення
x_{u2}	Оцінка стану здоров'я	[0, 20]
x_{u3}	Наявність здорового образу життя	1 – наявний -1 – відсутній

Слід звернути увагу на фактор x_{u2} . Його можна також розбити на різні оцінки показників здоров'я. Проте у загальному описі використовуємо абстрактне середнє значення такої оцінки.

Матимемо загальне представлення даної математичної моделі у вигляді рівняння регресії (формула 2.2):

$$Y_u = k_{u0} + k_{u1}x_{u1} + k_{u2}x_{u2} + k_{u3}x_{u3} \quad (2.2)$$

Коефіцієнти $k_{u0} \dots k_{u3}$ надаватимуть деяке уявлення про вплив відповідних факторів $x_{u1} \dots x_{u3}$ на оцінку стану здоров'я, отриману під час дотримання рекомендацій.

У результаті отримані дві математичні моделі, які дозволяються дослідити ефективність представленого методу надання рекомендацій з двох боків. Дані фактори можна було б об'єднати у одній моделі, проте такий підхід не дозволяє розрізнити вплив користувачьких факторів та факторів, отриманих від безпосередньої реалізації ПЗ.

2.4 Методологія та проведення експерименту

У попередньому розділі було формалізовано проблему дослідження, та отримано дві схожі математичні моделі.

Для експерименту має бути обране значення $N_{\text{заг}}$, що відповідає за загальну експериментальну кількість днів, наданих на використання рекомендацій.

Також, серед групи користувачів мають бути встановлені значення age_{min} , age_{max} , що відповідають найменшому та найбільшому значенню віку користувачів, що приймають участь у експерименті.

Для проведення експерименту, що стосується першої моделі, слід розділити користувачів на групи за віком та наявністю ведення здорового образу життя. Це надасть змогу більш точно оцінити вплив саме факторів $x_{s1}..x_{s5}$ так, щоб вони найменше залежали від відповідних факторів для другої моделі.

Згрупувавши результати експерименту щодо першої моделі за комбінаціями відповідних факторів $x_{s1}..x_{s5}$, та отримавши середні значення Y_s , відповідних даним групам, можна буде отримати рівняння регресії та коефіцієнти $k_{s0}..k_{s5}$.

Після чого, необхідно виконати наступний експеримент, який базується на опитуванні користувачів-учасників першого експерименту. Отримані дані, співставлені зі значеннями Y_u , використаємо для знаходження другого рівняння регресії і коефіцієнтів $k_{u0}..k_{u3}$.

Результатом експерименту є набір коефіцієнтів, що надають представлення про вплив відповідних факторів на ефективність надання рекомендацій представленим методом. Більш ефективним результатом вважається той, який має більше значення, отже, якщо коефіцієнт додатний, то відповідний йому фактор при збільшенні свого значення покращує ефективність, а при зменшенні погіршує, і навпаки, якщо коефіцієнт від'ємний.

Проаналізувавши результати першого експерименту, можна буде зробити висновок про найкращий підхід до вибору реалізації методу надання рекомендацій. Аналіз же другого експерименту виявляє, на яку саме групу користувачів можна орієнтувати такий підхід до надання рекомендацій.

Звісно, на результат експериментів можуть вплинути згадані раніше показники, які важко оцінити кількісно: неконтрольоване вживання інших продуктів та розміри порцій, обмін речовинами, тощо. Тож, відслідкувати вплив

таких факторів на експеримент важко. Проте, цілком можна вважати, що людина, яка розпочала дотримання рекомендацій, швидше за все зацікавлена у покращенні свого стану здоров'я та не буде зловживати неконтрольованими факторами. Щодо інших чинників, можна вважати, що результати експерименту будуть більш усередненими, що йде лише на користь експерименту.

2.5 Вибір рекомендаційного алгоритму

Рекомендаційні системи – програми, які намагаються передбачити, які об'єкти (фільми, музика, книги, новини, веб-сайти) будуть цікавими для користувача, маючи певну інформацію про його профіль. Дві основні стратегії створення рекомендаційних систем – фільтрація на основі змісту та колаборативна фільтрація.

При фільтрації на основі змісту створюються профілі користувачів та об'єктів, профілі користувачів можуть включати демографічну інформацію або відповіді на певний набір питань, профілі об'єктів можуть містити назви жанрів, імена акторів, імена виконавців та іншу атрибутивну інформацію залежно від типу об'єкта.

При колаборативній фільтрації використовується інформація про поведінку користувачів у минулому – наприклад, інформація про покупки або оцінки. У цьому випадку немає значення, з якими типами об'єктів ведеться робота, але при цьому можуть враховуватися неявні характеристики, які складно було б врахувати при створенні профілю. Основна проблема цього типу рекомендаційних систем – «холодний старт»: відсутність даних про користувачів, що недавно з'явилися в системі, або об'єктах.

Очевидно, що для рекомендаційних систем, пов'язаних зі здоровим харчуванням колаборативна фільтрація не може бути застосована, оскільки кожна людина повинна мати індивідуальний підхід. Колаборативна фільтрація зазвичай ігнорує ці окремі фактори та може не надати точних рекомендацій, адаптованих до унікальних потреб кожного користувача. Тож всі алгоритми для цієї сфери основані на фільтрації на основі змісту, тож порівняння відбувалося на основі

того, як створюються профілі в даних системах. Для того, щоб дослідити існуючі алгоритми рекомендаційних систем для здорового харчування необхідно сформулювати змістовну постановку багатокритеріальної задачі прийняття рішень в області даного магістерського дослідження, провести етап детальної інформаційної підготовки прийняття рішення, навести векторний опис цієї задачі та знайти її найкраще рішення на базі метода теорії корисності.

2.6 Проведення лінійної згортки

Була сформована наступна задача для вирішення – індивідуальна рекомендація щодо здорового харчування є доволі складною задачею, оскільки потрібно враховувати безліч параметрів, що б обрати найкращий алгоритм. Тож потрібно врахувати функціональні можливості алгоритмів та обрати найбільш привабливий [3].

Було проведено лінійну згортку з ваговими коефіцієнтами на зазначені алгоритми.

Критерії:

F1 – присутність розпізнавання по фото;

F2 – врахування віку;

F3 – можливість внесення фізіологічних особливостей людини;

F4 – врахування калорій;

F5 – врахування статі;

F6 – швидкість виконання;

F7 – Враховує активність життя;

F8 – Відсоток розпаралелення виконання алгоритму;

F9 – можливість сформулювати план харчування.

Критерії: $f_1, f_2, f_4, f_5, f_7, f_9$ мають значення 0 – 1 {ні, так}

Критерії f_3, f_6, f_8 мають звичайні значення які потрібно нормалізувати.

Векторний опис задачі (формула 2.3):

$$w(x) = 0.25f_1 + 0.05f_2 + 0.15f_3 + 0.1f_4 + 0.05f_5 + 0.1f_6 + 0.05f_7 + 0.1f_8 + 0.15f_9 \quad (2.3)$$

Таблиця 2.3 – Значення критеріїв для обраних алгоритмів

Алгоритми	F1	F2	F3	F4	F5	F6	F7	F8	F9
A	1	1	14%	0	0	300 інструкцій/мс	0	27.5%	1
B	0	0	14%	0	1	180 інструкцій/мс	1	12.5%	0
C	0	1	70%	1	0	100 інструкцій/мс	0	5%	1
D	1	0	42%	0	1	500 інструкцій/мс	1	65%	0
E	0	1	0%	0	1	260 інструкцій/мс	0	80%	0

Було зроблено перевірку за принципом Парето та визначено, що не існують домінантності між двома алгоритмами (формула 2.4).

$$x_{norm} = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (2.4)$$

де x – значення що зараз обробляється,
 $\min(x)$ – мінімальне значення серед усіх,
 $\max(x)$ – максимальне значення серед усіх.

Таблиця 2.4 – Нормовані значення критеріїв для обраних алгоритмів

Алгоритми	F1	F2	F3	F4	F5	F6	F7	F8	F9
A	1	1	0.2	0	0	0.5	0	0.3	1
B	0	0	0.2	0	1	0.2	1	0.1	0
C	0	1	1	1	0	0	0	0	1
D	1	0	0.6	0	1	1	1	0.8	0
E	0	1	0	0	1	0.4	0	1	0

Для кожного алгоритму було знайдено значення $w(x)$

Таблиця 2.5 – Значення $w(x)$

Алгоритм	$w(x)$
A	0.56
B	0.16

Кінець таблиці 2.5

Алгоритм	w(x)
C	0.45
D	0.62
E	0.24

Було визначено, що алгоритм D є найбільш привабливим, а алгоритм B та E мають найнижчі показники.

2.7 Аналіз програмного забезпечення за допомогою планування експерименту

Під час аналізу апріорної інформації про програмне забезпечення стало відомо наступне:

- ПЗ є мобільним додатком, написаним на Xamarin;
- сервер написаний на .Net Core 3.1, де є модуль розпізнавання тексту, парсингу інформації з тексту та модуль генерації рекомендацій.
- ПЗ використовує СУБД PostgreSQL для зберігання даних.

Було використано математичну модель експерименту у вигляді лінійної моделі без залежностей між факторами [4]. Таке припущення і сукупність факторів не є ідеальними, проте суттєво спрощує роботу. Задача зводиться до знаходження значень коефіцієнтів рівняння регресії (див.рис.2.1).

$$y = k_0 + k_1x_1 + k_2x_2 + k_3x_3 + k_4x_4 + k_5x_5$$

Рисунок 2.1 – Рівняння регресії

Фактори з найбільшими значеннями будуть найбільше впливати на вихідну характеристику.

Найбільш суттєвим з апріорного аналізу є отримання необхідної інформації, тому було обрано наступні фактори впливу, що мають зв'язок з цим параметром (див. Таблицю 2.6).

Таблиця 2.6 – Фактори впливу

Фактор впливу	Опис
X ₁	Розмір фотографії, яка передається до серверу на обробку
X ₂	Швидкість інтернет з'єднання
X ₃	Час розпізнавання тексту
X ₄	Час парсингу тексту для виділення важливих значень.
X ₅	Час формування рекомендацій для конкретного користувача.

Було обрано верхній і нижній рівень для кожного фактора:

+1 відповідає верхньому рівню фактора;

-1 відповідає нижньому рівню фактора.

Нижче наведено параметри, що відповідають верхньому та нижньому рівню (див. таблицю 2.7).

Таблиця 2.7 – Верхні та нижні рівні факторів впливу

Фактор впливу	Верхнє значення	Нижнє значення
X ₁	0.1 MB	20 MB
X ₂	60 Mbps	10 Mbps
X ₃	0.01 s	2 s
X ₄	0.01 s	2 s
X ₅	0.1 s	2 s

Відповідно до факторів та їх верхнього та нижнього рівнів було створено таблицю та виконано експеримент, результати якого занесено до Таблиці 2.8.

Таблиці 2.8 – Результати виконання експерименту

	X ₁	X ₂	X ₃	X ₄	X ₅	Y
1	-1	-1	-1	-1	-1	10.05441525
2	-1	-1	-1	-1	1	6.743169476
3	-1	1	-1	1	-1	5.251189962
4	-1	-1	-1	1	1	6.060438861
5	-1	-1	1	-1	-1	9.644006354

Кінець таблиці 2.8

	x_1	x_2	x_3	x_4	x_5	Y
6	-1	-1	1	-1	1	6.079392243
7	-1	-1	1	1	-1	9.373032003
8	-1	-1	1	1	1	6.362392976
9	-1	1	-1	-1	-1	4.893253624
10	-1	1	-1	-1	1	2.138394979
11	-1	1	-1	1	-1	4.684791247
12	-1	1	-1	1	1	1.532113626
13	-1	1	1	-1	-1	4.608159443
14	-1	1	1	-1	1	2.081677907
15	-1	1	1	1	-1	4.248022165
16	-1	1	1	1	1	1.210605102
17	1	-1	-1	-1	-1	8.691659326
18	1	-1	-1	-1	1	5.670575138
19	-1	-1	-1	1	-1	9.269251772
20	1	-1	-1	1	1	5.791910647
21	1	-1	1	-1	-1	8.792166448
22	1	-1	1	-1	1	5.474078901
23	1	-1	-1	1	-1	8.656451821
24	1	-1	1	1	-1	8.086674062
25	1	1	-1	-1	-1	4.735497155
26	1	1	-1	-1	1	1.712193448
27	1	1	-1	1	-1	4.017601767
28	1	1	-1	1	1	0.834007104
29	1	1	1	-1	-1	3.773650453
30	1	1	1	-1	1	0.813942138
31	1	1	1	1	-1	3.177434101
32	1	-1	1	1	1	4.654726937

Де Y відповідає часу завантаження сторінки, що буде використано для аналізу в програмі, визначається в секундах.

Коефіцієнти рівняння регресії було знайдено за наступною формулою, яку наведено нижче (формула 2.5)

$$k_j = \frac{\sum_{i=1}^N x_{ji} Y_i}{N}, j = 1, k \quad (2.5)$$

Результати обчислення було розміщено в таблиці 2.9.

Таблиця 2.9 – Результати обчислення

a_0	a_1	a_2	a_3	a_4	a_5
5.1465	-0.4418	-2.3163	-0.2232	-0.2226	-1.5495

Коефіцієнти рівняння регресії зображено на рисунку 2.2.

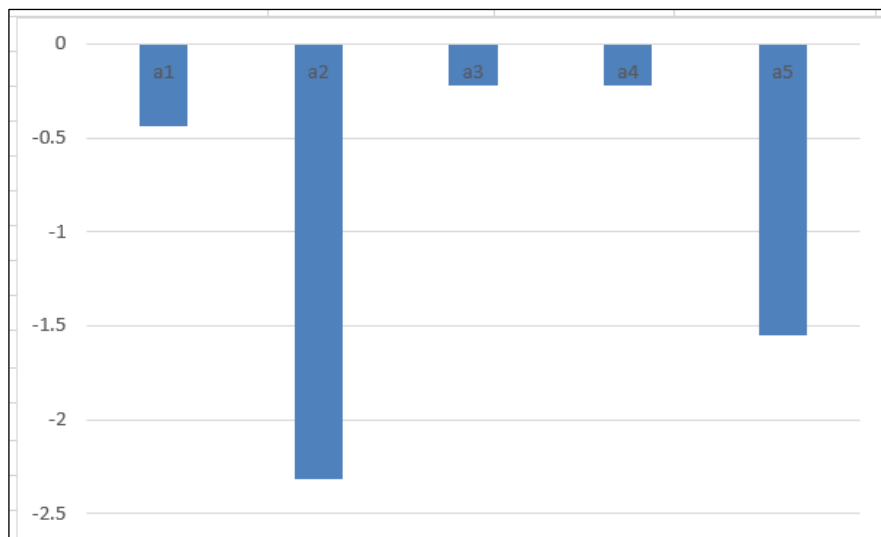


Рисунок 2.2 – Коефіцієнти рівняння регресії

Рівняння регресії має наступний вигляд (формула 2.6):

$$Y = 5,1465 - 0,4418x_1 - 2,3163x_2 - 0,2232x_3 - 0,2226x_4 - 1,5495x_5 \quad (2.6)$$

За допомогою коефіцієнтів було визначено, що найбільший вплив на результат мають фактори x_2 та x_5 , які відповідають за швидкість інтернету та генерування рекомендацій для користувача. Від'ємне значення коефіцієнтів регресії вказує на те, що дані фактори зменшують час відповіді серверу [5].

За допомогою Excel було побудовано регресійну модель. Для цього було відкрито Excel та створено там новий workbook «Regression». Після цього було створено новий аркуш, а стовпці були позначені у відповідності до назв коефіцієнтів для вхідних даних.

Потім було введено ці дані в робочий аркуш. Після введення, було виконано детальну перевірку того, що кожне числове значення введено в правильний стовпець.

Результат внесення вхідних даних відповідно до коефіцієнтів, зображено на рисунку 2.3.

	A	B	C	D	E	F	G
1		Y	x1	x2	x3	x4	x5
2	1	10.05441525	-1	-1	-1	-1	-1
3	2	6.743169476	-1	-1	-1	-1	-1
4	3	5.251189962	-1	1	-1	1	-1
5	4	6.060438861	-1	-1	-1	1	1
6	5	9.644006354	-1	-1	1	-1	-1
7	6	6.079392243	-1	-1	1	-1	1
8	7	9.373032003	-1	-1	1	1	-1
9	8	6.362392976	-1	-1	1	1	1
10	9	4.893253624	-1	1	-1	-1	-1
11	10	2.138394979	-1	1	-1	-1	-1
12	11	4.684791247	-1	1	-1	1	-1
13	12	1.532113626	-1	1	-1	1	1
14	13	4.608159443	-1	1	1	-1	-1
15	14	2.081677907	-1	1	1	-1	1
16	15	4.248022165	-1	1	1	1	-1
17	16	1.210605102	-1	1	1	1	1
18	17	8.691659326	1	-1	-1	-1	-1
19	18	5.670575138	1	-1	-1	-1	-1
20	19	9.269251772	-1	-1	-1	1	-1
21	20	5.791910647	1	-1	-1	1	1
22	21	8.792166448	1	-1	1	-1	-1
23	22	5.474078901	1	-1	1	-1	1
24	23	8.656451821	1	-1	-1	1	-1
25	24	8.086674062	1	-1	1	1	-1
26	25	4.735497155	1	1	-1	-1	-1
27	26	1.712193448	1	1	-1	-1	1
28	27	4.017601767	1	1	-1	1	-1
29	28	0.834007104	1	1	-1	1	1
30	29	3.773650453	1	1	1	-1	-1
31	30	0.813942138	1	1	1	-1	1
32	31	3.177434101	1	1	1	1	-1
33	32	4.654726937	1	-1	1	1	1

Рисунок 2.3 – Вхідні дані в програмі Excel

Після цього було запущено регресійний аналіз, результати якого зображено на рисунку 2.4.

SUMMARY OUTPUT		Regression Statistics	
4	Multiple R	0.995604307	
5	R Square	0.991227935	
6	Adjusted R Square	0.989541	
7	Standard Error	0.282719973	
8	Observations	32	

ANOVA		df	SS	MS	F	Significance F
12	Regression	5	234.8324062	46.96648124	587.5908754	7.27637E-26
13	Residual	26	2.078195158	0.079930583		
14	Total	31	236.9106014			

	Coefficients	Standard Error	t Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%	
17	Intercept	5.146502487	0.050280898	102.3550225	2.03516E-35	5.04314862	5.249856354	5.04314862	5.249856354
18	X Variable 1	-0.441767014	0.050280898	-8.785980915	2.92014E-09	-0.545120881	-0.338413147	-0.545120881	-0.338413147
19	X Variable 2	-2.316268902	0.050280898	-46.06657747	1.86466E-26	-2.419622768	-2.212915035	-2.419622768	-2.212915035
20	X Variable 3	-0.223180027	0.050280898	-4.438664266	0.000147883	-0.326533894	-0.11982616	-0.326533894	-0.11982616
21	X Variable 4	-0.222637031	0.050280898	-4.427865014	0.000152152	-0.325990898	-0.119283164	-0.325990898	-0.119283164
22	X Variable 5	-1.549451386	0.050280898	-30.81590496	5.36691E-22	-1.652805253	-1.44609752	-1.652805253	-1.44609752

Y = a0 + a1x1 + a2x2 + a3x3 + a4x4 + a5x5

Регресійна модель

Y = 5,1465 - 0,4418x1 - 2,3163x2 - 0,2232x3 - 0,2226x4 - 1,5495x5

Рисунок 2.4 – Результати регресійного аналізу

Після регресійного аналізу Excel створив відповідні таблиці та графік, які відповідають за коефіцієнти.

За допомогою рівняння регресії були вираховані прогнозовані значення, які знаходяться в стовпчику Y^{\wedge} (див. рис. 2.5).

SUM								
=SUM(5.1465- 0.4418*C2 - 2.3163*D2 - 0.2232*E2 - 0.2226*F2 - 1.5495*G2)								
A	B	C	D	E	F	G	H	
	Y	x1	x2	x3	x4	x5	Y'	
2	1	10.05441525	-1	-1	-1	-1	-1	2 - 1.5495*G2
3	2	6.743169476	-1	-1	-1	-1	1	6.8009
4	3	5.251189962	-1	1	-1	1	-1	4.8221
5	4	6.060438861	-1	-1	-1	1	1	6.3557
6	5	9.644006354	-1	-1	1	-1	-1	9.4535
7	6	6.079392243	-1	-1	1	-1	1	6.3545
8	7	9.373032003	-1	-1	1	1	-1	9.0083
9	8	6.362392976	-1	-1	1	1	1	5.9093
10	9	4.893253624	-1	1	-1	-1	-1	5.2673
11	10	2.138394979	-1	1	-1	-1	1	2.1683
12	11	4.684791247	-1	1	-1	1	-1	4.8221
13	12	1.532113626	-1	1	-1	1	1	1.7231
14	13	4.608159443	-1	1	1	-1	-1	4.8209
15	14	2.081677907	-1	1	1	-1	1	1.7219
16	15	4.248022165	-1	1	1	1	-1	4.3757
17	16	1.210605102	-1	1	1	1	1	1.2767
18	17	8.691659326	1	-1	-1	-1	-1	9.0163
19	18	5.670575138	1	-1	-1	-1	1	5.9173
20	19	9.269251772	-1	-1	-1	1	-1	9.4547
21	20	5.791910647	1	-1	-1	-1	1	5.4721
22	21	8.792166448	1	-1	1	-1	-1	8.5699
23	22	5.474078901	1	-1	-1	-1	1	5.4709
24	23	8.656451821	1	-1	-1	1	-1	8.5711
25	24	8.086674062	1	-1	1	1	-1	8.1247
26	25	4.735497155	1	1	-1	-1	-1	4.3837
27	26	1.712193448	1	1	-1	-1	1	1.2847
28	27	4.017601767	1	1	-1	1	-1	3.9385
29	28	0.834007104	1	1	-1	1	1	0.8395
30	29	3.773650453	1	1	1	-1	-1	3.9373
31	30	0.813942138	1	1	1	-1	1	0.8383
32	31	3.177434101	1	1	1	1	-1	3.4921
33	32	4.654726937	1	-1	1	1	1	5.0257

Рисунок 2.5 – Прогнозовані значення

Було побудовано графіки для реальних вимірів та прогнозованих, які зображено на рисунку 2.6.

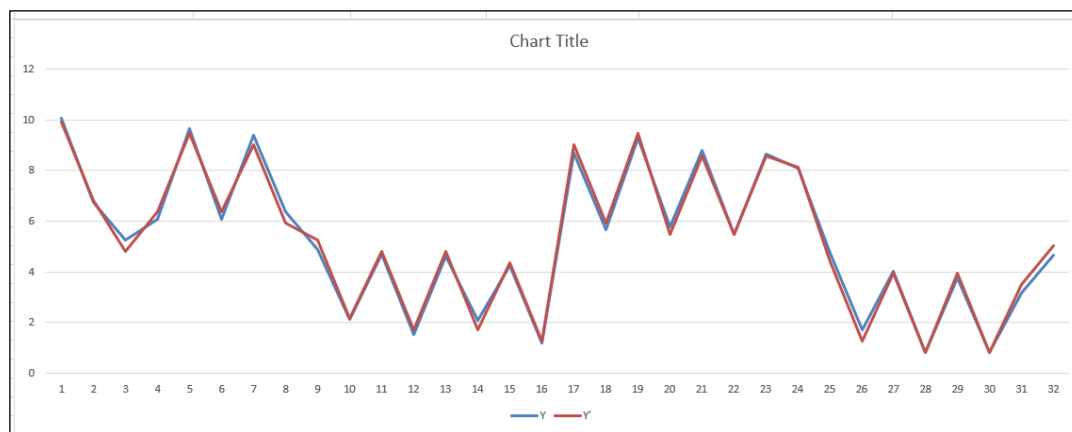


Рисунок 2.6 – Графік реальних та спрогнозованих значень

В результаті було встановлено, що графіки для прогнозованих та реальних значень майже збігаються.

3 АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕНЬ

3.1 Аналіз

Було досліджено кілька програмних систем, пов'язаних із здоровим харчуванням, щоб визначити їхні особливості, сильні та слабкі сторони. Завдяки цим дослідженням було встановлено, що багато існуючих систем на ринку мають такі корисні функції, як здатність відстежувати споживання калорій, надавати рекомендації щодо здорового харчування та контролювати фізичну активність. Однак ці системи обмежені в кількості параметрів, які відстежуються та на основі яких надаються рекомендації.

Метою даного розділу є аналіз результатів дослідження, яке було проведено на програмних системах, що вже існують в сфері, пов'язаною із здоровим харчуванням. Дослідження включало широкий аналіз різноманітних додатків і веб-сайтів, які пропонують послуги, пов'язані з прийомом їжі. Акцент був зосереджений на виявленні тих можливостей, які надаються проаналізованими програмними системами, і функцій, які найчастіше використовуються.

Для проведення дослідження використовувався системний підхід. Комплексний пошук проводився на різних онлайн-платформах, включаючи магазини додатків, веб-сайти та пошуковими сервісами. Для пошуку використовувалися найрізноманітніші ключові слова, такі як «харчування», «дієта», «здоров'я», «планування харчування», «рецепт» та ін.. Всього було ідентифіковано та проаналізовано 50 програмних систем. Ті системи, що відрізняються від інших найбільше були наведені в розділі 1.

Аналіз виявлених систем програмного забезпечення показав, що існує широкий спектр функцій та можливостей, які зазвичай пропонують ці системи. До них належать планування та відстеження прийому їжі, керування рецептами, аналіз поживних речовин, харчовий щоденник і списки покупок. Деякі системи також пропонують додаткові функції, такі як соціальні мережі, відстеження фізичної активності та персоналізований тренер.

Крім того, аналіз також показав, що існують деякі обмеження в наявних програмних системах. Деякі системи незручні для користувача, а інші не надають

точної інформації або недостатньо гнучкі, щоб задовольнити індивідуальні потреби користувачів. Таким чином, існує потреба в розробці удосконаленої програмної системи, яка могла б надавати користувачам персоналізовані рекомендації щодо харчування.

Виходячи з досліджень, очевидно, що наявний високий попит на програмні системи, які пропонують послуги, пов'язані з харчуванням. Однак ті системи, що вже існують, мають деякі обмеження щодо їх можливостей та налаштувань. Більшість систем пропонують фіксований набір функцій і особливостей, які можуть підійти не всім користувачам.

Щоб усунути ці обмеження, необхідно розробити програмну систему, яка поєднує в собі ті особливості, що використовуються найчастіше в існуючих рекомендаційних системах, а також додати до цього переліку нові функції, які будуть базуватися на останніх дослідженнях у галузі здорового харчування. Така система буде достатньо гнучкою та з наявністю додаткового налаштування, щоб задовольнити потреби різних користувачів.

Розробка системи повинна починатися з моделювання цієї системи. Абстрактна модель є представленням функціональних можливостей системи на найвищому рівні, що є важливим для розвитку чіткого розуміння вимог і можливостей системи. Створивши абстрактну модель для програмної системи, яка надає рекомендації щодо харчування, можливо встановити спільне розуміння того, для чого призначена система та як вона працюватиме, як всередині, так і зовні [6].

Це спільне розуміння має вирішальне значення, оскільки воно допомагає переконатися, що всі зацікавлені сторони узгоджені та працюють для досягнення однієї мети. Крім того, створення абстрактної моделі дає змогу визначити потенційні проблеми чи обмеження в системі, перш ніж вкладати значний час і ресурси в її впровадження. Це може допомогти зменшити ризик дорогих помилок або затримок під час розробки.

Отримавши чітке розуміння вимог і можливостей системи через абстрактну модель, можливо почати планувати архітектуру системи. Це передбачає

визначення компонентів системи, взаємодії та інтерфейсів, а також визначення того, як вони працюватимуть разом для досягнення цілей системи [7].

Після встановлення архітектури системи можна приступати до планування особливостей її впровадження. Це включає вибір технологій, визначення структур даних і розробку алгоритмів для досягнення запланованої функціональності системи.

Дотримуючись цього підходу, можливо розробити програмну систему для рекомендацій щодо здорового харчування, яка є добре розробленою, ефективною та корисною для задоволення потреб користувачів. Такий підхід допомагає мінімізувати ризики та підвищити шанси на успіх проекту.

3.2 Формування моделі рекомендаційної системи

Перед детальним проектуванням реальної програмної системи необхідно визначитись з принципами та логікою, за якими дана система буде працювати та які саме дії вона буде виконувати. Важливо враховувати інфраструктуру, на якій вона буде побудована. Одне важливе рішення, яке ми повинні прийняти, полягає в тому, чи реалізувати систему як клієнтську чи серверну архітектуру [8].

Пропонується абстрактна модель рекомендаційної системи для здорового харчування, яка всі функції буде виконувати на сервері, а не на клієнтській платформі. Хоча клієнтська архітектура може запропонувати певні переваги, наприклад підвищену швидкість і реакцію, зрештою було вирішено зупинитись на серверній архітектурі. Існує кілька причин, чому серверне виконання є більш доречним для цього типу системи. По-перше, обробка на сервері забезпечує високу продуктивність, необхідну для роботи з великими обсягами даних, пов'язаних з харчовими продуктами та вмістом у них поживних речовин. По-друге, обробка на сервері гарантує, що система може обробляти запити від кількох клієнтів одночасно, що важливо для масштабування системи зі збільшенням кількості користувачів. За допомогою серверної системи легко скористатися перевагами потужних апаратних і програмних ресурсів, які важко або неможливо відтворити на клієнтському пристрої. Це означає, що можливо

запускати складніші алгоритми та виконувати інтенсивніші обчислення за менший проміжок часу, що забезпечує плавну та безперебійну роботу користувача в цілому.

Ще однією важливою перевагою серверної системи є здатність обробляти великі обсяги даних. В даному випадку це означає надання користувачеві можливості аналізу кількох зображень одного й того ж харчового продукту, а також етикеток з текстовою інформацією. Сервер витягне якомога більше інформації з цих зображень і міток. Обробляючи ці зображення на сервері, можна отримати необхідні дані про харчовий продукт, які було б важко або неможливо отримати іншими способами. Хоча отримана інформація не завжди може бути повною, вона все ж може містити в собі достатній рівень деталей про харчовий продукт, щоб система могла надавати обґрунтовані рекомендації.

Однією з ключових особливостей системи є її здатність до аналізу харчових продуктів [9]. Отримавши цю інформацію, рекомендаційна система повинна виділити в харчовому продукті різні компоненти, які в ньому наявні. До таких компонентів можна віднести білки, жири, вуглеводи, мікроелементи, глюкозу тощо. Для кожного з таких компонентів існують певні обмеження щодо їх споживання. Такі обмеження можуть містити в собі як мінімальну, так і максимальну кількість компоненту, яку варто вжити за певний проміжок часу. Варто зазначити, що дана кількість напряму залежить від фізіологічних особливостей людини, тому правила вживання повинні їх враховувати і застосовуватися до людей залежно від віку, ваги, способу життя та інших факторів. В даній системі повинна бути можливість вільно налаштовувати всі ці правила.

Наступною вимогою до системи повинна бути здатність збереження історії спожитих харчових продуктів. Налаштування в даній системі правил лімітів та можливість відстежувати яку їжу вживає користувач, дає легкість в обчисленні залишків допустимих норм і попереджати користувача, якщо він спожив недостатньо якогось компонента.

Також повинна бути можливість налаштувати список алергій на компоненти в даній системі та попереджати користувачів, якщо вони збираються споживати їжу, яка містить проблемний компонент. Відстежуючи, що їсть користувач і скільки він споживає, можна надати йому точну інформацію про споживання поживних речовин і переконатися, що вони залишаються в межах рекомендованих лімітів.

Окрім аналізу поживних речовин, дана система також повинна дозволяти користувачам зберігати інформацію про харчові продукти, які вони споживають, зокрема про те, скільки вони їдять і коли. Цю інформацію можна використовувати для надання точніших рекомендацій і допомоги користувачам у прийнятті зважених рішень щодо свого харчування.

Також дана система може попередити користувачів, якщо закінчився термін придатності їжі, яку вони збираються споживати. Попереджаючи користувачів про потенційні ризики для здоров'я, пов'язані з простроченою їжею, система може допомогти їм зробити більш безпечний і здоровий вибір.

Загалом серверна модель, яку було запропоновано для даної рекомендаційної системи щодо здорового харчування, пропонує низку ключових переваг порівняно з клієнтською моделлю. Використовуючи потужні апаратні та програмні ресурси, обробляючи великі обсяги даних і пропонуючи розширені можливості відстеження та аналізу, дана система може надавати користувачам дуже персоналізовані та точні рекомендації щодо харчування.

4 РОЗРОБКА ПРОГРАМНОЇ СИСТЕМИ

4.1 Архітектура та проектування

Загальна архітектура для проектування системи – «клієнт-сервер», яка являє собою класичний шаблон для проектування, який можна побачити на діаграмі розгортання. Вся основна функціональна логіка системи повинна бути зосереджена на сервері, а мобільні додатки та сайт повинні просто відображати результати роботи.

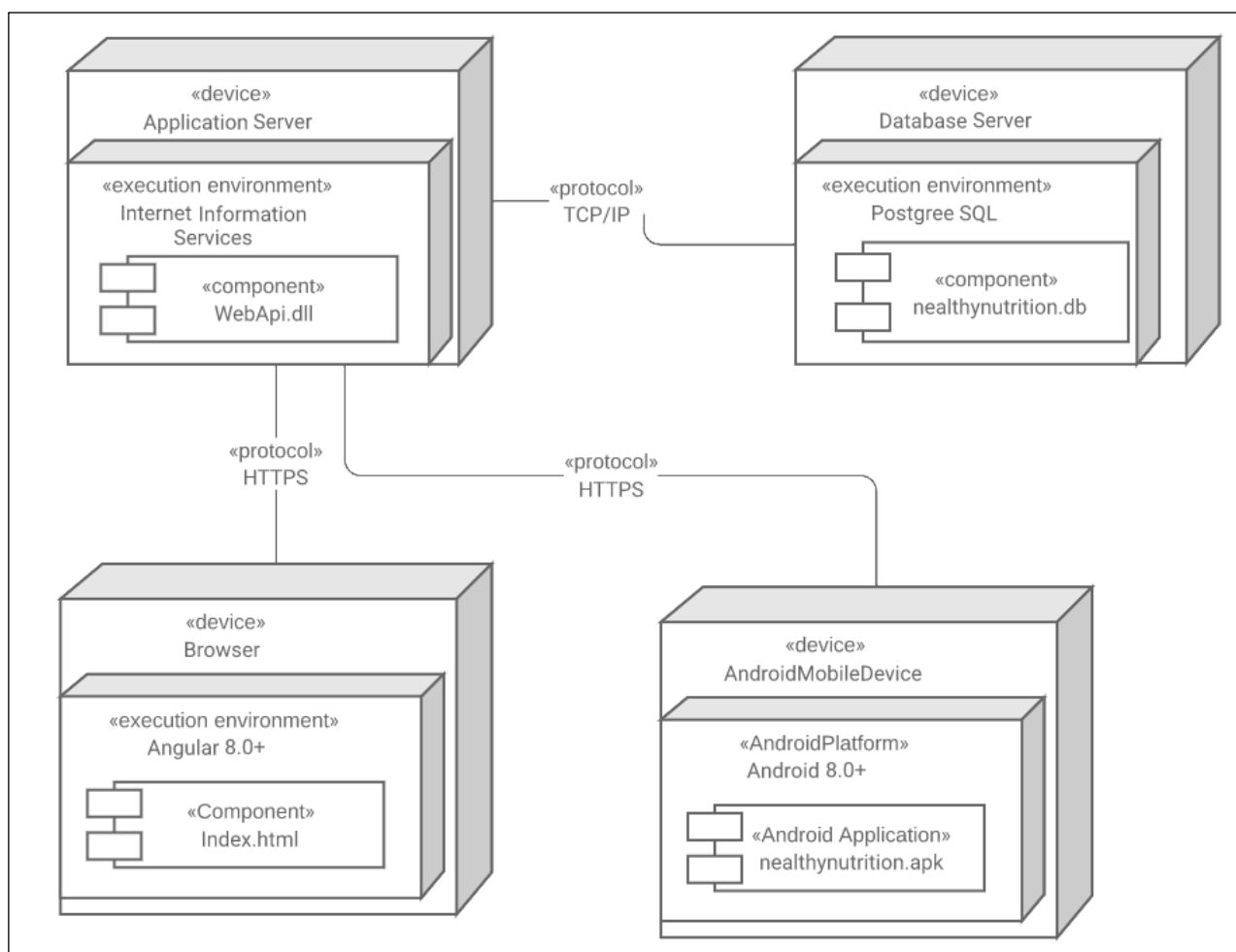


Рисунок 4.1 – Архітектура програмної системи

Користувач зможе використовувати всі функції даної рекомендаційної системи як на своєму мобільному пристрої, так і в браузері, оскільки вся логіка зосереджена на сервері, а клієнтські додатки зможуть її використовувати.

Логіка системи полягає в тому, що адміністратори (люди, які мають доступ до медичних знань) будуть налаштовувати ті компоненти, з яких складається

харчовий продукт, а також норми прийому даних компонентів. Для опису функцій системи було створено діаграму прецедентів, яку зображено на рисунку 4.2.

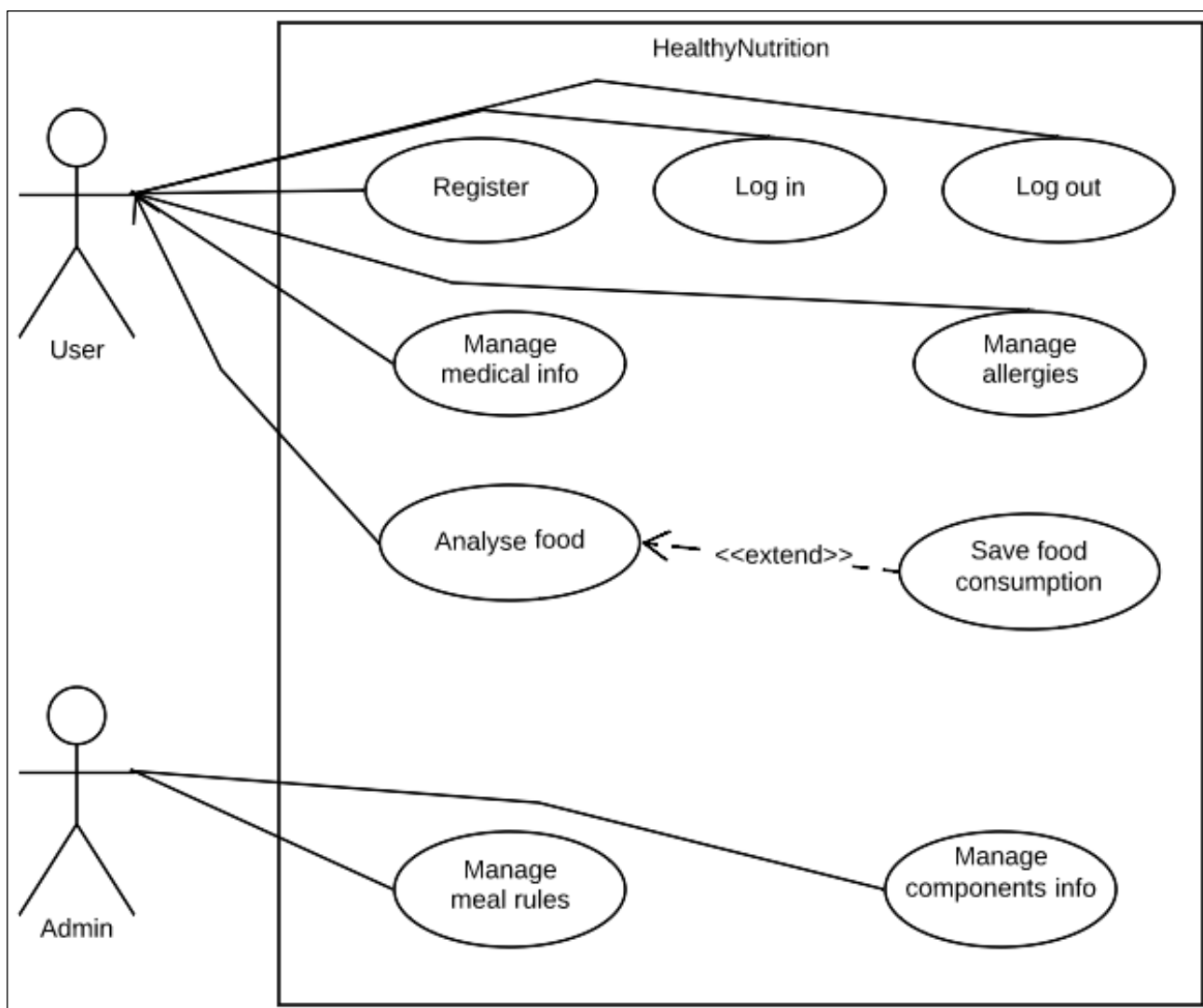


Рисунок 4.2 – Use Case діаграма роботи системи

Після виділення вимог до системи та архітектури, було спроектовано класи, необхідні для надання рекомендацій. Є класи, які представляють інформацію про користувача, його фізіологічні особливості здоров'я. Є клас, який відповідає за таке поняття, як компонент харчового продукту, що може містити в собі будь-яке значення, таке як калорія або калій. Є клас, який відповідає за історію споживання відповідного компонента користувачем, він містить в собі дату та час прийому, а також кількість та який саме компонент було вжито. Є клас, який відповідає за норму прийому компонента. Воно містить кількість та період в часі. Кожне

правило можна налаштувати тим, до кого це правило відноситься. Існує ціла ієрархія спадкування для цих налаштувань. Для того, що все зрозуміти детально було створено діаграму класів, зображену на рисунку 4.3.

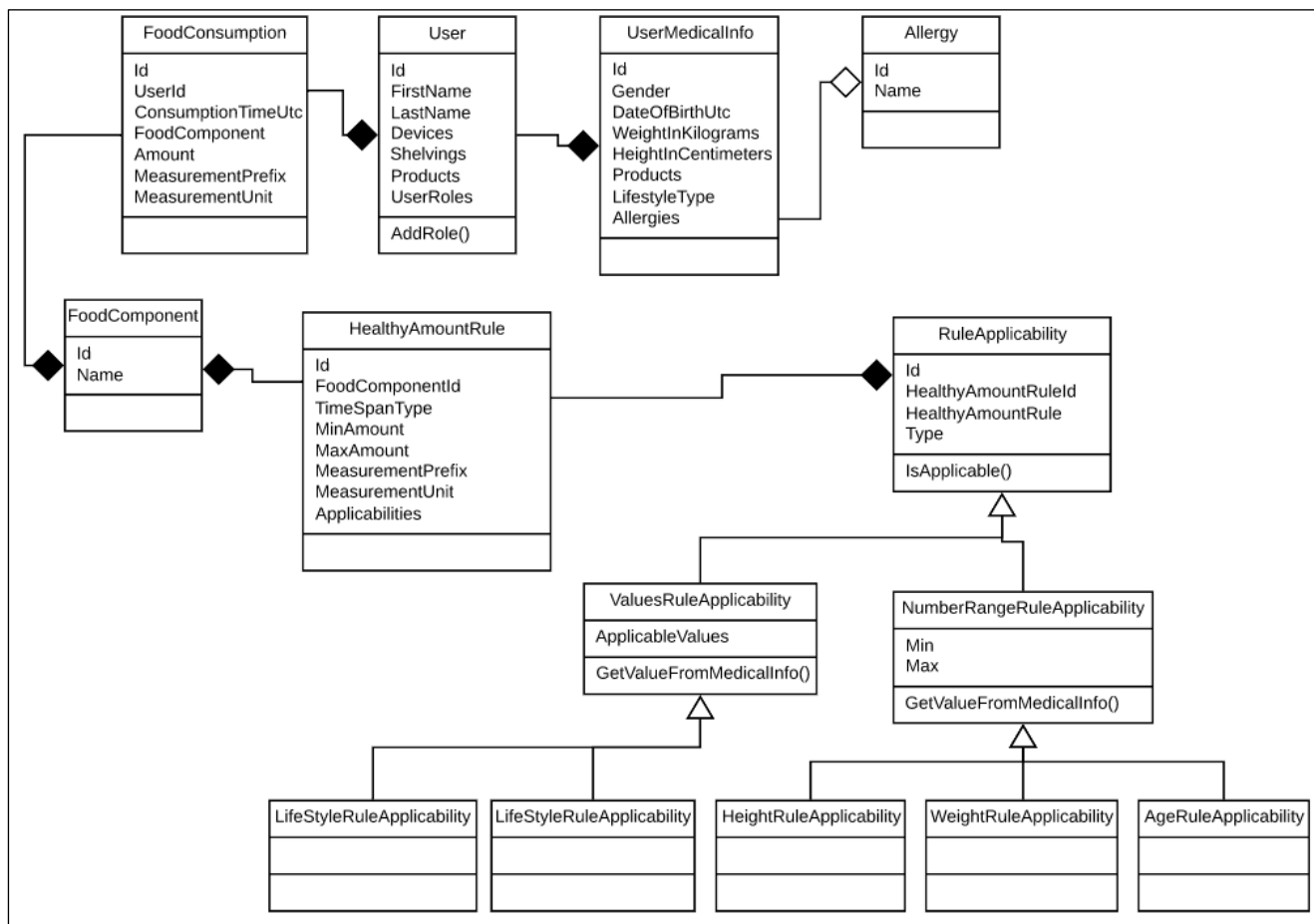


Рисунок 4.3 – Діаграма класів

Зазвичай спочатку створюють діаграму класів, оскільки вона представляє концептуальний погляд на структуру програмної системи. Після встановлення класів та їх взаємодії наступним кроком є розробка схеми бази даних системи, де в де в пригоді стає ER діаграма. Вона допомагає візуалізувати логічну структуру бази даних та забезпечує чітке уявлення про сутності, їхні атрибути та зв'язки між ними, що полегшує перетворення діаграми класів у схему бази даних. Створюючи діаграму ER після діаграми класів, це гарантує, що схема бази даних відображає вимоги системи та забезпечує узгодженість між двома представленнями. Створена діаграма зображена на рисунку 4.4.

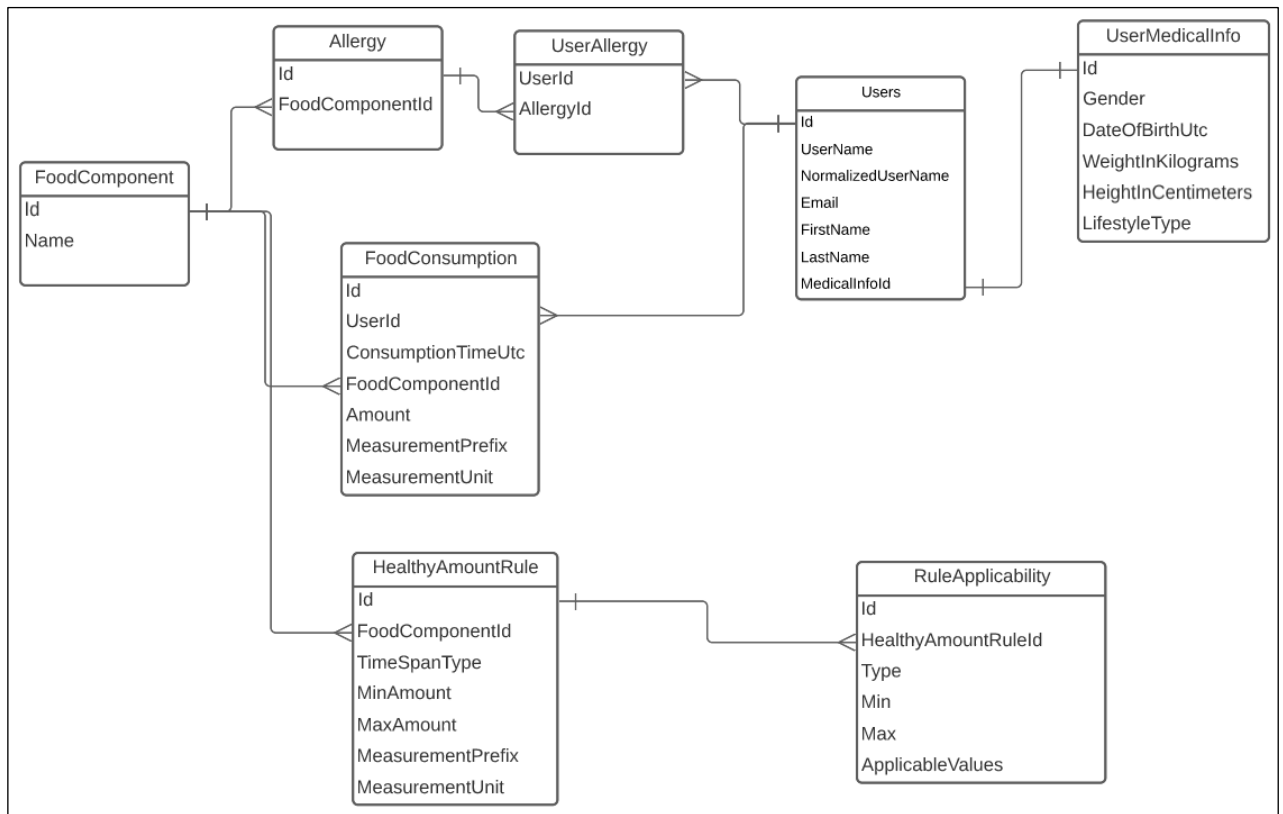


Рисунок 4.4 – ER діаграма

Після проектування системи була виконана безпосередня реалізація даної рекомендаційної системи щодо здорового харчування.

4.2 Програмні рішення

Сервер було створено мові програмування C# та фреймворку ASP.Net Core 3.1. Доступ до бази даних виконується за допомогою Entity Framework Core. Сервер реалізований у стилі RESTful, що забезпечує масштабовану та гнучку архітектуру. За допомогою RESTful API клієнти можуть взаємодіяти з сервером за допомогою стандартних методів HTTP, що полегшує інтеграцію з будь-якою платформою чи мовою програмування. Крім того, було використано Swagger для документування кінцевих точок API, щоб іншим розробникам було легше зрозуміти, як використовувати API без необхідності вручну створювати клієнтські додатки. Запущений сервер зображено на рисунку 4.5.

Щоб отримати текст із зображень етикеток продуктів харчування, було вирішено використати бібліотеку .NET OCR. Ця бібліотека надає набір інструментів, які дозволяють розпізнавати текст із зображень з високою точністю.

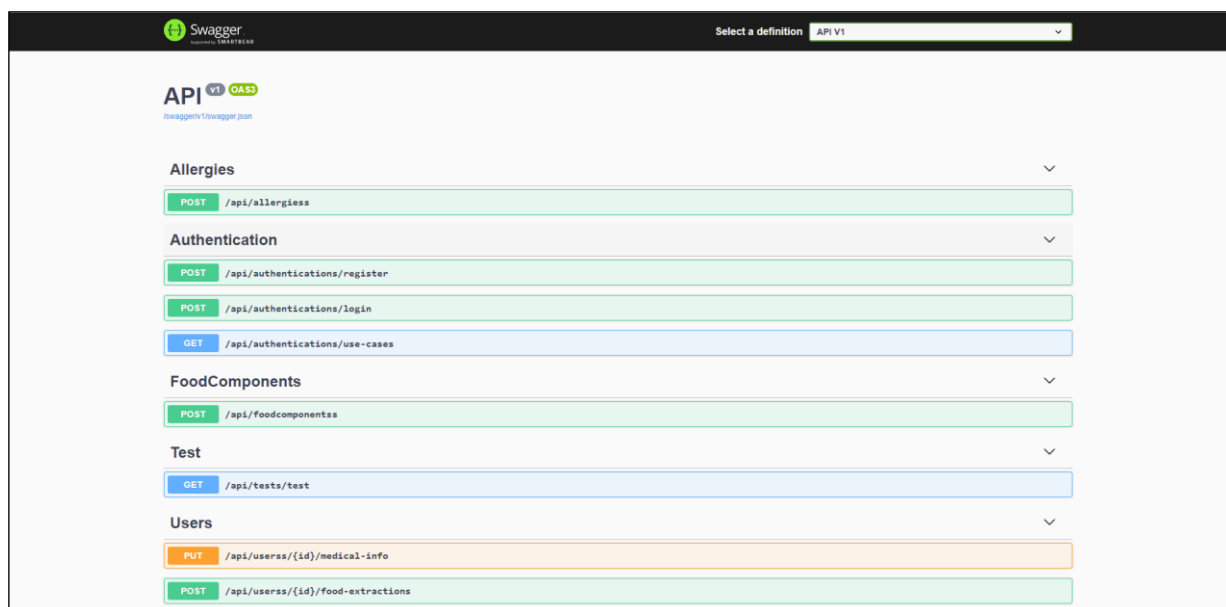


Рисунок 4.5 – Документація Swagger

Після цього ви використовували механізм OCR для розпізнавання тексту на зображенні. Використовуючи бібліотеку .NET OCR було отримано змогу швидко й точно діставати релевантну інформацію із зображень етикеток, що спростило відстеження поживного вмісту продуктів, які споживає користувач [10].

Під час розробки такої системи довелось створити багато енамів, які описують обраний домен. Одним з прикладів може стати описання способу ведення життя користувачем, який він сам може налаштувати. Всі ці енами активно використовуються під час опису норм вживання продуктів.

```
public enum LifestyleType
{
    Sedentary,
    LightlyActive,
    ModeratelyActive,
    VeryActive
}
```

В системі існують різні етапи, через це наявна велика кількість коду. Одним з основних прикладів можна виділити метод, який отримавши весь текст з .NET OCR оброблює його та знаходить необхідні компоненти харчового продукту.

```
public async Task<ExtractedFoodComponent[]>
ExtractFoodComponentsAsync(string[] parsedTexts)
{
    var joinedWords = string.Join(" ", parsedTexts).ToLower();
    var distinctWords = new HashSet<string>(joinedWords.Split(' '));

    var foodComponents = await _dbContext.FoodComponents
        .Where(fc => distinctWords.Contains(fc.Name.ToLower()))
        .ToListAsync();

    var extractedComponents = foodComponents.Select(fc => new
ExtractedFoodComponent
    {
        Id = fc.Id,
        Name = fc.Name
    }).ToArray();

    return extractedComponents;
}
```

Ще одним прикладом може стати метод, який аналізує історію прийому їжі користувачем та з врахуванням фізіологічних даних та норм прийому вираховує ліміти за вказаний проміжок часу.

```
public async Task<Dictionary<Guid, double>> CalculateRemainingAmounts(Guid
userId)
{
    var userMedicalInfo = await
_dbContext.UserMedicalInfos.FirstOrDefaultAsync(umi => umi.UserId ==
userId);
    var foodConsumptions = await _dbContext.FoodConsumptions
        .Include(fc => fc.FoodComponent)
        .Where(fc => fc.UserId == userId)
        .ToListAsync();

    var remainingAmounts = new Dictionary<Guid, double>();
    foreach (var consumption in foodConsumptions)
    {
        var applicableRules = _dbContext.RuleApplicabilities
            .Include(ra => ra.HealthyAmountRule)
            .Where(ra => ra.HealthyAmountRule.FoodComponentId ==
consumption.FoodComponentId
                && ra.IsApplicable(userMedicalInfo))
            .ToListAsync();

        foreach (var rule in applicableRules)
```

```

    {
        if
(!remainingAmounts.ContainsKey(rule.HealthyAmountRule.FoodComponentId))
        {
            remainingAmounts[rule.HealthyAmountRule.FoodComponentId] =
0;
        }

        var remaining =
CalculateRemainingAmount(rule.HealthyAmountRule, rule.Type,
consumption.Amount,
consumption.MeasurementPrefix, consumption.MeasurementUnit,
userMedicalInfo);

        remainingAmounts[rule.HealthyAmountRule.FoodComponentId] +=
remaining;
    }
}

return remainingAmounts;
}

```

Загалом, весь код написаний з чіткою та структурованою архітектурою, яка полегшує розуміння та використання. Завдяки чітко визначеним класам, інтерфейсам і методам кодової бази легко орієнтуватися та змінювати. Крім того, кодова база зберігається чистою та не містить непотрібної складності, що полегшує її підтримку та розширення. Загалом, добре структурована кодова база забезпечує міцну основу для розробки системи рекомендацій щодо харчування, що дозволяє легко модифікувати та розширювати її для задоволення майбутніх потреб.

5 МОЖЛИВОСТІ ВИКОРИСТАННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

Тепер, після того як етап дослідження та проектування даної рекомендаційної програмної системи для здорового харчування завершено, настав час визначити, які можливості наявні, щоб використати отримані результати. Аналіз і дослідження тих рішень для здорового харчування, що вже існують на ринку, дають цінну інформацію та достатньо глибоке розуміння того, які функції та особливості потрібні для такої системи. Етап моделювання та проектування допоміг нам створити абстрактну модель, яка може надавати рекомендації щодо харчування на основі даних користувача та інформації про продукти харчування. Нарешті, етап впровадження дозволив нам застосувати цю модель на практиці та створити діючий прототип програмної системи. Цю інформацію можна використовувати для розробки та впровадження власної системи, гарантуючи, що будуватися буде те, що вже добре працює, і можна уникнути будь-яких прихованих небезпек чи перешкоди, які траплялися в попередніх рішеннях. Ці знання можна використати для створення більш персоналізованого досвіду для користувачів даної системи, дозволяючи їм встановлювати власні цілі та отримувати індивідуальні рекомендації, які допоможуть їм досягти мету.

Однією з ключових можливостей даної системи є надання персоналізованих рекомендацій щодо харчування на основі даних користувачів. Користувачі можуть вводити інформацію про свій вік, вагу, зріст, рівень активності та інші фактори, які можуть вплинути на їхні потреби в харчуванні. Потім система використовуватиме ці дані для створення персоналізованого плану, який враховує щоденне споживання відповідних компонентів харчових продуктів, а також будь-які обмеження, які вони можуть мати.

Система також може аналізувати харчові продукти, витягуючи інформацію з етикеток продуктів і фотографій. Користувачі можуть використовувати камеру свого мобільного пристрою для аналізу їжі, яку вони їдять, або етикеток продуктів, а система використовуватиме методи обробки зображень, щоб отримати відповідну інформацію, таку як харчовий вміст, алергени, термін придатності та ін. Потім цю інформацію можна використати для розрахунку

залишкових обмежень для різних поживних речовин і попередження користувача, якщо він перевищив свої обмеження або збирається споживати щось, що може бути для нього достатньо шкідливим.

Ще однією важливою особливістю даної системи є можливість відстежувати та аналізувати прогрес користувача з часом. Користувачі можуть реєструвати споживання їжі, а система відстежуватиме їхній прогрес у дотриманні норм щодо здорового харчування.

Абстрактна модель програмної системи є ключовим інструментом, який керуватиме розробкою даної системи. Розбиваючи систему на різні компоненти та визначаючи, як вони взаємодіють один з одним, можна гарантувати, що кінцевий продукт буде функціональним і зручним для користувача. Крім того, абстрактну модель можна використовувати для визначення потенційних областей вдосконалення або розширення системи в майбутньому.

Загалом, дана рекомендаційна програмна система пропонує широкий спектр можливостей і функцій, щоб допомогти користувачам досягти своїх цілей щодо підтримки здорового харчування. Використовуючи результати даних досліджень, проектувань та готової реалізації в коді, було створено ефективний, зручний та гнучкий прототип системи. Незалежно від того, чи хочуть користувачі покращити загальний стан здоров'я, боротися з хронічним захворюванням або просто підтримувати здоровий спосіб життя, дана система може надати інструменти та ресурси, необхідні для досягнення успіху.

ВИСНОВКИ

В рамках даної роботи було сформульовано мету та науково-технічну задачу, проаналізовано стан вирішення проблеми за матеріалами вітчизняних і зарубіжних публікацій та патентно-інформаційних досліджень, проведено виявлення протиріч відомих теоретичних або експериментальних результатів з метою визначення наукового і технічного рівня вирішеності та обґрунтування мети дослідження. Після цього було поставлено задачу, проаналізовані методи дослідження, які застосовувалися під час вирішення задачі, розроблено методики дослідження, його апаратного забезпечення. Був проведений науковий аналіз і узагальнено фактичний матеріал, який використовувався у процесі дослідження. Отримано нові результати, що мають теоретичне та прикладне значення.

За результатами дослідження були проаналізовані вже працюючі на ринку системи для здорового харчування, які можливості вони пропонують користувачам та які рекомендації надають. Після цього були виявлені недоліки в існуючих алгоритмах рекомендаційних систем для здорового харчування. Було визначено важливість кожного параметра серед алгоритмів, виміряно їх швидкість виконання та проаналізовано результати, які вони надають. Було проведено порівняння між алгоритмами, які вже використовуються в даній сфері. Також було об'єднано та доповнено переваги вже готових алгоритмів, та таким чином створено зовсім новий алгоритм, який потенційно може витіснити своїх попередників через свою більш різноманітну функціональність.

Важливість даного дослідження є доволі високою, оскільки харчування є одним з головних чинників, які впливають на здоров'я людини, а в подальшому і на все її життя. Контроль допустимих норм для різних видів харчових продуктів є необхідним в сучасному житті, оскільки це попереджує появу різних хвороб та сприяє покращенню поточного стану. Удосконалення існуючих та створення більш точних та коректних рекомендаційних алгоритмів для здорового харчування майже повністю вирішують проблеми в області здорового прийому їжі, оскільки використання таких алгоритмів автоматизує обчислення необхідних норм та має більш точні результати, ніж абстрактні поради.

Оскільки винайдений алгоритм сам по собі не є залежним від специфічної технології або мови програмування, тому його можливість реалізації дає найбільш гнучкий спектр використання. Він може бути прописаний на будь-якій мові програмування, включаючи його часткову оптимізацію у відповідності з обраною технологією, та бути запущений на будь-якому пристрою, починаючи від серверу та фронтенду до смартфона або смарт годинника. Наявна цілком реалістична можливість створення бібліотеки або модулю (в залежності від технології) на певній мові програмування, а потім подальше повторне використання в різних додатках або системах. Але й наявні проблеми пов'язані зі складністю логіки даного алгоритму.

Обрана область дослідження є достатньо обширною для розгляду та має подальші напрямки розвитку. Можливо продовжити в тому ж напрямку та збільшувати кількість параметрів алгоритму, які будуть враховуватися під час надання рекомендацій щодо вживання їжі; або зосередитись на інших джерелах, з яких можна діставати необхідні дані щодо продуктів харчування для генерації рекомендацій користувачу; або ж працювати над збільшенням ефективності та продуктивності даного алгоритму. Також можливо спробувати поєднати алгоритм з більш детальними медичними даними людини, для якої створюються рекомендації

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Experts from Dole Food Company. Encyclopedia of Foods. A Guide to Healthy Nutrition / Experts from Dole Food Company, Experts from The Mayo Clinic, Experts from UCLA Center for H., 2002. – 529 с.
2. Tessmer K. A. Your nutrition solution to a healthy gut: a meal-based plan to help prevent and treat constipation, diverticulitis, ulcers, and other common digestive problems / Kimberly Tessmer., 2015. – 223 с.
3. Recommender Systems: Algorithms and Applications / P. K.Pavan, S. Vairachilai, P. Sirisha, N. M. Sachi., 2021. – 249 с.
4. Recommender Systems: An Introduction / D.Jannach, M. Zanker, A. Felfernig, G. Friedrich., 2010. – 353 с.
5. Khalid O. Big Data Recommender Systems - Volume 1 Algorithms, Architectures, Big Data, Security and Trust / O. Khalid, S. U. Khan, A. Y. Zomaya., 2019. – 368 с.
6. Frey M. Netflix Recommends: Algorithms, Film Choice, and the History of Taste / Mattias Frey., 2021. – 282 с.
7. Uchyigit G. Personalization Techniques And Recommender Systems / G. Uchyigit, M. Y. Ma., 2008. – 334 с. – (Series in Machine Perception and Artificial Intelligence).
8. Ricci F. Recommender Systems Handbook / F. Ricci, L. Rokach, B. Shapira., 2022. – 1061 с. – (3).
9. Smelyakov K. Object-Based Image Comparison Algorithm Development for Data Storage Management Systems / K. Smelyakov, O. Prokopenko, A. Chupryna., 2022. – 3171 с.
10. Effectiveness of modern text recognition solutions and tools for common data sources / K.Smelyakov, A. Chupryna, D. Darahan, S. Midina., 2021. – 2870 с.