

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЗАПОРІЗЬКА ПОЛІТЕХНІКА»

Факультет комп'ютерних наук та технологій
Кафедра комп'ютерних систем та мереж

Пояснювальна записка

до дипломного проекту (роботи)

магістра

(ступінь вищої освіти)

на тему СИСТЕМА ГЕНЕРАЦІЇ ТА МОНІТОРИНГУ
ПЕРСОНАЛЬНИХ ПЛАНІВ ХАРЧУВАННЯ

Виконав: студент 2 курсу, групи КНТ-514м
спеціальності _____

123 Комп'ютерна інженерія

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Комп'ютерні системи та мережі

МОТОШИН А.А.

(ПРІЗВИЩЕ та ініціали)

Керівник ІЛЬЯШЕНКО М.Б.

(ПРІЗВИЩЕ та ініціали)

Рецензент ЩЕРБАКОВ В.І.

(ПРІЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет Комп'ютерних наук і технологій
 Кафедра «Комп'ютерні системи та мережі»
 Ступінь вищої освіти магістерський
 Спеціальність 123 Комп'ютерна інженерія

(код і найменування)

Освітня програма (спеціалізація) Комп'ютерні системи та мережі
 (назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Зав. кафедри Кудерметов Р.К.

“15” жовтня 2025 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА

МОТОШИНА Андрія Андрійовича

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Система генерації та моніторингу персональних планів харчування

керівник проєкту (роботи) кан. техн. наук, доцент, ІЛЬЯШЕНКО Матвій Борисович
 (науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом вищого навчального закладу від “ 30 ” жовтня 2025 року № 491

2. Строк подання студентом проєкту (роботи) 23 грудня 2025 року

3. Вихідні дані до проєкту (роботи) наявні методи розрахунку енергетичних потреб організму, формули базального метаболізму Міффіна-Сан Жеора, рекомендації щодо розподілу макронутрієнтів для різних цілей, вимоги до точності балансування макронутрієнтів $\pm 5\%$, час генерації плану не більше 10 секунд, база продуктів USDA FoodData Central понад 350 000 позицій

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1) Аналіз існуючих рішень у сфері планування харчування, дослідження методів розрахунку енергетичних потреб та балансу макронутрієнтів;

2) Розробка математичної моделі задачі формування плану харчування як задачі багатокритеріальної оптимізації;

3) Проектування клієнт-серверної архітектури мобільної системи;

4) Реалізація модулів реєстрації користувачів, генерації персональних планів, моніторингу споживання та аналітики;

5) Проведення експериментального дослідження ефективності системи.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	ПРІЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4	ІЛЛЯШЕНКО М.Б., доцент		
нормоконтроль	ЩЕРБАК Н.В., ст. викл.		

7. Дата видачі завдання 01.10.2025 р.**КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Аналіз технічного завдання, огляд літератури та Існуючих рішень у сфері планування харчування	05.10.2025 р.	
2	Дослідження методів розрахунку енергетичних потреб	10.10.2025 р.	
3	Розробка математичної моделі	15.10.2025 р.	
4	Проектування архітектури системи	25.11.2025 р.	
5	Реалізація і дослідження системи	30.11.2025 р.	
6	Оформлення отриманих результатів у ПЗ	05.12.2025 р.	
7	Оформлення графічного матеріалу	10.12.2025 р.	
8	Оформлення допоміжного матеріалу	15.12.2025 р.	

Студент _____ **Андрій МОТОШИН**
(підпис) (ім'я, ПРІЗВИЩЕ)Керівник проєкту (роботи) _____ **Матвій ІЛЛЯШЕНКО**
(підпис) (ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

ПЗ: 103 стор., 24 рис., 6 табл., 30 джерел.

БАЗАЛЬНИЙ МЕТАБОЛІЗМ, ДІЄТОЛОГІЯ, КАЛОРІЙНІСТЬ,
МАКРОНУТРИЄНТИ, МОБІЛЬНИЙ ЗАСТОСУНОК, МОНІТОРИНГ
ХАРЧУВАННЯ, NODE.JS, ОПТИМІЗАЦІЯ, ПЕРСОНАЛІЗАЦІЯ,
ПЛАНУВАННЯ ХАРЧУВАННЯ, REACT NATIVE

Об'єкт дослідження - процеси планування та моніторингу персонального харчування з використанням мобільних технологій.

Предмет дослідження - методи автоматизованої генерації збалансованих планів харчування з урахуванням індивідуальних параметрів користувачів.

Мета роботи - розробка мобільної системи для автоматизованої генерації персональних планів харчування та моніторингу досягнення цілей.

У першому розділі проведено аналіз існуючих рішень MyFitnessPal, Lifesum, Eat This Much та визначено методи формування планів харчування.

У другому розділі розроблено архітектуру системи на базі React Native-Node.js-PostgreSQL та алгоритм генерації планів з комбінованим підходом евристичних методів та локальної оптимізації.

Третій розділ присвячено реалізації модулів реєстрації користувачів, генерації планів з точністю $\pm 5\%$, моніторингу споживання та аналітики.

У четвертому розділі проведено експериментальне дослідження за участю 25 користувачів протягом 4 тижнів. Оцінка зручності за шкалою SUS склала 82.4 бала.

ABSTRACT

Explanatory note to the master's work: 103 p., 24 fig., 6 tabl., 30 sources.

MOBILE APPLICATION, MEAL PLANNING, PERSONALIZATION, MACRONUTRIENTS, CALORIC INTAKE, BASAL METABOLISM, OPTIMIZATION, REACT NATIVE, NODE.JS, POSTGRESQL

Research object - personal nutrition planning and monitoring processes using mobile technologies.

Research subject - methods for automated generation of balanced meal plans considering individual user parameters.

Research goal - development of mobile system for automated generation of personalized meal plans and goal achievement monitoring.

The first section analyzes existing solutions MyFitnessPal, Lifesum, Eat This Much and identifies meal planning methods.

The second section develops system architecture based on React Native-Node.js-PostgreSQL and plan generation algorithm with combined heuristic and local optimization approach.

The third section implements user registration modules, plan generation with $\pm 5\%$ accuracy, consumption monitoring and analytics.

The fourth section conducts experimental research with 25 users over 4 weeks. System Usability Scale evaluation reached 82.4 points.

ЗМІСТ

Вступ.....	7
1 Аналітична частина.....	11
1.1 Огляд джерел та методів збору даних.....	11
1.2 Аналіз існуючих рішень та технологій	22
1.4 Постановка задачі дослідження.....	39
2 Проектування системи генерації планів харчування.....	43
2.1 Побудова концептуальної моделі системи	43
2.2 Проектування архітектури системи.....	49
2.3 Розробка алгоритмічного забезпечення.....	54
2.4 Вибір технологічного стеку	56
3 Ралізація мобільної системи моніторингу харчування	61
3.1 Апаратна складова системи.....	61
3.2 Реалізація модуля реєстрації та профілів користувачів	64
3.3 Реалізація модуля генерації планів харчування.....	66
3.4 Реалізація модуля моніторингу та аналітики	70
3.5 Інтеграція з базами харчових продуктів	73
4 Експериментальне дослідження системи	83
4.1 Методика експериментального дослідження.....	83
4.2 Результати оцінки точності генерації планів	85
4.3 Аналіз дотримання протоколу користувачами	87
4.5 Оцінка зручності використання системи	92
Висновки	94
Перелік джерел посилання	95
Додаток А Слайди презентації.....	98

ВСТУП

Актуальність теми дослідження обумовлена зростанням проблем надмірної ваги та ожиріння у сучасному суспільстві, а також підвищенням інтересу населення до здорового способу життя та збалансованого харчування. За даними Всесвітньої організації охорони здоров'я, у 2022 році понад 2,5 мільярда дорослих людей мали надмірну вагу, з яких понад 890 мільйонів страждали на ожиріння [1]. Водночас доступ до професійних дієтологів та нутриціологів залишається обмеженим через високу вартість послуг, нестачу спеціалістів у регіонах та часові обмеження для відвідування консультацій. Це створює значний розрив між потребою населення у кваліфікованій допомозі з питань харчування та реальними можливостями її отримання.

Традиційні методи планування харчування базуються на загальних рекомендаціях щодо калорійності та співвідношення макронутрієнтів, які не враховують індивідуальні особливості організму, харчові вподобання, алергії та медичні обмеження конкретної людини. Складання персоналізованого раціону вимагає значних зусиль та спеціалізованих знань у галузі нутриціології, що робить цей процес недоступним для більшості людей. Навіть за наявності загальних рекомендацій користувачі часто не мають часу або достатніх знань для самостійного формування збалансованого меню на тиждень з урахуванням усіх необхідних параметрів. Цифрові системи генерації планів харчування можуть заповнити цю прогалину, забезпечуючи автоматизоване створення збалансованих раціонів з урахуванням індивідуальних параметрів користувача, його цілей та обмежень.

Існуючі мобільні додатки для планування харчування часто обмежуються простим підрахунком калорій або пропонують статичні меню без можливості адаптації під конкретного користувача. Більшість популярних рішень, таких як MyFitnessPal, зосереджені на трекінгу споживаної їжі, але не надають інструментів для автоматичного формування персоналізованих планів. Інші

платформи, наприклад Lifesum, використовують попередньо створені шаблони дієт без врахування індивідуальних особливостей метаболізму, вподобань та доступності продуктів у конкретному регіоні. Відсутність комплексного підходу до формування персоналізованих планів харчування створює потребу у розробці інтелектуальної системи, що поєднує наукові принципи дієтології з сучасними технологіями мобільної розробки та алгоритмами оптимізації.

Метою дипломної роботи є розробка мобільної системи для автоматизованої генерації персональних планів харчування з урахуванням індивідуальних параметрів користувача, його цілей та обмежень, а також забезпечення моніторингу дотримання раціону та аналізу досягнення поставлених цілей. Система має надавати зручний інтерфейс для щоденного відстеження споживання їжі, генерувати збалансовані меню з наявних продуктів, розраховувати оптимальні порції та забезпечувати детальну аналітику динаміки дотримання плану харчування. Важливою складовою є можливість врахування харчових обмежень, алергій та індивідуальних вподобань користувачів для забезпечення максимальної персоналізації рекомендацій.

Для досягнення поставленої мети необхідно вирішити низку завдань різного характеру. Перш за все, потрібно провести детальний аналіз існуючих рішень у сфері планування харчування та виявити їхні недоліки й переваги для формування вимог до розроблюваної системи. Наступним кроком є розробка математичної моделі задачі формування плану харчування як задачі оптимізації з множинними обмеженнями, що дозволить забезпечити збалансованість раціону за всіма ключовими параметрами. Необхідно спроектувати клієнт-серверну архітектуру системи з чітким розділенням відповідальності між мобільним застосунком та серверною частиною. Окремим завданням є реалізація модулів реєстрації користувачів, генерації персональних планів харчування, моніторингу фактичного споживання їжі та аналітики досягнутих результатів. Завершальним етапом має стати проведення експериментального дослідження для оцінки якості генерованих планів харчування та ефективності системи у досягненні цільових результатів користувачів.

Об'єктом дослідження є процеси планування та моніторингу персонального харчування з використанням мобільних технологій та алгоритмів оптимізації. Предметом дослідження виступають методи та алгоритми автоматизованої генерації збалансованих планів харчування з урахуванням індивідуальних параметрів користувачів, а також програмні засоби реалізації мобільних систем управління харчуванням.

Наукова новизна роботи полягає у розробці комбінованого підходу до генерації персональних планів харчування, що поєднує евристичні методи для швидкого формування початкового допустимого розв'язку з методами локального пошуку для оптимізації балансу макронутрієнтів. Запропоновано математичну модель задачі планування харчування як задачі багатокритеріальної оптимізації з обмеженнями на калорійність, співвідношення макронутрієнтів, виключення алергенів та забезпечення різноманітності раціону. Розроблено архітектуру мобільної системи з модульною структурою, що забезпечує гнучкість у розширенні функціональності та інтеграції з зовнішніми джерелами даних.

Практичне значення отриманих результатів полягає у створенні функціональної мобільної системи, яка може використовуватися широким колом користувачів для самостійного планування та контролю харчування без необхідності звертання до професійних дієтологів. Розроблена система забезпечує автоматизацію процесу формування збалансованих раціонів, знижує часові витрати на планування меню та підвищує усвідомленість користувачів щодо їхніх харчових звичок. Система може бути адаптована для використання у фітнес-центрах, клініках дієтології, санаторно-курортних закладах та інших установах, де здійснюється консультування з питань харчування.

Методи дослідження включають аналіз літературних джерел та існуючих програмних рішень у сфері планування харчування, математичне моделювання задачі оптимізації з множинними обмеженнями, алгоритмічне проєктування комбінованих методів генерації планів, об'єктно-орієнтоване проєктування архітектури програмної системи, експериментальне дослідження ефективності

розробленої системи за участю тестової групи користувачів, а також статистичний аналіз отриманих результатів для оцінки якості генерованих планів та досягнення користувачами поставлених цілей.

Апробація результатів роботи здійснена шляхом проведення експериментального дослідження за участю двадцяти п'яти користувачів-добровольців протягом чотирьох тижнів. Учасники представляли три цільові групи з різними завданнями: схуднення, підтримка ваги та набір м'язової маси. Результати експерименту підтвердили ефективність розробленої системи у досягненні користувачами їхніх цілей та продемонстрували високий рівень задоволеності функціональністю та зручністю використання додатку.

1 АНАЛІТИЧНА ЧАСТИНА

1.1 Огляд джерел та методів збору даних

Визначення індивідуальних потреб організму в енергії та поживних речовинах є фундаментальною основою для створення персоналізованих планів харчування. Існує декілька підходів до оцінки енергетичних потреб людини, що базуються на різних параметрах та показниках. Найбільш поширеним та науково обґрунтованим методом є розрахунок базального метаболізму, який відображає мінімальну кількість енергії, необхідну організму для підтримання життєво важливих функцій у стані спокою. Цей показник залежить від багатьох факторів, включаючи вік, стать, масу тіла, зріст, склад тіла та генетичні особливості метаболізму. Розуміння механізмів енергетичного обміну та факторів, що впливають на потреби організму, дозволяє створювати точні та ефективні рекомендації щодо харчування для досягнення різноманітних цілей.

Основними параметрами для визначення індивідуальних потреб користувача у енергії є розрахунок базального метаболізму за формулою Міффіліна-Сан Жеора, яка враховує вік, стать, вагу та зріст людини. Ця формула була розроблена у 1990 році американськими дослідниками на основі непрямой калориметрії у великій групі здорових осіб та показала вищу точність порівняно з раніше використовуваною формулою Гарріса-Бенедикта, особливо для осіб з надмірною вагою. Базальний метаболізм являє собою кількість енергії, яку організм витрачає у стані спокою для підтримання життєво важливих функцій, таких як дихання з частотою від дванадцяти до двадцяти разів на хвилину, кровообіг з серцевим викидом близько п'яти літрів на хвилину, підтримання температури тіла на рівні тридцять шість і шість десятих градуса Цельсія, синтез білків зі швидкістю близько триста грамів на добу, робота нервової системи та інших базових процесів.

Формула для чоловіків має вигляд $BMR = 10 \times \text{вага(кг)} + 6.25 \times \text{зріст(см)} - 5 \times \text{вік(роки)} + 5$, тоді як для жінок константа становить мінус сто шістдесят

один замість плюс п'ять [2]. Така різниця обумовлена фізіологічними відмінностями між статями, зокрема різним співвідношенням м'язової та жирової тканини у тілі, де чоловіки зазвичай мають від сорока до п'ятдесяти відсотків м'язової маси, а жінки від тридцяти шести до сорока двох відсотків. Гормональний фон також суттєво впливає на метаболізм, оскільки тестостерон стимулює анаболічні процеси та підвищує базальний метаболізм, тоді як естроген сприяє накопиченню жирової тканини у специфічних зонах. Особливості метаболізму включають різну активність ферментів ліполізу, термогенезу та окислення жирних кислот між статями.

Формула Міффіна-Сан Жеора була валідована на основі вибірки з понад п'ятисот здорових людей різних етнічних груп, включаючи європеїдів, негроїдів, монголоїдів та представників змішаних популяцій, віком від дев'ятнадцяти до сімдесяти восьми років з індексом маси тіла від вісімнадцяти до сорока кілограмів на метр квадратний. Дослідження показали, що середня похибка формули становить близько десяти відсотків для загальної популяції, що є прийнятним результатом для клінічної практики та планування харчування. Однак слід зазначити, що формула може давати значні похибки, що перевищують п'ятнадцять-двадцять відсотків, для осіб з екстремальними значеннями м'язової маси, таких як професійні бодібілдери з м'язовою масою понад п'ятдесят відсотків або атлети силових видів спорту.

М'язова тканина споживає більше енергії у стані спокою порівняно з жировою, приблизно тринадцять кілокалорій на кілограм на добу проти п'ять кілокалорій відповідно, що пояснює необхідність корекції розрахунків для осіб з нетиповим складом тіла. Також формула може бути менш точною для людей похилого віку старше сімдесяти років через специфічні вікові зміни метаболізму, включаючи зниження м'язової маси на один-два відсотки на рік після шістдесяти років, зменшення активності мітохондрій та зниження чутливості до гормонів щитоподібної залози. Для осіб з метаболічними порушеннями, такими як гіпотиреоз з зниженням базального метаболізму на п'ятнадцять-тридцять

відсотків або гіпертиреоз з підвищенням на двадцять-п'ятдесят відсотків, необхідні специфічні корекції формули.

Базальний метаболізм множиться на коефіцієнт фізичної активності для отримання загальної добової потреби в енергії, яка враховує енергетичні витрати на рух, фізичні вправи та повсякденну діяльність. Цей підхід базується на припущенні, що загальні енергетичні витрати організму складаються з трьох основних компонентів з різними відносними внесками. Перший компонент - базальний метаболізм, який становить приблизно шістдесят-сімдесят відсотків від загальних енергетичних витрат для більшості людей з помірною фізичною активністю. Другий компонент - термічний ефект їжі, який становить близько восьми-десяти відсотків та відображає енергію, необхідну для перетравлювання, всмоктування, транспорту та метаболізму поживних речовин, причому білки мають найвищий термічний ефект близько двадцяти п'яти-тридцяти відсотків від їхньої енергетичної цінності.

Третій компонент - енергія, витрачена на фізичну активність, яка становить від п'ятнадцяти до тридцяти відсотків залежно від способу життя та може досягати сорока-п'ятдесяти відсотків у професійних атлетів. Коефіцієнти активності варіюються від 1.2 для малорухливого способу життя без регулярних фізичних вправ, коли людина переважно сидить або лежить протягом дня з менше п'яти тисяч кроків на добу, проводить понад вісім годин у сидячому положенні та не займається організованими формами фізичної активності. Такий спосіб життя характерний для офісних працівників, програмістів, операторів колл-центрів та інших професій з мінімальною фізичною активністю. Коефіцієнт 1.9 застосовується для дуже активних людей з інтенсивними тренуваннями майже щодня або фізично важкою роботою, такою як будівництво з переміщенням вантажів масою понад двадцять кілограмів, сільськогосподарська праця з ручною обробкою землі або професійний спорт з тренуваннями два рази на день.

Проміжні значення коефіцієнтів активності детально відображають різні рівні фізичної активності населення з урахуванням як організованих тренувань,

так і повсякденної активності. Коефіцієнт 1.375 відповідає легкій активності, що включає вправи один-три рази на тиждень тривалістю від тридцяти до шістдесяти хвилин, такі як прогулянки швидким кроком зі швидкістю п'ять-шість кілометрів на годину, легкі заняття йогою без інтенсивних силових асан, плавання невисокої інтенсивності з відпочинком між доріжками або їзда на велосипеді в спокійному темпі по рівній місцевості. Також до цієї категорії належать люди, які пішки долають відстань два-три кілометри на день для доїзду до роботи або у побутових цілях.

Коефіцієнт 1.55 використовується для помірної активності з вправами три-п'ять разів на тиждень тривалістю від сорока п'яти до дев'яноста хвилин, що може включати біг підтюпцем зі швидкістю сім-дев'ять кілометрів на годину та пульсом сто тридцять-сто п'ятдесят ударів на хвилину, заняття у тренажерному залі середньої інтенсивності з комбінацією кардіо та силових вправ, активні ігрові види спорту такі як футбол, баскетбол, теніс з ракеткою або танці з аеробним навантаженням. Такий рівень активності характерний для людей, які усвідомлено підтримують фізичну форму та регулярно відвідують спортивні заняття, але не займаються професійним спортом.

Коефіцієнт 1.725 застосовується для високої активності з вправами шість-сім разів на тиждень тривалістю від шістдесяти до ста двадцяти хвилин, наприклад, інтенсивні силові тренування з використанням вільних ваг та базових вправ, таких як присідання, станова тяга та жим лежачи, бігові тренування для підготовки до марафону з тижневим об'ємом від сорока до вісімдесяти кілометрів, заняття кросфітом з високоінтенсивними функціональними тренуваннями або інші високоінтенсивні види активності, що включають інтервальні тренування з пульсом понад вісімдесят відсотків від максимального. До цієї категорії також належать люди, які поєднують регулярні інтенсивні тренування з активною фізичною роботою або займаються аматорським спортом на конкурентному рівні з участю у змаганнях.

Вибір правильного коефіцієнта активності є критично важливим для точного розрахунку калорійності раціону, оскільки помилка у визначенні рівня

активності безпосередньо трансформується у похибку енергетичного балансу з відповідними наслідками. Недооцінка рівня активності, наприклад, вибір коефіцієнта 1.375 замість коректного 1.55, призводить до занадто низької розрахункової калорійності з дефіцитом близько триста-чотириста кілокалорій на день, що за місяць може призвести до непланованої втрати близько двох кілограмів ваги. Така ситуація небезпечна можливою втратою м'язової маси, зниженням рівня енергії, погіршенням спортивних показників, порушенням менструального циклу у жінок через недостатнє споживання енергії та гормональні зміни.

Переоцінка рівня активності, навпаки, призводить до надмірного споживання калорій та небажаного набору ваги переважно у вигляді жирової тканини. Якщо людина з фактично помірною активністю обирає коефіцієнт 1.725, замість коректного 1.55, вона споживатиме надлишкові триста-чотириста кілокалорій на день, що за місяць може призвести до набору близько двох кілограмів переважно жирової маси. Систематична переоцінка свого рівня активності є однією з найпоширеніших причин відсутності прогресу у людей, які намагаються схуднути, оскільки вони не розуміють, чому дотримання розрахункової калорійності не призводить до очікуваних результатів.

Залежність загальної денної потреби в енергії від рівня фізичної активності для типового чоловіка віком тридцять років, вагою вісімдесят кілограмів та зростом сто сімдесят п'ять сантиметрів проілюстровано на рисунку 1.1. Базальний метаболізм для такої людини становить приблизно тисячу вісімсот кілокалорій на добу.

Як видно з рисунка 1.1, різниця у денній потребі в калоріях між малорухливим та дуже активним способом життя може досягати понад тисячу двісті шістдесят кілокалорій, що становить близько п'ятдесяти восьми відсотків від базального метаболізму або приблизно тридцять сім відсотків від загальних енергетичних витрат малорухливої людини. Це підкреслює важливість точного визначення рівня фізичної активності для коректного планування раціону харчування. Навіть перехід з одного рівня активності на суміжний змінює денну

потребу в енергії на триста-чотириста кілокалорій, що еквівалентно енергетичній цінності повноцінного прийому їжі або приблизно п'ятнадцяти відсоткам від загальної калорійності.

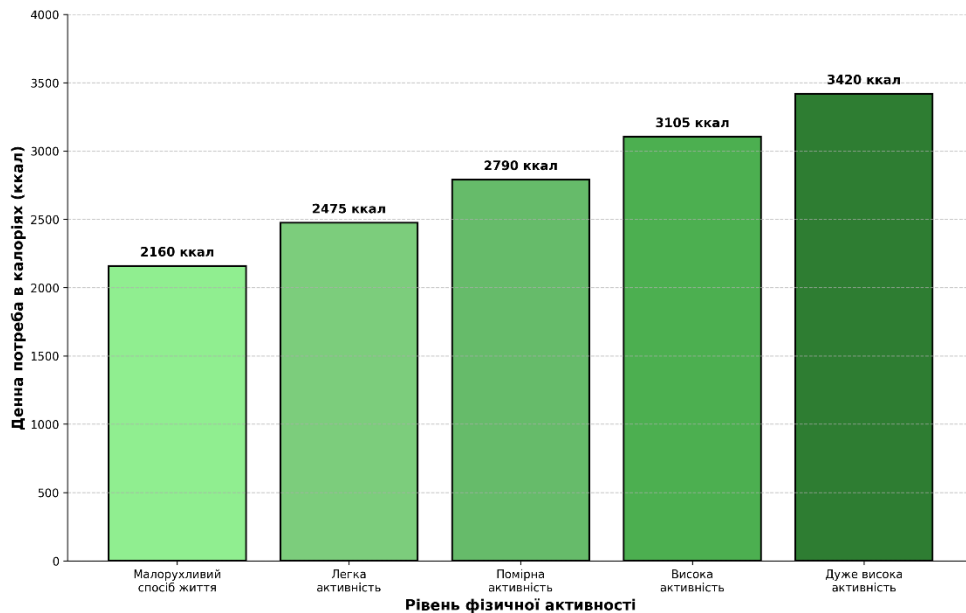


Рисунок 1.1 - Залежність денної потреби в калоріях від рівня активності

Баланс макронутрієнтів у раціоні відіграє критичну роль для досягнення різних цілей харчування та підтримання оптимального здоров'я організму. Макронутрієнти включають білки, жири та вуглеводи, кожен з яких виконує специфічні функції в організмі, має різну енергетичну цінність та по-різному впливає на метаболічні процеси. Розуміння ролі кожного макронутрієнта, їхніх оптимальних співвідношень для різних цілей та механізмів їхнього впливу на організм є ключовим для ефективного планування харчування та досягнення бажаних результатів без шкоди для здоров'я.

Білки є основним будівельним матеріалом для тканин організму та необхідні для безлічі життєво важливих процесів на клітинному та системному рівнях. Вони складаються з амінокислот, з'єднаних пептидними зв'язками у специфічні послідовності, що визначають їхню тривимірну структуру та функціональні властивості. З двадцяти амінокислот, що входять до складу білків людського організму, дев'ять є незамінними та мають надходити з їжею, оскільки

організм не може їх синтезувати самотійно або синтезує у недостатній кількості. До незамінних амінокислот належать лейцин, ізолейцин, валін, які особливо важливі для синтезу м'язових білків, лізин, метіонін, треонін, триптофан, фенілаланін та гістидін.

Білки необхідні для побудови та постійного відновлення м'язових тканин, що особливо важливо під час силових тренувань, коли м'язеві волокна зазнають мікропошкоджень та потребують амінокислот для репарації та гіпертрофії. Процес м'язевого синтезу стимулюється фізичним навантаженням та споживанням білка, причому оптимальна доза становить близько двадцять-сорок грамів високоякісного білка на прийом їжі. Сполучні тканини, включаючи сухожилля, зв'язки та хрящі, також складаються переважно з білка колагену та потребують достатнього надходження амінокислот, особливо гліцину, проліну та гідроксипроліну. Епітеліальні тканини шкіри, слизових оболонок та внутрішніх органів мають високу швидкість оновлення, наприклад, клітини кишкового епітелію оновлюються кожні три-п'ять днів, що вимагає постійного постачання амінокислот.

Синтез ферментів, які каталізують практично всі біохімічні реакції в організмі, від перетравлювання їжі до синтезу ДНК, також залежить від достатнього надходження білка. В організмі людини функціонує понад п'ять тисяч різних ферментів, кожен з яких має специфічну структуру та функцію. Виробництво гормонів білкової природи, таких як інсулін, що регулює рівень глюкози в крові, гормон росту, що стимулює ріст та відновлення тканин, глюкагон та багато інших, безпосередньо залежить від доступності амінокислот. Підтримання імунної функції через продукцію антитіл, що захищають організм від патогенів, також вимагає достатнього споживання білка, оскільки імуноглобуліни є складними білковими молекулами.

Транспорт речовин у крові здійснюється спеціалізованими білками-переносниками, наприклад, гемоглобін транспортує кисень від легень до тканин та вуглекислий газ у зворотному напрямку, альбумін переносить жирні кислоти, гормони та багато інших речовин, а трансферин забезпечує транспорт заліза.

Регуляція водного балансу також залежить від білків плазми крові, що створюють онкотичний тиск та запобігають надмірному виходу рідини з судинного русла у тканини. Один грам білка забезпечує чотири кілокалорії енергії, хоча використання білків як джерела енергії є менш ефективним та небажаним порівняно з вуглеводами або жирами через високі енергетичні витрати на їхнє перетравлювання, дезамінування та перетворення у глюкозу або кетонові тіла.

Жири є найбільш концентрованим джерелом енергії серед макронутрієнтів, надаючи дев'ять кілокалорій на грам, що більш ніж удвічі перевищує енергетичну щільність білків і вуглеводів з їхніми чотирма кілокалоріями на грам. Така висока енергетична щільність робить жири ефективним способом зберігання енергії в організмі у формі жирової тканини, причому один кілограм чистого жиру містить близько дев'яти тисяч кілокалорій енергії. Проте роль жирів у організмі далеко виходить за межі простого енергопостачання та включає множину структурних і регуляторних функцій, що робить їх абсолютно необхідними для нормального функціонування організму.

Жири необхідні для засвоєння жиророзчинних вітамінів А, D, Е та К, без яких ці важливі мікронутрієнти не можуть бути ефективно абсорбовані в кишечнику навіть за умови їхньої достатньої кількості у їжі. Вітамін А необхідний для зору, імунної функції та здоров'я шкіри. Вітамін D регулює обмін кальцію та фосфору, підтримує здоров'я кісток та має імуномодулюючі властивості. Вітамін Е діє як потужний антиоксидант, захищаючи клітинні мембрани від окислювального пошкодження. Вітамін К необхідний для згортання крові та метаболізму кісткової тканини. Мінімальна кількість жиру в раціоні для адекватного засвоєння цих вітамінів становить близько десяти-п'ятнадцяти грамів на прийом їжі.

Жири є основним компонентом клітинних мембран, складаючи фосфоліпідний бішар, що забезпечує структурну цілісність клітин та регулює проникність для різних речовин. Склад жирних кислот у мембранах впливає на їхню гнучкість, плинність та функціональні властивості, що критично важливо

для роботи мембранних білків, рецепторів та іонних каналів. Жири служать субстратом для синтезу гормонів стероїдної природи, включаючи статеві гормони тестостерон та естроген, що регулюють репродуктивну функцію, розвиток вторинних статевих ознак та метаболізм, кортизол, що відповідає за стресову відповідь та регуляцію імунітету, а також альдостерон, що контролює водно-сольовий баланс.

Синтез простагландинів, тромбоксанів та лейкотрієнів з арахідонової кислоти забезпечує регуляцію запальних процесів, згортання крові, тонуусу гладких м'язів та багатьох інших фізіологічних функцій. Підшкірний жир забезпечує термоізоляцію організму, зменшуючи втрати тепла та підтримуючи стабільну температуру тіла навіть за несприятливих зовнішніх умов. Вісцеральний жир, що оточує внутрішні органи, забезпечує механічний захист від ударів та струсів, хоча його надлишкове накопичення асоціюється з метаболічними порушеннями та підвищеним ризиком серцево-судинних захворювань.

Незамінні жирні кислоти омега-3, що включають альфа-ліноленову, ейкозапентаєнову та докозагексаєнову кислоти, та омега-6, зокрема лінолеву та арахідонову кислоти, не можуть синтезуватися організмом людини через відсутність необхідних ферментів десатураз та мають надходити з їжею для забезпечення нормального функціонування. Омега-3 жирні кислоти виконують критично важливі функції у роботі мозку, складаючи значну частину структурних ліпідів нейронів, особливо у сірій речовині кори та сітківці ока. Вони також мають протизапальні властивості, знижують рівень тригліцеридів у крові, покращують функцію ендотелію судин та знижують ризик серцево-судинних захворювань.

Вуглеводи служать основним та найбільш швидкодоступним джерелом енергії для організму, надаючи чотири кілокалорії на грам. Особливо важливими вони є для функціонування центральної нервової системи, оскільки мозок переважно використовує глюкозу як джерело енергії через обмежену здатність окислювати жирні кислоти, споживаючи близько ста двадцяти грамів глюкози

або п'ятсот кілокалорій на добу, що становить приблизно двадцять відсотків від базального метаболізму дорослої людини. Нейрони надзвичайно чутливі до дефіциту глюкози, і навіть короточасна гіпоглікемія з рівнем глюкози нижче трьох мілімолів на літр може призвести до порушення когнітивних функцій, сплутаності свідомості та у тяжких випадках втрати свідомості.

М'язи також активно використовують вуглеводи, особливо під час інтенсивних фізичних навантажень середньої та високої інтенсивності з пульсом понад сто сорок ударів на хвилину, коли анаеробний гліколіз забезпечує швидке вивільнення енергії з м'язевого глікогену без потреби у кисні. Запаси глікогену в м'язах становлять близько триста-п'ятсот грамів або тисячу двісті-дві тисячі кілокалорій, що достатньо для забезпечення енергією інтенсивного тренування тривалістю від шістдесяти до дев'яноста хвилин. Печінковий глікоген у кількості близько ста грамів або чотирьохсот кілокалорій виконує роль буфера для підтримання стабільного рівня глюкози в крові між прийомами їжі, особливо важливого для забезпечення роботи мозку під час нічного голодування.

Вуглеводи можна розділити на прості, або цукри, що включають моносахариди глюкозу, фруктозу, галактозу та дисахариди сахарозу, лактозу, мальтозу, які швидко всмоктуються в тонкому кишечнику протягом п'ятнадцяти-тридцяти хвилин та викликають різке підвищення рівня глюкози в крові з піком через тридцять-шістдесят хвилин після споживання. Така швидка абсорбція стимулює інтенсивне виділення інсуліну підшлунковою залозою для транспорту глюкози у клітини, що може призвести до реактивної гіпоглікемії через дві-три години з відчуттям слабкості, голоду та зниженням концентрації уваги. Складні вуглеводи, або полісахариди, що включають крохмаль у злаках, бобових, картоплі та інших крохмалистих овочах, а також глікоген у м'ясі, розщеплюються повільніше ферментами амілазою слини та підшлункової залози протягом декількох годин та забезпечують більш стабільне енергопостачання з поступовим підвищенням глюкози крові.

Харчові волокна, хоча формально є вуглеводами, представленими целюлозою, геміцелюлозою, пектинами, гуарою та іншими полісахаридами, не

перетравлюються ферментами людського організму через відсутність необхідних целюлаз та інших ферментів, здатних розщеплювати бета-глікозидні зв'язки. Однак вони відіграють важливу роль у підтриманні здоров'я травної системи через кілька механізмів. Нерозчинні волокна збільшують об'єм калових мас, стимулюють перистальтику кишечника та попереджують запори, скорочуючи час транзиту їжі через товсту кишку. Розчинні волокна формують гель у шлунково-кишковому тракті, уповільнюють всмоктування глюкози та холестерину, що сприяє стабілізації рівня цукру в крові та зниженню рівня холестерину ліпопротеїнів низької щільності.

Оптимальне співвідношення макронутрієнтів залежить від конкретних цілей харчування користувача, його рівня фізичної активності, метаболічних особливостей та стану здоров'я. На рисунку 1.2 представлено рекомендовані розподіли макронутрієнтів для трьох основних цілей харчування: підтримка ваги, схуднення та набір м'язової маси.

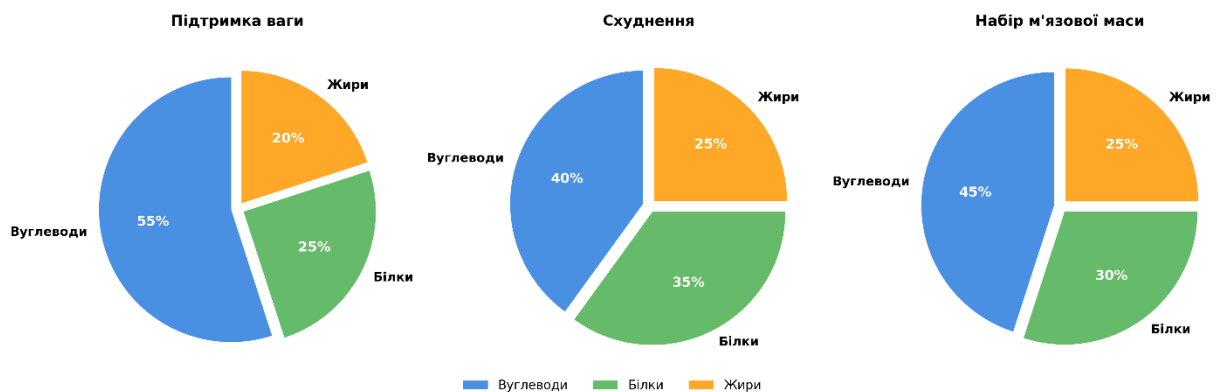


Рисунок 1.2 - Рекомендований розподіл макронутрієнтів для різних цілей

Як видно з рисунка 1.2, для підтримки ваги рекомендоване співвідношення макронутрієнтів забезпечує збалансований підхід з акцентом на вуглеводи як основне джерело енергії. Частка вуглеводів від п'ятдесяти до шістдесяти відсотків забезпечує достатню енергію для мозку, який споживає близько ста двадцяти грамів глюкози на день, та для фізичної активності середньої інтенсивності без накопичення надлишкових запасів глікогену.

1.2 Аналіз існуючих рішень та технологій

Ринок мобільних додатків для управління харчуванням та здоров'ям демонструє стрімке зростання протягом останнього десятиліття, що обумовлено кількома ключовими факторами. За даними аналітичних агентств, глобальний ринок додатків для здоров'я та фітнесу оцінюється у понад чотири мільярди доларів США станом на 2023 рік з прогнозованим щорічним зростанням близько двадцяти відсотків до 2028 року. Цей ріст обумовлений підвищенням обізнаності населення про важливість здорового харчування та профілактики хронічних захворювань, поширенням смартфонів з потужними процесорами та якісними камерами у всіх сегментах населення, пандемією COVID-19, яка стимулювала інтерес до самостійного моніторингу здоров'я та дистанційних консультацій, а також зростанням кількості людей з надмірною вагою та супутніми метаболічними порушеннями.

Аналіз існуючих мобільних додатків для планування харчування виявив широкий спектр рішень з різним функціоналом, підходами до монетизації та науковою обґрунтованістю методів. Деякі додатки зосереджені виключно на трекінгу споживаної їжі без надання рекомендацій щодо планування, інші пропонують готові меню на основі популярних дієт, треті використовують складні алгоритми оптимізації для персоналізованої генерації планів. Більшість успішних рішень використовують freemium модель монетизації, де базовий функціонал доступний безкоштовно, а розширені можливості, такі як детальна аналітика, персоналізовані рекомендації, відсутність реклами та інтеграції з іншими сервісами, доступні за платною підпискою.

Додаток MyFitnessPal є одним з найпопулярніших та найбільш встановлюваних інструментів для трекінгу харчування у світі з понад двома сотнями мільйонів зареєстрованих користувачів станом на 2024 рік та щоденною активною аудиторією понад двадцять мільйонів користувачів. Платформа має базу даних понад чотирнадцять мільйонів продуктів з понад п'ятдесяти країн, яка

постійно поповнюється як професійними адміністраторами, так і самими користувачами через краудсорсинговий підхід з системою верифікації та рейтингування записів. Щодня до бази додається понад п'ять тисяч нових продуктів, що забезпечує актуальність інформації та охоплення нових товарів харчової промисловості.

Додаток інтегрується з більш ніж п'ятдесятьма різними фітнес-трекерами та додатками для автоматичного врахування фізичної активності та енергетичних витрат під час тренувань, включаючи популярні платформи Fitbit з понад тридцятьма мільйонами користувачів, Garmin зі спеціалізацією на спортивних GPS-годинниках, Apple Health як центральний хаб для всіх даних про здоров'я в екосистемі iOS, Google Fit для користувачів Android, Strava для бігунів та велосипедистів, Polar для моніторингу пульсу та багато інших. MyFitnessPal дозволяє встановлювати цілі щодо калорійності та макронутрієнтів на основі введених користувачем параметрів, використовуючи алгоритм розрахунку, що базується на формулі Міффіліна-Сан Жеора з урахуванням цільової швидкості зміни ваги від нуля цілих п'яти десятих до одного кілограма на тиждень.

Користувачі можуть сканувати штрих-коди продуктів за допомогою камери смартфона для швидкого додавання до щоденника харчування, що працює для понад мільйона упакованих продуктів у базі даних та значно прискорює процес реєстрації споживання з п'яти-десяти хвилин ручного введення до десяти-п'ятнадцяти секунд сканування. Додаток надає можливість зберігати улюблені страви та рецепти для швидкого доступу без повторного пошуку, створювати власні рецепти з автоматичним розрахунком харчової цінності на основі інгредієнтів та їхніх пропорцій з урахуванням втрат маси при термічній обробці, копіювати цілі прийоми їжі з попередніх днів одним натисканням для повторюваних паттернів харчування та аналізувати динаміку споживання поживних речовин через інтерактивні графіки та детальні звіти за будь-який період часу.

MyFitnessPal також має розвинені соціальні функції, що дозволяють користувачам додавати друзів з можливістю перегляду їхнього прогресу, ділитися своїми досягненнями у вигляді віх втрати ваги або дотримання цілей, надавати взаємну підтримку через коментарі та вподобання, брати участь у групових челенджах з досягнення цілей за певний період та обмінюватися рецептами здорових страв. Преміум-версія додатку за ціною близько десяти доларів на місяць пропонує додаткові функції, такі як налаштування цільових значень макронутрієнтів окремо за прийомами їжі для оптимізації розподілу білка після тренування, більш детальну аналітику з відстеженням понад двадцяти мікронутрієнтів включаючи вітаміни та мінерали, відсутність реклами для покращення користувацького досвіду, можливість експорту даних у форматах CSV та PDF для аналізу у зовнішніх програмах та пріоритетну підтримку.

Однак ключовим обмеженням MyFitnessPal є те, що система не генерує персоналізовані плани харчування автоматично, а лише відстежує фактичне споживання та порівнює його з цільовими показниками, покладаючи на користувача повну відповідальність за планування меню та підбір продуктів [6]. Це може бути складним завданням для людей без базових знань у нутриціології, які не знають, як скласти збалансований раціон самотійно, що саме їсти для досягнення цілей, які продукти поєднувати між собою та як розподілити калорії та макронутрієнти протягом дня. Багато користувачів просто вводять усе, що з'їли, без попереднього планування, що часто призводить до незбалансованого харчування з надлишком або дефіцитом окремих нутрієнтів.

Платформа Lifesum позиціонується як комплексний інструмент для здорового харчування з більш holistic підходом до управління способом життя, що включає не тільки харчування, але й фізичну активність, якість сну та рівень гідратації. Додаток має понад п'ятдесят мільйонів завантажень у магазинах додатків та активну базу користувачів у понад десяти мільйонів людей щомісяця. На відміну від MyFitnessPal, Lifesum пропонує функції генерації планів харчування на основі обраної користувачем дієти з понад п'ятнадцяти різних

опцій, кожна з яких базується на науковій літературі та рекомендаціях провідних організацій охорони здоров'я.

Доступні дієти включають середземноморську дієту, що базується на традиційному харчуванні країн Середземномор'я з акцентом на оливкову олію як основне джерело жирів, жирну рибу багату омега-3 два-три рази на тиждень, велику кількість свіжих овочів та фруктів п'ять-сім порцій на день, цільнозернові продукти та помірне споживання червоного м'яса; скандинавську дієту з традиційними продуктами північних країн, що включає місцеві ягоди багаті антиоксидантами, корінні овочі такі як буряк та морква, жирну рибу з холодних морів, цільнозернові продукти з жита та вівса; кетогенну дієту з дуже низьким вмістом вуглеводів до двадцяти-п'ятдесяти грамів на день та високим вмістом жирів до сімдесяти-вісімдесяти відсотків від калорійності для досягнення стану кетозу з використанням кетонових тіл як альтернативного джерела енергії.

Також доступні інтервальне голодування з різними протоколами, такими як 16:8 з вікном харчування вісім годин та голодуванням шістнадцять годин, 18:6 з більш тривалим періодом голодування, 5:2 з п'ятьма днями нормального харчування та двома днями з обмеженням калорій до п'ятисот-шістсот кілокалорій або метод eat-stop-eat з повним голодуванням двадцять чотири години один-два рази на тиждень; палеодієта, що намагається імітувати харчування давніх предків людини та виключає зернові продукти, бобові, молочні продукти, цукор та оброблені продукти, зосереджуючись на м'ясі, рибі, овочах, фруктах, горіхах та насінні; веганська дієта з повним виключенням продуктів тваринного походження та вегетаріанська дієта з виключенням м'яса та риби, але збереженням молочних продуктів та яєць.

Додаток надає користувачам понад дві тисячі рецептів з покроковими інструкціями приготування на понад двадцяти мовах, високоякісними професійними фотографіями страв з різних ракурсів, детальною інформацією про час підготовки інгредієнтів, час активного приготування та загальний час від початку до подачі на стіл, рівень складності від простого без кулінарних навичок до експертного з технічними прийомами, а також калорійність та повний склад

макронутрієнтів і ключових мікронутрієнтів на порцію. Рецепти організовані за категоріями сніданків з опціями від швидких сніданків за п'ять хвилин до поживних чаш зі злаками та фруктами, обідів з легкими салатами та сендвічами для робочих днів, вечерь з повноцінними гарячими стравами та гарнірами, перекусів з здоровими опціями до ста п'ятдесяти кілокалорій та десертів з низьким вмістом цукру.

Lifesum автоматично формує списки покупок на основі обраних страв для тижневого меню, організовуючи продукти за категоріями такими як овочі та фрукти, м'ясо та риба, молочні продукти, зернові та бобові, олії та спеції для зручності покупок у супермаркеті з можливістю обходу магазину по відділах без повернень. Додаток забезпечує трекінг споживання води з персоналізованими рекомендаціями щодо денної норми на основі ваги, рівня активності та кліматичних умов, зазвичай тридцять-сорок мілілітрів на кілограм маси тіла, та нагадуваннями через push-повідомлення кожні дві-три години протягом дня. Візуалізація прогресу досягнення цілей реалізована через інтуїтивні графіки динаміки ваги з трендом за тиждень та місяць, кругові діаграми балансу макронутрієнтів за день та тиждень, та кольорові індикатори дотримання цільових показників із зеленим кольором при дотриманні цілей та червоним при значних відхиленнях.

Однак генерація планів харчування в Lifesum відбувається на основі попередньо створених шаблонів для кожного типу дієти професійними дієтологами та нутриціологами без глибокого врахування індивідуальних особливостей користувача [7]. Система не враховує специфічні вподобання користувача щодо конкретних продуктів за межами загальних категорій, детальні алергії на окремі інгредієнти крім основних вісьми алергенів, таких як глютен, лактоза, горіхи, соя, реальну доступність продуктів у конкретному регіоні або країні з урахуванням сезонності та асортименту місцевих магазинів, бюджетні обмеження користувача на тижневі або місячні витрати на харчування, часові можливості для приготування страв з урахуванням робочого графіку та

сімейних обов'язків, наявність кухонного обладнання такого як духовка, мультиварка, блендер або інші пристрої.

Це означає, що всі користувачі, які обрали однакову дієту, наприклад кетогенну, та мають схожі цільові показники калорійності у діапазоні дві тисячі кілокалорій з відхиленням плюс-мінус сто кілокалорій, отримують практично ідентичні плани харчування незалежно від їхніх особистих обставин, переваг, культурних традицій, релігійних обмежень та практичних можливостей. Наприклад, користувач з обмеженим бюджетом може отримати план з дорогими інгредієнтами, такими як лосось, авокадо або кіноа, які складно знайти або придбати у його регіоні. Користувач з алергією на певні види горіхів може побачити у плані страви, що містять різні види горіхів без можливості їхнього виключення окремо.

Мобільний застосунок Eat This Much займає унікальну нішу на ринку завдяки спеціалізації на автоматичній генерації персоналізованих планів харчування з використанням алгоритмів оптимізації та має понад один мільйон завантажень з високим рейтингом чотири цілих п'ять десятих зірки в обох магазинах додатків. Система працює на основі детального профілю користувача, який включає не тільки стандартні параметри віку від вісімнадцяти до вісімдесяти років, статі, ваги від сорока до двохсот кілограмів та зросту від ста сорока до двохсот двадцяти сантиметрів, але й розширену інформацію про харчові вподобання з можливістю вибору до ста конкретних продуктів з категорій улюблених, небажаних інгредієнтів з виключенням до п'ятдесяти позицій.

Користувачі також вказують алергени з вісьмох основних категорій та можливістю додавання специфічних алергій, бюджетні обмеження з встановленням мінімальної та максимальної вартості на день, тиждень або місяць у місцевій валюті, максимально допустимий час на приготування однієї страви від п'яти хвилин для швидких рецептів до двох годин для складних страв, кількість прийомів їжі на день від трьох стандартних до шести для дрібного харчування, включення або виключення перекусів між основними прийомами

їжі, кулінарні навички користувача від початківця без досвіду приготування їжі до досвідченого кулінара з широким репертуаром технік, доступність кухонного обладнання з чек-листом основних пристроїв, таких як плита, духовка, мікрохвильова піч, мультиварка, блендер, кухонний комбайн.

Користувачі можуть вказати рівень складності бажаних рецептів з п'ятибальною шкалою від дуже простих страв, що не вимагають приготування або потребують мінімальної теплової обробки, таких як салати з готових інгредієнтів або сендвічі, через прості страви з базовими техніками смаження або варіння, середні рецепти з комбінацією кількох методів приготування, складні страви з багатьма етапами підготовки та різними компонентами до дуже складних багатокомпонентних рецептів з професійними технічними прийомами, такими як су-від, карамелізація або фламбірування. Також є можливість вказати діючі дієтичні обмеження з понад двадцяти опцій, включаючи безглютенове, безлактозне, низьковуглеводне, низькожирове, високобілкове, вегетаріанське, веганське, палео, кето, DASH для контролю гіпертензії харчування.

Eat This Much використовує комбінацію алгоритмів оптимізації для підбору комбінацій страв, що відповідають цільовим показникам калорійності з точністю плюс-мінус п'ятдесят кілокалорій та макронутрієнтів з точністю плюс-мінус п'ять грамів при мінімізації вартості продуктів у межах заданого бюджету або часу приготування залежно від пріоритетів користувача, які можна налаштувати повзунками. Алгоритм працює ітеративно у декілька етапів: спочатку формує початковий план на основі евристичних правил та популярності страв серед користувачів з подібними параметрами, потім покращує його через локальний пошук з заміною окремих страв або продуктів для досягнення кращого балансу всіх критеріїв одночасно, використовуючи функцію оцінки, що враховує відхилення від цільових значень калорій та макронутрієнтів, вартість плану, час приготування та різноманітність продуктів.

Додаток дозволяє регенерувати окремі прийоми їжі або навіть окремі страви, якщо запропонована опція не підходить користувачу через будь-які причини, при цьому система автоматично коригує інші прийоми їжі протягом

дня для збереження загального балансу плану з досягненням цільової калорійності та співвідношення макронутрієнтів. Користувачі можуть заморожувати певні страви або продукти, які їм особливо подобаються або які вони вже придбали та мають у холодильнику, щоб вони залишалися незмінними при регенерації інших частин плану, що забезпечує гнучкість планування та зменшує харчові відходи. Також є функція масштабування рецептів з автоматичним перерахунком кількості інгредієнтів для приготування більшої або меншої кількості порцій, що зручно для сімейного харчування або приготування страв наперед на кілька днів.

Преміум-версія Eat This Much за ціною близько дев'яти доларів на місяць надає автоматичну генерацію списків покупок з можливістю експорту у різні формати, організацію продуктів за відділами супермаркету відповідно до типового розташування товарів у великих мережах, можливість планування на тиждень, два тижні або місяць наперед з урахуванням повторення страв для зручності приготування, інтеграцію з онлайн-сервісами доставки продуктів у США та деяких європейських країнах з можливістю замовлення всіх інгредієнтів одним натисканням, підтримку планування для декількох членів родини одночасно з різними цілями, обмеженнями та вподобаннями для кожної персони, а також адаптацію планів на основі сезонності продуктів з пріоритетом місцевих овочів та фруктів у сезон для економії коштів та максимальної свіжості.

Основним недоліком Eat This Much є обмежена база рецептів, особливо для неамериканських кухонь, оскільки більшість з близько п'яти тисяч рецептів у базі орієнтовані на продукти та страви, популярні у Сполучених Штатах, такі як буріто, тако, бургери, панкейки, блюда з індички, кранч-рол та інші специфічні позиції. Користувачі з інших регіонів, особливо з Азії, Африки або Східної Європи, можуть відчувати нестачу традиційних страв своєї кухні та незнайомість з пропонованими інгредієнтами або способами приготування. Також відсутня адаптація планів на основі реального дотримання користувачем раціону та накопичення історичних даних, тобто система не навчається на історії вподобань користувача протягом тижнів та місяців використання [8].

Система не аналізує, які страви користувач часто пропускає або замінює іншими, які продукти він ніколи не використовує навіть якщо вони регулярно потрапляють у план, які прийоми їжі він найчастіше дотримується або порушує, який час приготування є реально прийнятним у будні та вихідні дні. Це означає, що навіть після місяців використання додаток може продовжувати пропонувати страви, які користувач систематично відхиляє через смакові переваги, складність приготування, недоступність інгредієнтів або інші причини, не враховуючи ці дані для покращення майбутніх рекомендацій та підвищення персоналізації з часом.

Порівняльний аналіз функціональності та можливостей найпопулярніших додатків для планування харчування систематизовано у таблиці 1.1, яка дозволяє наочно оцінити сильні та слабкі сторони кожного рішення за ключовими параметрами.

Аналіз даних таблиці 1.1 показує, що існуючі рішення демонструють різні підходи до балансу між функціональністю та складністю використання з різними компромісами. MyFitnessPal забезпечує найпростіший досвід для трекінгу харчування з мінімальним порогом входу та найбільшою базою продуктів, але не надає інструментів для планування меню, що робить його непридатним для користувачів без знань у нутриціології. Lifesum пропонує готові плани харчування на основі популярних науково обґрунтованих дієт з привабливим інтерфейсом, проте має обмеження у гнучкості налаштувань під індивідуальні потреби через шаблонний підхід. Eat This Much демонструє найвищий рівень персоналізації через використання алгоритмів оптимізації з урахуванням множини параметрів, але стикається з проблемою обмеженості бази рецептів, особливо для неамериканських кухонь, та відсутності механізмів адаптивного навчання на основі історичних даних користувача.

Таблиця 1.1 - Порівняння існуючих рішень для планування харчування

Додаток	Генерація планів	Персоналізація	Трекінг	База продуктів	Основні недоліки
MyFitnessPal	Ні	Низька	Так	14+ млн продуктів	Відсутність автоматичної генерації планів
Lifesum	Так	Середня	Так	2000+ рецептів	Шаблонні плани без глибокої персоналізації
Eat This Much	Так	Висока	Так	5000+ рецептів	Обмежена база рецептів, відсутність адаптивного навчання

Узагальнюючи результати аналізу існуючих рішень, можна виділити декілька категорій спільних обмежень, що характерні для більшості додатків незалежно від їхньої популярності та функціональності. По-перше, спостерігається недостатня персоналізація генерованих планів харчування, оскільки більшість систем використовують загальні шаблони або обмежені набори правил без глибокого врахування індивідуальних особливостей метаболізму користувача, його способу життя з графіком роботи та відпочинку, графіку роботи з можливістю приготування страв, сімейного стану з наявністю дітей або інших утриманців, рівня кулінарних навичок від початківця до експерта, харчових звичок, сформованих роками або десятиліттями. Навіть системи, що декларують персоналізацію, часто обмежуються базовими

параметрами та не враховують складні взаємодії між різними факторами, такі як зв'язок між стресом на роботі та тягою до солодкого, вплив якості сну на апетит наступного дня або сезонні зміни харчових вподобань.

1.3 Дослідження методів формування планів харчування

Формування персоналізованих планів харчування є складною багатокритеріальною задачею комбінаторної оптимізації, що вимагає одночасного врахування множини обмежень та оптимізації декількох часто конфліктуючих цільових функцій. Ця задача може розглядатися з різних точок зору залежно від обраного методологічного підходу, кожен з яких має свої переваги у певних аспектах, недоліки у інших та оптимальні сценарії застосування відповідно до специфіки вимог. Вибір методу суттєво впливає на якість генерованих планів з точки зору дотримання цільових показників, швидкість їхнього формування від секунд до хвилин, можливості персоналізації з урахуванням індивідуальних особливостей, різноманітність пропонованих рішень та складність реалізації системи з точки зору програмування.

Методи формування персоналізованих планів харчування можна класифікувати на декілька основних категорій з різними принципами роботи та областями застосування: шаблонні методи, що базуються на попередньо створених планах харчування професійними дієтологами; евристичні методи, що використовують набір правил для послідовного конструювання розв'язку; методи математичної оптимізації, що формулюють задачу як пошук глобального оптимуму функції з обмеженнями; генетичні алгоритми та еволюційні методи, що імітують природний відбір для знаходження якісних розв'язків; методи машинного навчання, що навчаються на історичних даних для прогнозування вподобань. Кожна категорія представляє унікальний підхід до вирішення задачі

планування з різним рівнем складності реалізації, обчислювальних вимог, автоматизації та здатності до адаптації під конкретного користувача.

Шаблонні методи базуються на використанні попередньо створених планів харчування професійними дієтологами та нутриціологами для різних категорій користувачів з типовими цілями та обмеженнями. Такі плани розробляються з урахуванням наукових рекомендацій щодо збалансованого харчування, співвідношення макронутрієнтів для конкретних цілей, розподілу калорій між прийомами їжі протягом дня та різноманітності продуктів для забезпечення всіх необхідних мікронутрієнтів. Наприклад, може існувати шаблон для схуднення жінки віком тридцять-сорок років з калорійністю тисяча п'ятсот кілокалорій, шаблон для набору м'язової маси чоловіка з калорійністю три тисячі кілокалорій або шаблон вегетаріанської дієти з калорійністю дві тисячі кілокалорій. Кожен шаблон містить детальний розклад прийомів їжі на тиждень або місяць з конкретними стравами, продуктами, розмірами порцій та інструкціями з приготування.

Основною перевагою шаблонного підходу є дуже висока швидкість генерації плану, оскільки система просто підбирає найбільш відповідний шаблон з бібліотеки на основі параметрів користувача та цільової калорійності з можливою незначною корекцією порцій для точного співпадіння з цілями. Час генерації становить менше однієї секунди, що дозволяє миттєво надавати рекомендації користувачам. Також перевагою є гарантована збалансованість планів, оскільки вони створені кваліфікованими спеціалістами з урахуванням усіх нутрієнтів, а не тільки калорій та макронутрієнтів. Професійні дієтологи забезпечують різноманітність продуктів, правильне поєднання інгредієнтів для кращого засвоєння та врахування кулінарних аспектів приготування страв.

Однак шаблонний підхід має суттєві недоліки з точки зору персоналізації та гнучкості. Низька персоналізація проявляється у тому, що всі користувачі з подібними параметрами отримують ідентичні або майже ідентичні плани без урахування їхніх індивідуальних вподобань щодо конкретних продуктів, небажаних інгредієнтів крім основних алергенів, культурних або релігійних

обмежень, бюджетних можливостей та доступності продуктів у регіоні. Також шаблони не можуть враховувати специфічні медичні обмеження користувача, такі як підвищений рівень холестерину, що вимагає обмеження насичених жирів, предіабет з необхідністю контролю глікемічного індексу страв або гіпертензія з обмеженням натрію до двох грамів на день.

Складність підтримки шаблонів також є значним недоліком, оскільки для охоплення всього різноманіття користувачів потрібна велика бібліотека шаблонів з десятками або сотнями варіантів для різних комбінацій статі, віку, цілей, рівня активності, типу дієти та обмежень. Створення кожного шаблону вимагає значних зусиль кваліфікованих дієтологів з розрахунками харчової цінності всіх страв, балансуванням нутрієнтів, підбором продуктів та тестуванням реалістичності плану. Крім того, шаблони потребують регулярного оновлення для відображення нових наукових рекомендацій щодо харчування, які можуть змінюватися кожні декілька років, появи нових продуктів на ринку та зміни доступності сезонних продуктів.

Евристичні методи використовують набір правил та стратегій, розроблених на основі експертних знань у галузі дієтології та досвіду практичного планування харчування, для послідовного конструювання плану шляхом вибору продуктів та страв на кожному кроці. Типова евристична стратегія може працювати наступним чином: спочатку визначається кількість прийомів їжі на день, зазвичай від трьох до п'яти, та розподіляється цільова калорійність між ними відповідно до рекомендацій, наприклад, сніданок двадцять п'ять відсотків, обід тридцять п'ять відсотків, вечеря тридцять відсотків та перекуси десять відсотків. Потім для кожного прийому їжі послідовно обираються страви або продукти з відповідних категорій з урахуванням цільового балансу макронутрієнтів.

Наприклад, для сніданку алгоритм може спочатку обрати джерело складних вуглеводів, таке як вівсянка або цільнозерновий хліб, потім додати джерело білка, наприклад яйця або грецький йогурт, далі джерело здорових жирів, такі як горіхи або авокадо, і нарешті свіжі фрукти або овочі для вітамінів

та клітковини. На кожному кроці продукт обирається з урахуванням поточного балансу макронутрієнтів, що залишилися для досягнення цілей, вподобань користувача з пріоритетом улюблених продуктів, обмежень на алергени та небажані інгредієнти, різноманітності з уникненням повторення продуктів протягом дня або кількох днів, бюджету з перевагою економічних опцій та сезонності з пріоритетом місцевих продуктів у сезон.

Перевагами евристичних методів є прийнятна швидкість роботи, оскільки алгоритм виконує обмежену кількість кроків без складних обчислень, зазвичай формуючи план за декілька секунд, що прийнятно для інтерактивних додатків. Евристики забезпечують достатню гнучкість для врахування різноманітних обмежень користувача через систему правил, що можуть легко розширюватися для нових типів обмежень без переробки всього алгоритму. Також метод забезпечує зрозумілість та прозорість процесу генерації, оскільки кожне рішення про вибір продукту базується на явних правилах, що може бути важливим для довіри користувачів та можливості пояснення рекомендацій.

Недоліком евристичних методів є відсутність гарантії оптимальності розв'язку, оскільки жадібний вибір на кожному кроці може призвести до субоптимального загального результату. Наприклад, обравши на початку дня продукти з високим вмістом жирів, алгоритм може зіткнутися з проблемою балансування макронутрієнтів наприкінці дня, коли залишається мало калорій, але потрібно додати білок без перевищення жирів. Також евристики можуть застрягати у локальних оптимумах, коли подальше покращення плану неможливе без перегляду раніше зроблених виборів. Залежність від якості правил означає, що ефективність методу сильно залежить від експертизи розробників правил та їхнього розуміння предметної області, а погано сформульовані правила можуть призводити до незбалансованих або нереалістичних планів.

Методи математичної оптимізації формують задачу планування харчування як задачу знаходження глобального оптимуму цільової функції при дотриманні множини обмежень, що може бути представлено у вигляді лінійного

або нелінійного програмування. Задача може бути сформульована наступним чином: необхідно визначити кількість кожного продукту або страви з доступного набору, що мінімізує відхилення від цільових значень калорійності та макронутрієнтів при дотриманні обмежень на виключення алергенів, мінімальну та максимальну кількість кожного продукту для забезпечення різноманітності, обмеження бюджету, обмеження на окремі нутрієнти такі як натрій або холестерин та інші специфічні вимоги.

Цільова функція може мати вигляд мінімізації зваженої суми квадратів відхилень фактичних значень від цільових: мінімізувати суму вагових коефіцієнтів помножених на квадрати відхилень калорій, білків, жирів та вуглеводів від їхніх цільових значень плюс штраф за порушення інших обмежень. Вагові коефіцієнти дозволяють регулювати відносну важливість різних критеріїв, наприклад, надаючи більшу вагу точності калорійності та меншу вагу точному співвідношенню окремих макронутрієнтів. Для розв'язання таких задач можна використовувати методи лінійного програмування, якщо всі функції та обмеження лінійні, або методи нелінійного програмування для більш складних випадків.

Основною перевагою методів оптимізації є можливість знаходження глобально оптимального розв'язку відносно заданої цільової функції, що гарантує найкращий можливий баланс між усіма критеріями з урахуванням їхніх вагових коефіцієнтів. Математична строгість підходу дозволяє формально довести властивості розв'язку та оцінити його якість. Методи оптимізації добре масштабуються до великих баз продуктів з тисячами позицій, оскільки ефективні алгоритми лінійного програмування можуть обробляти задачі з десятками тисяч змінних та обмежень. Також підхід забезпечує гнучкість у формулюванні різноманітних обмежень та цільових функцій без зміни базового алгоритму розв'язання.

Недоліком методів математичної оптимізації є висока обчислювальна складність, особливо для задач великої розмірності або нелінійних задач, що може призводити до часу розв'язування від десятків секунд до хвилин, що

неприйнятно для інтерактивних мобільних додатків. Складність формулювання задачі вимагає глибоких знань у галузі оптимізації для коректного визначення цільової функції, обмежень та вагових коефіцієнтів, а помилки у формулюванні можуть призводити до нереалістичних або незбалансованих розв'язків. Можливість отримання нереалістичних розв'язків виникає, коли математична модель не повністю відображає всі аспекти реальної задачі, наприклад, оптимальний розв'язок може містити дуже малі дробові кількості багатьох різних продуктів замість реалістичних порцій обмеженої кількості страв.

Генетичні алгоритми та еволюційні методи імітують процес природного відбору для знаходження якісних розв'язків задачі планування харчування без необхідності формулювання задачі як математичної оптимізації з явною цільовою функцією. Метод працює з популяцією можливих розв'язків, де кожен розв'язок представляє конкретний план харчування з набором страв та їхніх кількостей. Алгоритм починає з випадкової початкової популяції з декількох десятків або сотень різних планів харчування. На кожній ітерації виконується оцінка якості кожного плану за допомогою функції пристосованості, що враховує відхилення від цільових показників, дотримання обмежень, різноманітність продуктів та інші критерії.

Далі відбувається селекція, де кращі плани з вищою пристосованістю мають більшу ймовірність бути обраними для створення нащадків наступного покоління. Схрещування поєднує частини двох батьківських планів для створення нових планів, наприклад, беручи сніданок та обід від одного плану та вечерю від іншого. Мутація вносить випадкові зміни у плани з невеликою ймовірністю, наприклад, замінюючи один продукт іншим з тієї ж категорії або змінюючи розмір порції, що дозволяє досліджувати нові області простору розв'язків та уникати передчасної збіжності до локальних оптимумів. Процес повторюється протягом багатьох поколінь, зазвичай від ста до тисячі ітерацій, поки не буде досягнуто задовільної якості розв'язку або вичерпано ліміт часу.

Перевагою генетичних алгоритмів є здатність до багатокритеріальної оптимізації з одночасним врахуванням множини конфліктуючих цілей, таких як

мінімізація відхилень макронутрієнтів, максимізація різноманітності, мінімізація вартості та часу приготування, без необхідності зведення їх до єдиної скалярної функції з ваговими коефіцієнтами. Робастність до формулювання задачі означає, що метод може працювати з будь-якою функцією оцінки без вимог до її диференційовності або лінійності. Генетичні алгоритми добре підходять для паралелізації, оскільки оцінка пристосованості різних особин може виконуватися незалежно на різних процесорах або ядрах, що дозволяє суттєво прискорити обчислення на багатоядерних системах.

Недоліками генетичних алгоритмів є необхідність налаштування багатьох параметрів, таких як розмір популяції, ймовірність схрещування та мутації, стратегія селекції, критерій зупинки, що вимагає експериментального підбору для конкретної задачі та може суттєво впливати на якість результатів. Довгий час виконання є проблемою, оскільки потрібно оцінити пристосованість сотень або тисяч планів харчування протягом багатьох поколінь, що може займати від десятків секунд до хвилин залежно від розміру популяції та кількості ітерацій. Відсутність гарантії оптимальності означає, що метод може не знайти глобальний оптимум, особливо якщо припинити виконання передчасно або з невдалими параметрами алгоритму.

Порівняння методів формування планів харчування за ключовими характеристиками представлено у таблиці 1.2, що дозволяє оцінити переваги та недоліки кожного підходу.

Таблиця 1.2 - Порівняння методів планування харчування

Метод	Точність	Швидкість	Гнучкість	Складність	Персоналізація
Шаблонний	Висока	Дуже висока	Низька	Низька	Низька
Евристичний	Середня	Висока	Середня	Середня	Середня
Математична оптимізація	Дуже висока	Низька	Висока	Висока	Висока
Генетичні алгоритми	Висока	Середня	Дуже висока	Середня	Висока

Аналіз таблиці 1.2 показує, що кожен метод має свої сильні та слабкі сторони, що робить вибір підходу залежним від конкретних вимог системи. Для мобільного додатку критичними є швидкість генерації та прийнятна персоналізація, що робить комбінований підхід з евристичними та локальною оптимізацією найбільш перспективним.

1.4 Постановка задачі дослідження

На основі проведеного аналізу існуючих рішень, методів збору даних та підходів до формування планів харчування можна сформулювати задачу дослідження та розробки мобільної системи для автоматизованої генерації та моніторингу персональних планів харчування. Основною метою роботи є створення системи, що поєднає науково обґрунтовані принципи дієтології з сучасними методами алгоритмічної оптимізації та зручним мобільним інтерфейсом для забезпечення доступності персоналізованого планування харчування широкому колу користувачів без необхідності консультацій з професійними дієтологами.

Задача формулюється як розробка мобільної системи з комбінованим підходом до генерації планів харчування, що використовує евристичні методи для швидкого формування початкового допустимого розв'язку та методи локального пошуку для оптимізації балансу макронутрієнтів з досягненням точності плюс-мінус п'ять відсотків від цільових значень. Система має враховувати індивідуальні параметри користувача, включаючи вік від вісімнадцяти до вісімдесяти років, стать, вагу від сорока до двохсот кілограмів, зріст від ста сорока до двохсот двадцяти сантиметрів, рівень фізичної активності з п'яти категорій, цільову зміну ваги від мінус одного до плюс нуля цілих п'яти десятих кілограма на тиждень, харчові обмеження та алергії з можливістю

виключення до ста конкретних продуктів, вподобання щодо типів страв та кухонь світу, бюджетні обмеження та часові можливості для приготування їжі.

Для досягнення поставленої мети необхідно вирішити низку завдань, що охоплюють різні аспекти розробки системи від математичного моделювання до практичної реалізації. Першим завданням є розробка математичної моделі задачі формування плану харчування як задачі багатокритеріальної оптимізації з обмеженнями, що включає формулювання цільової функції для мінімізації відхилень від цільових значень калорійності та макронутрієнтів, визначення обмежень на виключення алергенів, мінімальну та максимальну кількість кожного продукту для забезпечення різноманітності, обмеження бюджету та часу приготування, а також врахування інших специфічних вимог користувача, таких як вегетаріанство або релігійні обмеження.

Другим завданням є проєктування клієнт-серверної архітектури системи з чітким розділенням відповідальності між мобільним застосунком та серверною частиною. Мобільний застосунок має забезпечувати інтуїтивний користувацький інтерфейс для введення персональних даних, перегляду згенерованих планів харчування, реєстрації фактичного споживання їжі через ручне введення, сканування штрих-кодів або фотографування страв, відображення аналітики та прогресу досягнення цілей через графіки та діаграми. Серверна частина має виконувати обчислювально складні операції генерації планів з використанням алгоритмів оптимізації, зберігати користувацькі дані та історію харчування у захищеній базі даних з шифруванням чутливої інформації, інтегруватися з зовнішніми базами даних харчових продуктів для отримання актуальної інформації про харчову цінність, надавати RESTful API для взаємодії з мобільним застосунком.

Третім завданням є розробка алгоритмічного забезпечення для генерації персональних планів харчування, що включає реалізацію комбінованого підходу з евристичним методом для швидкого формування початкового розв'язку за декілька секунд та методом локального пошуку для покращення балансу макронутрієнтів через заміну окремих продуктів або коригування порцій.

Алгоритм має забезпечувати генерацію планів за час не більше десяти секунд на сучасних мобільних пристроях середнього класу, досягати точності балансу макронутрієнтів з середнім відхиленням не більше п'яти відсотків від цільових значень, враховувати всі обмеження користувача без порушень жорстких обмежень на алергени, забезпечувати різноманітність продуктів з мінімальним повторенням протягом тижня.

Четвертим завданням є реалізація функціональних модулів системи, що включає модуль реєстрації користувачів з валідацією введених даних, шифруванням паролів та безпечною автентифікацією через JSON Web Tokens; модуль управління профілями з можливістю оновлення параметрів, цілей та вподобань; модуль генерації планів харчування з викликом відповідних алгоритмів та збереженням результатів; модуль моніторингу фактичного споживання з підтримкою ручного введення, сканування штрих-кодів через інтеграцію з OpenFoodFacts API та фотографування страв з розпізнаванням через машинне навчання; модуль аналітики з розрахунком статистики дотримання плану, визначенням динаміки досягнення цілей, виявленням паттернів харчування та візуалізацією результатів через інтерактивні графіки.

П'ятим завданням є інтеграція з зовнішніми джерелами даних про харчову цінність продуктів, що включає імпорт даних з бази USDA FoodData Central з понад трьомастами п'ятдесятьма тисячами продуктів, інтеграцію з OpenFoodFacts API для отримання інформації про упаковані продукти через штрих-коди з можливістю кешування даних для офлайн-доступу, створення власної бази українських продуктів та традиційних страв на основі таблиць Скурихіна з можливістю розширення користувачами через модерацію, забезпечення регулярного оновлення даних для відображення змін у рецептурах продуктів виробниками.

Шостим завданням є проведення експериментального дослідження для оцінки ефективності розробленої системи, що включає залучення тестової групи користувачів з двадцяти п'яти осіб з різними цілями, включаючи схуднення, підтримку ваги та набір м'язової маси, збір даних про використання системи

протягом чотирьох тижнів з щоденною реєстрацією споживання їжі та щотижневими вимірюваннями ваги, оцінку точності генерованих планів харчування через порівняння фактичних та цільових значень калорійності і макронутрієнтів, оцінку зручності використання системи через опитування System Usability Scale з десятима питаннями, аналіз досягнення користувачами їхніх цілей щодо зміни ваги за період дослідження з статистичною перевіркою значущості результатів.

Ключовими викликами у реалізації системи є забезпечення одночасного врахування множини обмежень при генерації планів, оскільки користувач може мати десятки виключених продуктів, кілька алергій, обмеження бюджету та часу, що значно звужує простір допустимих розв'язків та ускладнює знаходження збалансованого плану. Досягнення швидкості генерації планів менше десяти секунд при збереженні якості балансування макронутрієнтів вимагає ретельної оптимізації алгоритмів, ефективних структур даних для зберігання та пошуку продуктів, кешування проміжних результатів та можливої попередньої обробки даних. Інтеграція з українським контекстом через створення бази традиційних українських страв та адаптацію до доступності продуктів на місцевому ринку є важливою для релевантності системи для цільової аудиторії.

Забезпечення довгострокової залученості користувачів через геймфікацію, соціальні функції та адаптацію рекомендацій на основі історичних даних є критичним для практичної цінності системи, оскільки багато користувачів припиняють використання додатків для здоров'я після першого місяця через втрату мотивації або складність постійного введення даних. Система має надавати механізми підтримки мотивації через відображення прогресу, досягнень, серій днів дотримання плану, можливість ділитися результатами та отримувати підтримку від спільноти користувачів.

Очікуваними результатами роботи є функціональна мобільна система для операційних систем Android та iOS з інтуїтивним інтерфейсом, що відповідає сучасним стандартам дизайну Material Design та Human Interface Guidelines відповідно; математична модель задачі планування харчування з чітко

сформульованими цільовими функціями та обмеженнями; ефективний комбінований алгоритм генерації планів з точністю балансу макронутрієнтів плюс-мінус п'ять відсотків та швидкістю менше десяти секунд; база даних українських продуктів з понад тисячею позицій; експериментальне підтвердження ефективності системи з оцінкою зручності використання за шкалою SUS не менше сімдесяти п'яти балів та статистично значущими результатами досягнення користувачами цільових змін ваги.

Практична цінність розробленої системи полягає у забезпеченні доступності персоналізованого планування харчування широкому колу користувачів без необхідності звертання до професійних дієтологів, вартість консультацій яких може становити від п'ятисот до двох тисяч гривень за прийом. Система дозволяє зекономити час на самостійному плануванні меню, що може займати від однієї до двох годин на тиждень, підвищити усвідомленість харчових звичок через детальний трекінг споживання та аналітику, досягти поставлених цілей щодо зміни ваги та покращення здоров'я через дотримання збалансованого раціону з оптимальним співвідношенням макронутрієнтів та достатньою кількістю мікронутрієнтів. Система може бути адаптована для використання у фітнес-центрах для клієнтів, що потребують контролю харчування під час тренувань, клініках дієтології як допоміжний інструмент для пацієнтів між консультаціями, санаторно-курортних закладах для гостей з медичними показаннями та корпоративних програмах здоров'я для співробітників компаній.

2 ПРОЄКТУВАННЯ СИСТЕМИ ГЕНЕРАЦІЇ ПЛАНІВ ХАРЧУВАННЯ

2.1 Побудова концептуальної моделі системи

Концептуальна модель системи визначає основні компоненти, їхні функції та взаємодію на високому рівні абстракції без деталей технічної реалізації. Розроблювана система базується на клієнт-серверній архітектурі з

чітким розділенням відповідальності між мобільним застосунком, що виконується на пристрої користувача, та серверною частиною, що забезпечує бізнес-логіку, обробку даних та інтеграцію з зовнішніми сервісами. Такий підхід дозволяє розподілити обчислювальне навантаження між клієнтом та сервером, забезпечити централізоване управління даними користувачів з можливістю синхронізації між пристроями, спростити оновлення бізнес-логіки без необхідності перевипуску мобільного застосунку через магазини додатків, масштабувати серверні ресурси відповідно до кількості користувачів.

Мобільний застосунок відповідає за представлення даних користувачу через інтуїтивний графічний інтерфейс, що відповідає платформним гайдлайнам Material Design для Android та Human Interface Guidelines для iOS. Застосунок забезпечує збір введених користувачем даних з валідацією на стороні клієнта для негайного зворотного зв'язку про помилки, відображення згенерованих планів харчування у зручному для перегляду форматі з можливістю перегляду деталей кожної страви, реєстрацію фактичного споживання їжі через різні методи введення, включаючи ручне введення з автодоповненням, сканування штрих-кодів через камеру пристрою з розпізнаванням у режимі реального часу, фотографування страв з подальшою ідентифікацією через машинне навчання, візуалізацію аналітики та прогресу через інтерактивні графіки, діаграми та індикатори з можливістю деталізації за різні періоди часу.

Мобільний застосунок також забезпечує локальне кешування даних для роботи в офлайн-режимі з можливістю перегляду раніше згенерованих планів, введення записів про споживання їжі без підключення до інтернету та автоматичної синхронізації з сервером при відновленні з'єднання. Застосунок підтримує push-повідомлення для нагадувань про прийоми їжі у заплановані години з можливістю налаштування користувачем, нагадувань про введення записів, якщо користувач забув зареєструвати споживання протягом дня, повідомлень про досягнення цілей або важливих віх у прогресі, мотиваційних повідомлень для підтримання довгострокової залученості. Локальне сховище використовується для збереження налаштувань користувача, токенів

автентифікації для безпечного доступу до API, кешованих даних про продукти для швидкого пошуку.

Серверна частина виконує обчислювально складні операції, що вимагають значних ресурсів процесора та пам'яті, зокрема генерацію персональних планів харчування з використанням комбінованих алгоритмів оптимізації, що можуть виконуватися декілька секунд для великих баз продуктів, обробку зображень страв для розпізнавання інгредієнтів через згорткові нейронні мережі з тисячами параметрів, агрегацію та аналіз історичних даних користувача для виявлення паттернів харчування, трендів споживання окремих макронутрієнтів, кореляцій між дотриманням плану та досягненням цілей. Сервер забезпечує зберігання користувацьких даних у реляційній базі даних з транзакційними гарантіями ACID для забезпечення цілісності та узгодженості інформації, шифрування чутливої інформації, такої як персональні параметри користувачів, медичні дані, історія ваги, регулярне резервне копіювання для запобігання втрати даних.

Серверна частина також відповідає за інтеграцію з зовнішніми сервісами та базами даних, включаючи USDA FoodData Central для отримання детальної інформації про харчову цінність продуктів з понад тридцятьма нутрієнтами, OpenFoodFacts API для розпізнавання упакованих продуктів за штрих-кодами з базою понад двох мільйонів товарів з усього світу, сервіси машинного навчання для розпізнавання зображень їжі з використанням попередньо навчених моделей або власних моделей, платіжні системи для обробки підписок на преміум-функціональність з підтримкою різних методів оплати, аналітичні сервіси для збору статистики використання застосунку та виявлення проблем або можливостей покращення. Сервер надає RESTful API для взаємодії з мобільним застосунком через стандартизовані HTTP методи GET для отримання даних, POST для створення нових записів, PUT або PATCH для оновлення існуючих, DELETE для видалення з JSON форматом даних для легкої серіалізації та десеріалізації.

Взаємодія основних компонентів системи на концептуальному рівні проілюстрована на рисунку 2.1, що показує розподіл функціональності між клієнтом та сервером, а також зовнішні інтеграції.

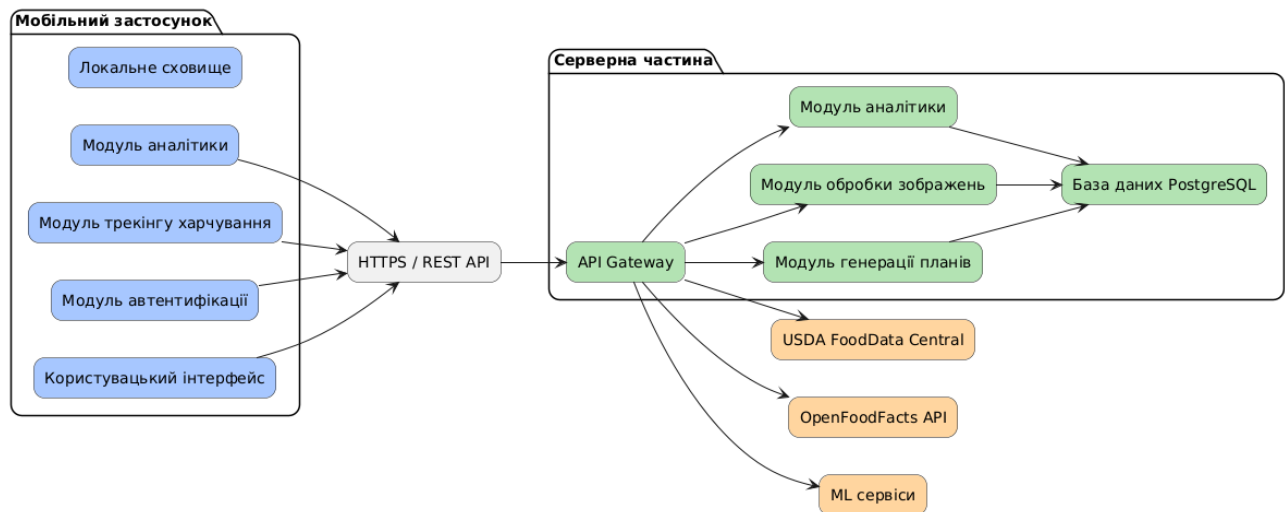


Рисунок 2.1 - Концептуальна модель системи

Як видно з рисунка 2.1, система має чітке розділення відповідальності між компонентами, що забезпечує модульність архітектури та можливість незалежного розвитку окремих частин. Мобільний застосунок зосереджений на презентаційній логіці та взаємодії з користувачем, мінімізуючи складні обчислення для економії батареї пристрою. Серверна частина виконує всю важку роботу з генерації планів, обробки даних та інтеграції з зовнішніми сервісами, що дозволяє легко масштабувати систему при зростанні навантаження шляхом додавання додаткових серверних інстанцій.

Основні варіанти використання системи визначають функціональні вимоги з точки зору користувача та охоплюють всі ключові сценарії взаємодії з системою. Реєстрація нового користувача передбачає введення базової інформації, такої як електронна пошта для ідентифікації та зв'язку, пароль з вимогами мінімум вісім символів, включаючи цифри та спеціальні символи для безпеки, ім'я для персоналізації інтерфейсу, підтвердження електронної пошти через посилання у листі для верифікації, згода з умовами використання та політикою конфіденційності відповідно до вимог GDPR. Система перевіряє

унікальність електронної пошти у базі даних, хешує пароль з використанням bcrypt з достатньою кількістю раундів для захисту від перебору, створює запис користувача у базі даних з унікальним ідентифікатором UUID, надсилає вітальний лист з інструкціями щодо налаштування профілю.

Налаштування профілю користувача є критично важливим для персоналізації планів харчування та включає введення фізичних параметрів: вік у роках для врахування вікових особливостей метаболізму, стать для застосування відповідної формули розрахунку базального метаболізму, поточна вага у кілограмах з точністю до одного десяткового знаку, зріст у сантиметрах для розрахунку індексу маси тіла та енергетичних потреб. Користувач також вказує цільову вагу, що визначає напрямок зміни та необхідний дефіцит або профіцит калорій, швидкість зміни ваги від нуля цілих п'яти десятих до одного кілограма на тиждень з попередженнями про ризики занадто швидкого схуднення або набору, рівень фізичної активності з детальними описами кожної категорії від малорухливого до дуже активного способу життя для точного розрахунку коефіцієнта активності.

Також важливою частиною профілю є харчові вподобання та обмеження, що включають вибір типу дієти з опцій всеїдна без обмежень, вегетаріанська з виключенням м'яса та риби, веганська з виключенням всіх продуктів тваринного походження, кетогенна з мінімумом вуглеводів, палео без зернових та молочних продуктів, середземноморська з акцентом на оливкову олію та рибу. Користувач може вказати алергії на конкретні продукти або групи продуктів з вибором з понад ста позицій, включаючи глютен у пшениці та інших зернових, лактозу у молочних продуктах, горіхи різних видів від арахісу до бразильського горіха, морепродукти та рибу, сою та соєві продукти, яйця курячі та перепелині. Небажані продукти дозволяють виключити продукти, які користувач просто не любить або не хоче споживати з культурних чи релігійних причин без медичних показань.

Генерація персонального плану харчування є центральною функціональністю системи та ініціюється користувачем через відповідну кнопку

в інтерфейсі після налаштування профілю або у будь-який момент для оновлення плану. Мобільний застосунок збирає всі релевантні параметри з профілю користувача, формує запит до API сервера з JSON об'єктом, що містить всі необхідні дані, надсилає запит через HTTPS з'єднання з токеном автентифікації у заголовку для ідентифікації користувача. Сервер отримує запит, перевіряє валідність токена автентифікації та права доступу користувача, витягує параметри з тіла запиту та виконує валідацію коректності значень, розраховує цільову калорійність на основі базального метаболізму, коефіцієнта активності та цільової швидкості зміни ваги.

Далі сервер визначає оптимальний розподіл макронутрієнтів відповідно до цілі користувача з використанням рекомендацій для схуднення, підтримки ваги або набору м'язової маси, викликає алгоритм генерації плану харчування з передачею всіх параметрів та обмежень, алгоритм виконується протягом декількох секунд з використанням комбінованого евристичного та оптимізаційного підходу, результат зберігається у базі даних з прив'язкою до користувача та поточної дати генерації, сервер формує відповідь у JSON форматі з детальним планом на тиждень, включаючи перелік страв для кожного прийому їжі з назвами, інгредієнтами, кількостями, інструкціями приготування, харчову цінність кожної страви та загальні денні підсумки калорій та макронутрієнтів, надсилає відповідь мобільному застосунку.

Мобільний застосунок отримує відповідь від сервера, перевіряє статус відповіді на успішність виконання запиту, парсить JSON об'єкт з планом харчування у внутрішні структури даних застосунку, зберігає план у локальному сховищі для можливості офлайн-доступу, відображає план користувачу у зручному інтерфейсі з можливістю перегляду по днях тижня та прийомах їжі, надає можливість переглянути деталі кожної страви, включаючи список інгредієнтів з точними кількостями, покрокові інструкції приготування з часом для кожного етапу, детальну харчову цінність з розбивкою по нутрієнтам, фотографію страви, якщо доступна у базі рецептів. Користувач може регенерувати окремі прийоми їжі або страви, якщо запропонований варіант не

підходить, система повторює процес генерації тільки для вибраного прийому їжі з коригуванням інших прийомів для збереження балансу дня.

Реєстрація фактичного споживання їжі дозволяє користувачу відстежувати дотримання плану та аналізувати реальне харчування з можливістю коригування поведінки. Користувач обирає один з методів введення залежно від ситуації та доступності інформації про продукт. Ручне введення передбачає пошук продукту або страви у базі даних через текстове поле з автодоповненням, що показує найбільш релевантні результати після введення трьох символів на основі частоти використання цього продукту користувачем та загальної популярності. Користувач вибирає знайдений продукт зі списку результатів, вказує кількість у грамах, мілілітрах або порціях з можливістю швидкого вибору стандартних розмірів порцій для зручності, обирає прийом їжі, до якого відноситься споживання, з опцій сніданок, другий сніданок, обід, полуденок, вечеря, пізня вечеря, додає запис до щоденника харчування з автоматичним розрахунком внеску у денні підсумки калорій та макронутрієнтів.

2.2 Проектування архітектури системи

Архітектура мобільного застосунку базується на патерні Model-View-ViewModel, який забезпечує чітке розділення відповідальності між компонентами та полегшує тестування і підтримку коду. Цей патерн особливо підходить для мобільної розробки з React Native, де компоненти мають чітку ієрархію та потребують ефективного управління станом. Модель представляє дані та бізнес-логіку застосунку, включаючи структури даних для користувача з полями ідентифікатор UUID, електронна пошта, хешований пароль, ім'я, дата реєстрації, дата останнього входу, профілю користувача з фізичними параметрами вік, стать, вага, зріст, цільовою вагою, швидкістю зміни, рівнем активності, вподобаннями та обмеженнями, плану харчування з денними меню,

прийомами їжі, стравами, інгредієнтами, харчовою цінністю, записів щоденника з датою, часом, продуктом, кількістю, прийомом їжі.

ViewModel діє як посередник між View та Model, забезпечуючи трансформацію даних з формату моделі у формат, зручний для відображення, обробку логіки презентації, такої як форматування чисел, дат, текстів, управління станом компонентів з використанням React hooks useState для локального стану, useEffect для побічних ефектів, useContext для глобального стану, виконання запитів до API з обробкою завантаження, успіху та помилок. View представляє користувацький інтерфейс через React компоненти, що відображають дані з ViewModel без знання про модель напряду, реагують на дії користувача через callback функції, передані з ViewModel, використовують стилізацію через React Native StyleSheet або styled-components для інкапсуляції стилів.

Діаграма класів основних сутностей системи представлена на рисунку 2.2, що показує структуру даних та зв'язки між ними.

Як видно з рисунка 2.2, структура даних організована ієрархічно з центральною сутністю User, від якої залежать інші компоненти системи. Зв'язок один до одного між User та UserProfile забезпечує єдиний профіль для кожного користувача з можливістю оновлення параметрів. Зв'язок один до багатьох між User та MealPlan дозволяє зберігати історію згенерованих планів з можливістю повернення до попередніх версій. Аналогічно зв'язок один до багатьох між User та FoodLog забезпечує накопичення повної історії харчування для аналізу трендів.

Серверна архітектура побудована на основі шарової моделі з чітким розділенням відповідальності між рівнями, що забезпечує модульність, тестовність та можливість горизонтального масштабування. Шар API Gateway є точкою входу для всіх запитів від мобільних застосунків та інших клієнтів, виконуючи функції маршрутизації запитів до відповідних сервісів на основі URL шляху та HTTP методу, автентифікації та авторизації через перевірку JWT токенів у заголовках запитів з валідацією підпису, терміну дії та прав доступу,

rate limiting для захисту від зловживань з обмеженням кількості запитів від одного користувача або IP адреси до ста запитів на хвилину з можливістю налаштування для різних endpoints, логування всіх запитів з деталями користувача, часу, параметрів, статусу відповіді для моніторингу та аудиту, обробки CORS для забезпечення безпечної взаємодії з веб-клієнтами.

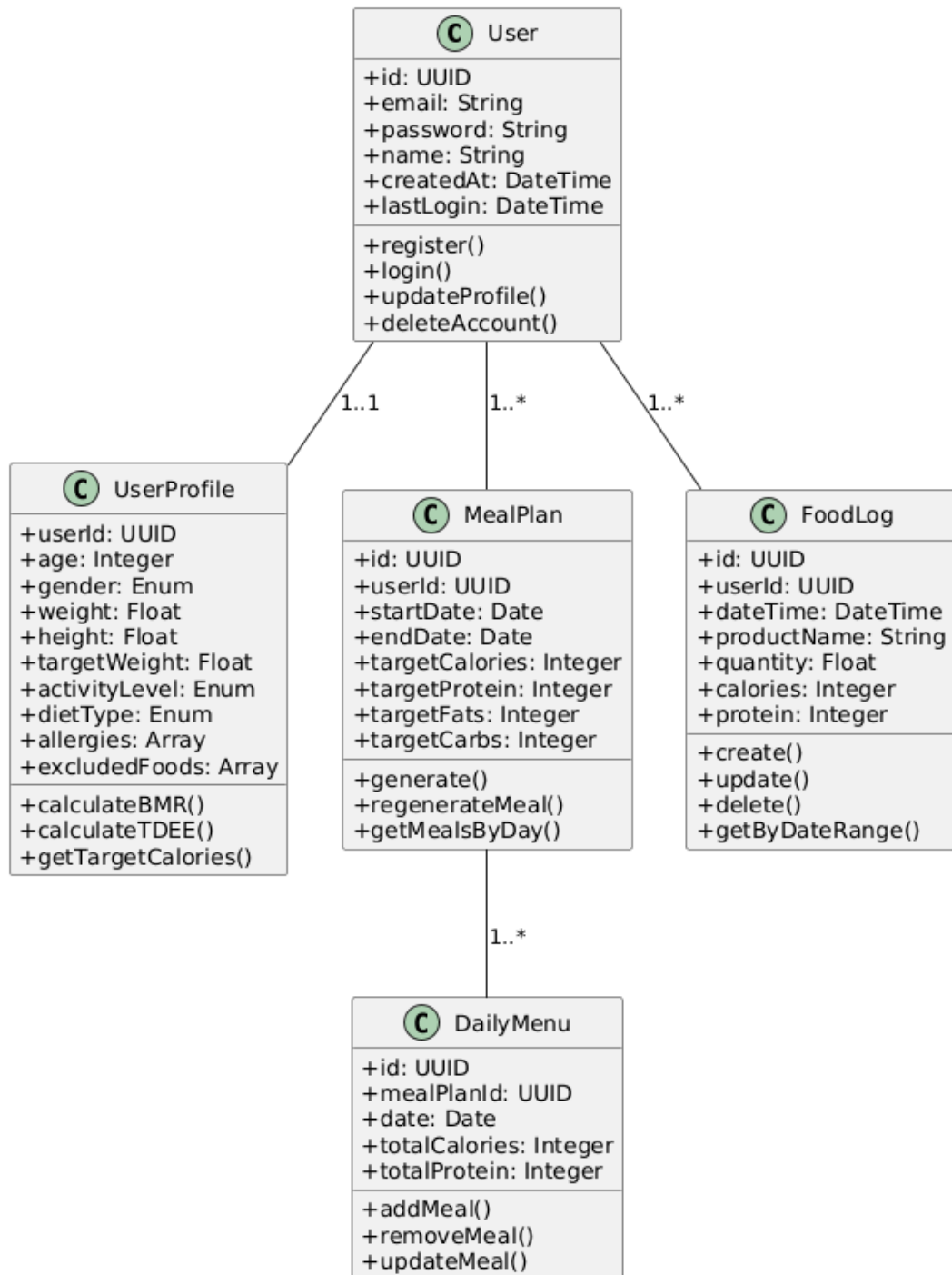


Рисунок 2.2 - Діаграма класів основних сутностей

Бізнес-логіка шар містить основну функціональність системи, організовану у сервіси з чітко визначеною відповідальністю. UserService відповідає за управління користувачами, включаючи реєстрацію нових користувачів з валідацією унікальності email, хешуванням пароля через bcrypt з десятима раундами, створенням запису у базі даних, автентифікацію через перевірку email та пароля з генерацією JWT токена з терміном дії двадцять чотири години, оновлення профілю з валідацією змін та збереженням історії, видалення акаунта з анонімізацією даних замість повного видалення для збереження статистики. MealPlanService керує генерацією та управлінням планами харчування, викликаючи алгоритми генерації з передачею параметрів користувача, зберігаючи згенеровані плани у базі даних з прив'язкою до користувача, надаючи можливість регенерації окремих прийомів їжі або днів, забезпечуючи експорт планів у різні формати PDF для друку, JSON для інтеграції.

FoodLogService обробляє записи щоденника харчування, створюючи нові записи з валідацією коректності даних, оновлюючи існуючі записи з перерахунком денних підсумків, видаляючи записи з можливістю відновлення протягом тридцяти днів, агрегуючи дані для аналітики з розрахунком середніх значень, трендів, відхилень від плану. AnalyticsService виконує складні аналітичні запити, розраховує статистику дотримання плану з відсотком днів дотримання, середнім відхиленням калорій та макронутрієнтів, визначає динаміку досягнення цілей з розрахунком швидкості зміни ваги, прогнозованої дати досягнення цільової ваги, виявляє паттерни харчування через аналіз часових рядів споживання, кореляцій між продуктами, генерує звіти та візуалізації з діаграмами прогресу, графіками споживання нутрієнтів.

Схема бази даних PostgreSQL організована з урахуванням нормалізації до третьої нормальної форми для мінімізації дублювання даних та забезпечення цілісності. Основні таблиці та їхні зв'язки представлені у таблиці 2.1.

Таблиця 2.1 - Схема бази даних системи

Таблиця	Основні поля	Зв'язки
users	id, email, password_hash, name, created_at, last_login, deleted_at	1:1 з user_profiles, 1:N з meal_plans, 1:N з food_logs
user_profiles	user_id, age, gender, weight, height, target_weight, activity_level, diet_type, updated_at	N:1 з users через user_id
meal_plans	id, user_id, start_date, end_date, target_calories, target_protein, target_fats, target_carbs, created_at	N:1 з users, 1:N з daily_menus
daily_menus	id, meal_plan_id, date, total_calories, total_protein, total_fats, total_carbs	N:1 з meal_plans, 1:N з meals
meals	id, daily_menu_id, meal_type, recipe_id, serving_size, calories, protein, fats, carbs	N:1 з daily_menus, N:1 з recipes
food_logs	id, user_id, date_time, product_name, quantity, unit, calories, protein, fats, carbs, meal_type	N:1 з users через user_id
recipes	id, name, description, cuisine_type, preparation_time, cooking_time, difficulty, calories, protein, fats, carbs	1:N з recipe_ingredients, 1:N з meals

База даних використовує індекси для оптимізації запитів на часто використовуваних полях: первинний ключ id у всіх таблицях з типом UUID для глобальної унікальності, зовнішні ключі для забезпечення референційної цілісності з каскадним оновленням та обмеженням видалення, індекс на email у

таблиці users для швидкого пошуку при автентифікації, індекс на user_id у всіх таблицях з користувацькими даними для ефективного фільтрування, композитний індекс на user_id та date у таблиці food_logs для запитів за періодом, індекс на created_at для сортування за датою створення. Додатково використовуються обмеження цілісності: UNIQUE на email для запобігання дублюванню облікових записів, NOT NULL на обов'язкових полях, CHECK для валідації діапазонів значень, наприклад вік від вісімнадцяти до ста двадцяти, вага від тридцяти до трьохсот кілограмів.

2.3 Розробка алгоритмічного забезпечення

Алгоритм генерації персональних планів харчування є центральним компонентом системи та використовує комбінований підхід, що поєднує переваги евристичних методів для швидкого формування початкового розв'язку з методами локальної оптимізації для покращення балансу макронутрієнтів. Такий гібридний підхід дозволяє досягти прийнятної якості планів за обмежений час виконання, що критично важливо для мобільного застосунку. Алгоритм складається з декількох послідовних етапів, кожен з яких вирішує специфічну підзадачу загальної проблеми планування з урахуванням всіх обмежень та цілей користувача.

Перший етап включає розрахунок цільових показників на основі параметрів користувача з профілю. Спочатку обчислюється базальний метаболізм за формулою Міффіліна-Сан Жеора з різними константами для чоловіків та жінок, що враховує вагу у кілограмах, зріст у сантиметрах та вік у роках. Для чоловіків BMR дорівнює десять помножене на вагу плюс шість цілих двадцять п'ять сотих помножене на зріст мінус п'ять помножене на вік плюс п'ять. Для жінок формула аналогічна, але константа становить мінус сто шістдесят один замість плюс п'ять. Далі розраховується загальна денна потреба

в енергії шляхом множення базального метаболізму на коефіцієнт фізичної активності, який обирається з п'яти категорій: 1.2 для малорухливого, 1.375 для легкої активності, 1.55 для помірної, 1.725 для високої, 1.9 для дуже високої активності.

Цільова калорійність визначається на основі мети користувача щодо зміни ваги з коригуванням загальної денної потреби. Для схуднення цільова калорійність зменшується на від трьохсот до п'ятисот кілокалорій на день для повільного схуднення нуля цілих п'яти десятих кілограма на тиждень або на від п'ятисот до тисячі кілокалорій для помірного схуднення одного кілограма на тиждень з попередженнями про ризики більш агресивного дефіциту. Для підтримки ваги цільова калорійність дорівнює загальній денній потребі без коригувань. Для набору м'язової маси цільова калорійність збільшується на від трьохсот до п'ятисот кілокалорій для повільного набору близько нуля цілих п'яти десятих кілограма на тиждень переважно м'язової маси за умови силових тренувань.

Розподіл макронутрієнтів визначається відповідно до мети користувача з використанням науково обґрунтованих рекомендацій. Для схуднення рекомендується від тридцяти до тридцяти п'яти відсотків білка для збереження м'язової маси під час дефіциту калорій, від двадцяти п'яти до тридцяти відсотків жирів для підтримання гормонального балансу, решта від тридцяти п'яти до сорока п'яти відсотків вуглеводів. Для підтримки ваги використовується збалансований розподіл від двадцяти п'яти до тридцяти відсотків білка, від двадцяти до двадцяти п'яти відсотків жирів, від п'ятдесяти до п'ятдесяти п'яти відсотків вуглеводів. Для набору м'язової маси збільшується частка вуглеводів для енергії тренувань від сорока п'яти до п'ятдесяти відсотків, підтримується висока кількість білка від двадцяти п'яти до тридцяти п'яти відсотків, жири становлять від двадцяти до двадцяти п'яти відсотків.

Другий етап алгоритму виконує фільтрацію бази продуктів та рецептів з урахуванням обмежень користувача для формування множини допустимих варіантів. Виключаються всі продукти, що містять алергени користувача з

перевіркою по списку інгредієнтів кожного рецепту, продукти зі списку небажаних користувачем незалежно від причин, продукти, що не відповідають обраному типу дієти, наприклад м'ясо та риба для вегетаріанців, всі продукти тваринного походження для веганів, зернові та бобові для палеодієти, продукти з високим вмістом вуглеводів для кетогенної дієти. Також фільтруються рецепти за максимальним часом приготування, якщо користувач вказав обмеження, за рівнем складності відповідно до кулінарних навичок користувача, за доступністю інгредієнтів у регіоні з пріоритетом місцевих продуктів.

Третій етап використовує жадібну евристичну стратегію для швидкого формування початкового плану харчування на тиждень з розподілом цільової калорійності між прийомами їжі. Для кожного дня тижня спочатку визначається кількість прийомів їжі зазвичай чотири або п'ять, включаючи сніданок, обід, вечерю та один-два перекуси, розподіляється цільова денна калорійність між прийомами у пропорціях сніданок двадцять п'ять відсотків, обід тридцять п'ять відсотків, вечеря тридцять відсотків, перекуси по п'ять відсотків кожен. Для кожного прийому їжі послідовно обираються рецепти або продукти з допустимої множини за певною стратегією відбору, що враховує поточний баланс макронутрієнтів для дня, різноманітність з уникненням повторень рецептів протягом тижня, вподобання користувача з пріоритетом продуктів з високим рейтингом, сезонність продуктів з перевагою місцевих у сезон, бюджетні обмеження з вибором економічних варіантів при їхній наявності.

2.4 Вибір технологічного стеку

Вибір технологічного стеку для розробки мобільної системи генерації та моніторингу персональних планів харчування базується на аналізі вимог проекту, особливостей цільових платформ, доступності ресурсів розробки та довгострокових перспектив підтримки системи. Критерії вибору технологій

включають продуктивність виконання критичних операцій, таких як генерація планів та обробка зображень, кросплатформеність для одночасної розробки під Android та iOS з мінімальним дублюванням коду, зрілість технології з наявністю великої спільноти розробників, документації та готових бібліотек, екосистему інструментів розробки, тестування та розгортання, простоту навчання для команди розробників з можливістю швидкого старту проекту.

Для розробки мобільного застосунку обрано фреймворк React Native версії нуля цілих сімдесят чотири десятих, що забезпечує кросплатформену розробку з єдиною кодовою базою JavaScript для Android та iOS платформ з можливістю досягнення близько дев'яноста відсотків спільного коду. React Native використовує декларативний підхід до створення інтерфейсів з компонентною архітектурою, що полегшує повторне використання коду, тестування окремих частин та командну розробку. Фреймворк надає доступ до нативних API платформ через JavaScript bridge з можливістю написання платформно-специфічного коду для критичних за продуктивністю компонентів. Велика екосистема бібліотек включає понад п'ятнадцять тисяч готових пакетів у npm реєстрі для вирішення типових задач мобільної розробки.

Управління станом застосунку реалізовано через бібліотеку Redux версії п'ять цілих нуль десятих, що забезпечує централізоване сховище стану з передбачуваними змінами через actions та reducers. Redux дозволяє відстежувати всі зміни стану для налагодження, реалізувати time-travel debugging для відтворення історії взаємодій користувача, легко тестувати бізнес-логіку незалежно від компонентів інтерфейсу. Для асинхронних операцій, таких як запити до API, використовується Redux Thunk middleware, що дозволяє dispatch функцій замість об'єктів actions для виконання складних асинхронних потоків з послідовними або паралельними запитами. Також використовується Redux Persist для збереження частини стану у локальному сховищі AsyncStorage для офлайн-роботи та швидшого завантаження при повторному відкритті застосунку.

Навігація між екранами застосунку реалізована через React Navigation версії шість цілих нуль десятих, що надає нативний вигляд та поведінку навігації для кожної платформи з підтримкою різних типів навігаторів: `stack navigator` для ієрархічної навігації з можливістю повернення назад, `tab navigator` для головного меню з вкладками внизу екрану, `drawer navigator` для бокового меню з основними розділами застосунку. Бібліотека забезпечує глибоке посилання для відкриття конкретних екранів з `push`-повідомлень або зовнішніх джерел, анімації переходів між екранами з налаштуванням швидкості та типу, передачу параметрів між екранами для контекстної навігації.

Для виконання HTTP запитів до серверного API використовується бібліотека `Axios` версії один цілих шість десятих, що надає зручний інтерфейс для роботи з промісами, автоматичну трансформацію JSON даних у запитах та відповідях, можливість налаштування глобальних перехоплювачів для додавання токенів автентифікації, обробки помилок, логування запитів. `Axios` підтримує скасування запитів через `AbortController` для уникнення обробки відповідей від застарілих запитів, автоматичні повтори запитів при тимчасових помилках мережі, таймаути для запобігання зависанню застосунку при проблемах зі з'єднанням. Базовий URL API та інші конфігурації зберігаються у змінних оточення через `react-native-config` для легкої зміни між середовищами розробки, тестування та продакшену.

Серверна частина розроблена на Node.js версії вісімнадцять цілих нуль десятих LTS з фреймворком Express версії чотири цілих вісімнадцять десятих для створення RESTful API. Node.js обраний завдяки високій продуктивності для I/O операцій через асинхронну неблокуючу архітектуру на базі V8 JavaScript engine, можливості використання однієї мови програмування JavaScript для клієнта та сервера, що спрощує обмін кодом та підвищує продуктивність команди, великій екосистемі пакетів npm з понад двома мільйонами бібліотек для вирішення різноманітних задач. Express надає мінімалістичний та гнучкий фреймворк для створення веб-додатків з підтримкою `middleware` для модульної

обробки запитів, маршрутизації для організації endpoints, шаблонізації для генерації HTML при необхідності.

Для взаємодії з базою даних використовується ORM Sequelize версії шість цілих тридцять п'ять десятих, що забезпечує об'єктно-орієнтований інтерфейс для роботи з реляційними базами з підтримкою PostgreSQL, MySQL, MariaDB, SQLite, Microsoft SQL Server. Sequelize надає автоматичне генерування SQL запитів з JavaScript об'єктів та методів, міграції бази даних для версіонування схеми та безпечного розгортання змін, валідації даних на рівні моделей перед збереженням у базу, транзакції для забезпечення атомарності складних операцій, асоціації між моделями для опису зв'язків один до одного, один до багатьох, багато до багатьох, hooks для виконання додаткової логіки до або після збереження, оновлення, видалення записів.

Як система управління базами даних обрано PostgreSQL версії п'ятнадцять цілих нуль десятих завдяки високій надійності з ACID транзакціями, що гарантують цілісність даних навіть при збоях системи, розширеним можливостям SQL з підтримкою складних запитів, підзапитів, оконних функцій, Common Table Expressions, JSON типам даних для зберігання структурованих документів з можливістю індексування та запитів, повнотекстовим пошуком з підтримкою різних мов включаючи українську через відповідні словники, механізмами реплікації для забезпечення високої доступності та розподілу навантаження читання. PostgreSQL має відкритий вихідний код з активною спільнотою розробників, регулярними оновленнями з новими функціями та виправленнями безпеки, хорошою документацією та великою кількістю інструментів адміністрування.

Порівняння обраних технологій з альтернативами представлено у таблиці 2.2, що показує переваги вибраного стеку для конкретних вимог проєкту.

Додаткові інструменти розробки включають систему контролю версій Git з хостингом на GitHub для співпраці команди, відстеження змін коду, code review через pull requests, continuous integration через GitHub Actions для автоматичного запуску тестів при кожному commit, ESLint для статичного аналізу коду

JavaScript з виявленням потенційних помилок, дотримання стилю коду, Prettier для автоматичного форматування коду відповідно до налаштованих правил, Jest для юніт-тестування з покриттям коду мінімум вісімдесят відсотків, Detox для end-to-end тестування мобільного застосунку на симуляторах та реальних пристроях.

Таблиця 2.2 - Порівняння технологічних рішень

Компонент	Обрана технологія	Альтернативи	Обґрунтування вибору
Мобільний фреймворк	React Native 0.74	Flutter, Ionic, Xamarin	Єдина кодова база для iOS та Android, велика екосистема бібліотек, можливість використання JavaScript
Управління станом	Redux 5.0	MobX, Context API, Zustand	Передбачувані зміни стану, відмінні інструменти налагодження, широка підтримка спільноти
Серверний фреймворк	Node.js + Express	Python + Django, Java + Spring	Висока продуктивність для I/O операцій, єдина мова з клієнтом, швидка розробка
База даних	PostgreSQL 15	MySQL, MongoDB, SQLite	ACID транзакції, розширені можливості SQL, підтримка JSON, повнотекстовий пошук
ORM	Sequelize 6.35	TypeORM, Prisma, Knex	Зріла бібліотека з великою спільнотою, підтримка міграцій, асоціацій, транзакцій
Автентифікація	JWT + bcrypt	OAuth2, Session-based	Stateless автентифікація, підходить для мобільних додатків, простота реалізації

Для моніторингу продакшен середовища використовується Sentry для відстеження помилок у реальному часі з деталями стеку викликів, контекстом користувача, даними пристрою, PM2 для управління процесами Node.js з автоматичним перезапуском при збоях, кластеризацією для використання всіх ядер процесора, логуванням, Prometheus для збору метрик продуктивності сервера, таких як використання CPU, пам'яті, кількість запитів, час відповіді, Grafana для візуалізації метрик через дашборди з графіками, діаграмами, сповіщеннями про аномалії. Для кешування часто запитуваних даних використовується Redis версії сім цілих нуль десятих для зберігання у пам'яті з швидким доступом мілісекунди, підтримкою різних структур даних рядки, списки, множини, хеші, можливістю налаштування термінів зберігання.

Обраний технологічний стек забезпечує збалансоване поєднання продуктивності, надійності, зручності розробки та довгострокової підтримки системи. Використання сучасних та зрілих технологій з великими спільнотами гарантує доступність документації, готових рішень типових проблем, регулярних оновлень з новими функціями та виправленнями безпеки. Кросплатформеність React Native значно знижує вартість розробки та підтримки порівняно з нативною розробкою окремо для iOS та Android. Серверний стек на Node.js та PostgreSQL забезпечує масштабованість системи для обслуговування тисяч одночасних користувачів з можливістю горизонтального масштабування через додавання серверних інстанцій за навантажувачем.

3 РЕАЛІЗАЦІЯ МОБІЛЬНОЇ СИСТЕМИ МОНІТОРИНГУ

ХАРЧУВАННЯ

3.1 Апаратна складова системи

Реалізована мобільна система призначена для роботи на сучасних смартфонах з операційними системами Android версії вісім цілих нуль десятих і

вище та iOS версії дванадцять цілих нуль десятих і вище, що охоплює більше дев'яноста п'яти відсотків активних пристроїв на ринку станом на 2024 рік. Мінімальні апаратні вимоги до пристроїв визначені з урахуванням необхідності забезпечення прийнятної продуктивності для всіх функцій застосунку, включаючи генерацію планів, обробку зображень та візуалізацію аналітики. Для комфортної роботи системи рекомендуються пристрої з процесором архітектури ARM64 з частотою від полутора гігагерц, що забезпечує швидке виконання JavaScript коду через V8 або JavaScriptCore движки, оперативною пам'яттю від двох гігабайт для Android пристроїв та від одного гігабайта для iOS пристроїв з більш ефективним управлінням пам'яттю.

Постійна пам'ять пристрою має містити мінімум сто мегабайт вільного простору для встановлення застосунку з базовими даними, включаючи APK або IPA файл розміром близько тридцять п'ять мегабайт після стиснення, початкову базу продуктів близько п'ятдесят мегабайт у стисненому форматі, додаткові бібліотеки та ресурси близько п'ятнадцять мегабайт. Під час використання застосунків може збільшити споживання пам'яті до трьохсот мегабайт через кешування зображень страв розміром від ста до двохсот кілобайт кожне з підтримкою до ста зображень у кеші, збереження історії харчування за останні тридцять днів близько десять мегабайт при активному використанні, локальну базу улюблених продуктів та рецептів до п'ятдесяти мегабайт, тимчасові файли та логи до двадцяти мегабайт.

Камера пристрою використовується для сканування штрих-кодів упакованих продуктів та фотографування страв для автоматичного розпізнавання інгредієнтів, тому рекомендується роздільна здатність мінімум вісім мегапікселів з автофокусом для чіткого захоплення штрих-кодів з відстані від п'яти до п'ятнадцяти сантиметрів та деталей страв для точного розпізнавання. Для сканування штрих-кодів використовується бібліотека react-native-camera з інтегрованим розпізнаванням форматів EAN-13, UPC-A, Code-128 та інших поширених стандартів у режимі реального часу з затримкою менше ста мілісекунд від наведення до розпізнавання. Фотографування страв вимагає

достатнього освітлення мінімум п'ятсот люкс для коректного розпізнавання кольорів та текстур їжі, що критично для точної ідентифікації інгредієнтів моделлю машинного навчання.

Підключення до інтернету необхідне для більшості функцій системи, включаючи генерацію нових планів харчування з викликом серверних API, синхронізацію даних між пристроями користувача через хмарне сховище, завантаження зображень страв з сервера для відображення у планах, запити до зовнішніх баз продуктів для отримання харчової цінності невідомих позицій, оновлення бази рецептів з новими стравами та виправленнями. Рекомендована швидкість з'єднання становить мінімум п'ятсот кілобіт на секунду для комфортної роботи з часом завантаження планів менше п'яти секунд, синхронізацією даних менше десяти секунд, завантаженням зображень менше двох секунд на фото. Застосунок підтримує офлайн-режим з обмеженою функціональністю, дозволяючи переглядати раніше завантажені плани харчування з кешованими зображеннями, вводити записи про споживання їжі з автоматичною синхронізацією при відновленні з'єднання, переглядати локально збережену аналітику за попередні періоди.

Серверна інфраструктура розгорнута на хмарній платформі з використанням сервісів AWS для забезпечення високої доступності, масштабованості та надійності системи. Основні компоненти інфраструктури та їхні характеристики представлені у таблиці 3.1.

Обрана конфігурація забезпечує обслуговування до п'яти тисяч одночасних користувачів з середнім часом відповіді API менше двохсот мілісекунд для простих запитів та менше п'яти секунд для генерації планів харчування. Автоматичне масштабування налаштоване через AWS Auto Scaling Groups з додаванням нових інстанцій при перевищенні використання CPU сімдесяти відсотків або середнього часу відповіді п'ятсот мілісекунд протягом п'яти хвилин. Резервне копіювання бази даних виконується щоденно з збереженням семи останніх копій та можливістю відновлення до будь-якої точки у часі протягом останніх семи днів через AWS RDS automated backups.

Таблиця 3.1 - Компоненти серверної інфраструктури

Компонент	Сервіс AWS	Конфігурація	Призначення
API сервер	EC2 t3.medium	2 vCPU, 4GB RAM	Обробка HTTP запитів, бізнес-логіка
База даних	RDS PostgreSQL	db.t3.small, 20GB SSD	Зберігання користувацьких даних
Кеш	ElastiCache Redis	cache.t3.micro, 1GB	Кешування сесій, продуктів
Файлове сховище	S3 Standard	50GB сховище	Зображення страв, резервні копії
Балансувальник	Application Load Balancer	Регіон EU-CENTRAL-1	Розподіл навантаження, SSL термінація

3.2 Реалізація модуля реєстрації та профілів користувачів

Модуль реєстрації та управління профілями користувачів реалізований з дотриманням сучасних стандартів безпеки та зручності використання, забезпечуючи надійну автентифікацію та захист персональних даних. Процес реєстрації нового користувача розпочинається з екрану реєстрації, що містить поля для введення електронної пошти з валідацією формату через регулярний вираз, що перевіряє наявність символу @, доменного імені та правильної структури адреси, пароля з вимогами мінімум вісім символів, включаючи хоча б одну велику літеру, одну малу літеру, одну цифру та один спеціальний символ для забезпечення достатньої складності, підтвердження пароля з перевіркою

співпадіння з основним полем для запобігання помилкам введення, імені користувача довжиною від двох до п'ятдесяти символів для персоналізації інтерфейсу.

Валідація введених даних виконується на стороні клієнта в реальному часі з негайним відображенням помилок під відповідними полями червоним кольором та іконками попередження для швидкого зворотного зв'язку користувачу. Після натискання кнопки реєстрації мобільний застосунок надсилає POST запит до endpoint `/api/auth/register` з JSON об'єктом, що містить введені дані. Серверна частина виконує додаткову валідацію всіх полів на бекенді для захисту від обходу клієнтської валідації, перевіряє унікальність електронної пошти у базі даних через запит до таблиці `users` з фільтром по полю `email`, хешує пароль з використанням `bcrypt` з десятьма раундами солі для захисту від атак перебору навіть у випадку витоку бази даних, створює новий запис у таблиці `users` з згенерованим UUID ідентифікатором, збереженням хешованого пароля, поточної дати та часу як `created_at`.

Після успішного створення облікового запису система генерує JWT токен з терміном дії двадцять чотири години для `access token`, що містить ідентифікатор користувача, електронну пошту, роль користувача за замовчуванням `user`, `timestamp` створення токена, підписує токен секретним ключем довжиною тридцять два байти, що зберігається у змінних оточення сервера та ніколи не передається клієнту. Також генерується `refresh token` з терміном дії сім днів для оновлення `access token` без повторної автентифікації, зберігається у базі даних у таблиці `refresh_tokens` з прив'язкою до користувача для можливості відкликання. Обидва токени повертаються клієнту у відповіді разом з інформацією про користувача ідентифікатор, ім'я, електронна пошта для ініціалізації стану застосунку.

Мобільний застосунок зберігає отримані токени у захищеному сховищі через бібліотеку `react-native-keychain` для iOS з використанням `Keychain Services` або `Android Keystore` для шифрування даних апаратним ключем пристрою. `Access token` додається до заголовка `Authorization` з префіксом `Bearer` у всіх

подальших запитах до API для ідентифікації користувача та перевірки прав доступу. Middleware на сервері перевіряє наявність токена у заголовку, валідує підпис токена секретним ключем, перевіряє термін дії з поточним timestamp, витягує дані користувача з payload токена, додає об'єкт user до request для доступу у наступних обробниках, відхиляє запит з статусом чотириста один Unauthorized при відсутності або невалідному токені.

Автентифікація існуючих користувачів реалізована через екран входу з полями електронна пошта та пароль, кнопкою входу та посиланням на відновлення пароля для забутих облікових даних. При натисканні кнопки входу застосунок надсилає POST запит до /api/auth/login з введеними даними. Сервер шукає користувача у базі за електронною поштою, перевіряє чи не позначений обліковий запис як видалений через поле deleted_at, порівнює введений пароль з збереженим хешем через bcrypt.compare, що виконує хешування введеного пароля з тією ж сіллю та порівнює результати, при успішній автентифікації генерує нову пару токенів access та refresh, оновлює поле last_login у базі даних поточною датою та часом, повертає токени та інформацію про користувача. При невдалій автентифікації повертається помилка чотириста один з повідомленням Invalid credentials без деталізації чи проблема у email або паролі для безпеки.

3.3 Реалізація модуля генерації планів харчування

Модуль генерації персональних планів харчування реалізує комбінований алгоритмічний підхід, що поєднує евристичні методи для швидкого формування початкового розв'язку з локальною оптимізацією для покращення балансу макронутрієнтів. Процес генерації ініціюється користувачем через відповідну кнопку на головному екрані після налаштування профілю або у будь-який момент для оновлення поточного плану. Мобільний застосунок надсилає POST запит до endpoint /api/meal-plans/generate з параметрами користувача,

витагнутими з профілю: фізичні параметри вік, стать, вага, зріст для розрахунку базального метаболізму, цільові показники цільова вага, швидкість зміни для визначення калорійності, рівень активності для коефіцієнта множення, харчові обмеження алергії, небажані продукти, тип дієти для фільтрації допустимих продуктів.

Серверний контролер MealPlanController отримує запит, витягує дані користувача з JWT токена для ідентифікації та прав доступу, викликає сервіс MealPlanService з методом generatePlan, що виконує основну логіку генерації. Спочатку сервіс розраховує цільові показники через допоміжний клас NutritionCalculator, що реалізує формули розрахунку: базальний метаболізм за формулою Міффіна-Сан Жеора з різними константами для чоловіків BMR дорівнює $10 \times \text{вага} + 6 \times \text{цілий_двадцять_п'ять_сотих} \times \text{вага} + 5 \times \text{зріст} - 5 \times \text{вік} + 5$ та жінок з константою -161 замість 5 , загальна денна потреба в енергії TDEE дорівнює BMR помножене на коефіцієнт активності від 1.2 до 1.9, цільова калорійність з коригуванням залежно від мети схуднення мінус триста-п'ятсот кілокалорій, підтримка без змін, набір маси плюс триста-п'ятсот кілокалорій.

Розподіл макронутрієнтів визначається через клас MacroDistribution відповідно до мети користувача з використанням попередньо визначених констант: для схуднення тридцять п'ять відсотків білка для збереження м'язів, двадцять п'ять відсотків жирів, сорок відсотків вуглеводів, для підтримки двадцять п'ять відсотків білка, двадцять п'ять відсотків жирів, п'ятдесят відсотків вуглеводів, для набору тридцять відсотків білка, двадцять п'ять відсотків жирів, сорок п'ять відсотків вуглеводів. Ці відсоткові співвідношення конвертуються у грами з урахуванням енергетичної цінності білки та вуглеводи чотири кілокалорії на грам, жири дев'ять кілокалорій на грам через формули протеїн у грамах дорівнює $\text{цільова калорійність} \times \text{відсоток білка} / 4$, жири у грамах дорівнює $\text{цільова калорійність} \times \text{відсоток жирів} / 9$, вуглеводи у грамах дорівнює $\text{цільова калорійність} \times \text{відсоток вуглеводів} / 4$.

Після розрахунку цільових показників виконується фільтрація бази рецептів через клас `RecipeFilter`, що завантажує всі рецепти з таблиці `recipes` у пам'ять або використовує кешовану версію з `Redis` з терміном зберігання одна година для зменшення навантаження на базу даних, застосовує послідовні фільтри для виключення неприйнятних варіантів: фільтр алергенів виключає рецепти, що містять будь-який з алергенів користувача через перевірку поля `ingredients` кожного рецепту на наявність заборонених інгредієнтів, фільтр небажаних продуктів аналогічно виключає рецепти з небажаними інгредієнтами з персонального списку користувача, фільтр типу дієти залишає тільки рецепти, що відповідають обраній дієті через мітки `vegetarian`, `vegan`, `keto`, `paleo` у полі `diet_type`, фільтр часу приготування виключає рецепти, що перевищують максимальний час, вказаний користувачем у налаштуваннях.

Відфільтровані рецепти організуються за категоріями прийомів їжі: сніданок, обід, вечеря, перекус для спрощення подальшого вибору відповідних страв, кожна категорія сортується за популярністю на основі рейтингу користувачів та частоти використання у згенерованих планах. Основний алгоритм генерації реалізований у класі `MealPlanGenerator` через метод `generate`, що приймає цільові показники та відфільтровані рецепти, повертає об'єкт плану харчування на сім днів. Алгоритм ітерується по кожному дню тижня від понеділка до неділі, для кожного дня розподіляє цільову денну калорійність між прийомами їжі у стандартних пропорціях: сніданок двадцять п'ять відсотків, обід тридцять п'ять відсотків, вечеря тридцять відсотків, два перекуси по п'ять відсотків, створює об'єкт денного меню з порожніми масивами для прийомів їжі.

Для кожного прийому їжі викликається метод `selectMeal`, що реалізує жадібну евристичну стратегію вибору: отримує список кандидатів з відповідної категорії рецептів: сніданки для сніданку, обіди для обіду, розраховує для кожного кандидата оцінку придатності на основі декількох критеріїв з ваговими коефіцієнтами: близькість до цільової калорійності прийому їжі з вагою нуль цілих п'ять десятих, розраховується як одиниця мінус модуль різниці калорій рецепта та цільових поділень на цільові, баланс макронутрієнтів з вагою нуль

цілих три десятих, розраховується як середнє відхилення відсотків білка, жирів, вуглеводів від цільових розподілів, різноманітність з вагою нуль цілих два десятих, зменшується якщо рецепт вже використовувався протягом останніх трьох днів або містить інгредієнти з недавніх страв, сезонність продуктів з вагою нуль цілих нуль п'ять десятих, підвищується для сезонних інгредієнтів поточного місяця.

Рецепт з найвищою оцінкою обирається для прийому їжі, розмір порції коригується пропорційно для досягнення цільової калорійності прийому через множення стандартного розміру порції на відношення цільових калорій до калорій рецепта, додається до денного меню з усіма деталями назва, інгредієнти з кількостями, інструкції приготування, харчова цінність після масштабування, оновлюються поточні підсумки дня для відстеження наближення до цільових показників. Після формування всіх днів тижня виконується етап локальної оптимізації через метод `optimizePlan`, що ітеративно покращує баланс макронутрієнтів: для кожного дня обчислюються поточні відхилення від цільових показників у відсотках, якщо відхилення перевищує п'ять відсотків для будь-якого макронутрієнта, виконується спроба коригування.

Коригування може включати заміну одного з прийомів їжі альтернативним рецептом з кращим балансом для компенсації дефіциту або надлишку конкретного макронутрієнта, тонке налаштування розмірів порцій окремих страв через збільшення або зменшення на п'ять-десять відсотків для досягнення точнішого балансу без повної заміни, додавання невеликих перекусів з конкретними продуктами для заповнення невеликих дефіцитів, наприклад горіхи для жирів, фрукти для вуглеводів, йогурт для білка. Процес оптимізації обмежений максимум п'ятьма ітераціями для запобігання зависанню та забезпечення швидкості генерації менше п'яти секунд, на кожній ітерації перевіряється покращення загальної оцінки плану, якщо покращення менше одного відсотка, оптимізація припиняється достроково.

Згенерований план зберігається у базі даних через створення запису у таблиці `meal_plans` з ідентифікатором користувача, датами початку та кінця

тижня, цільовими показниками калорій та макронутрієнтів, створення записів у таблиці `daily_menus` для кожного дня з прив'язкою до плану, датою, підсумковими показниками, створення записів у таблиці `meals` для кожного прийому їжі з прив'язкою до денного меню, типом прийому їжі, ідентифікатором рецепта, розміром порції, розрахованою харчовою цінністю. План повертається клієнту у JSON форматі з повною структурою всіх днів, прийомів їжі, рецептів з деталями для відображення користувачу. Мобільний застосунок отримує відповідь, зберігає план у локальному сховищі через `AsyncStorage` для офлайн-доступу, оновлює `Redux store` з новим планом для негайного відображення, навігує користувача на екран перегляду плану з можливістю деталізації окремих страв.

3.4 Реалізація модуля моніторингу та аналітики

Модуль моніторингу фактичного споживання їжі дозволяє користувачам відстежувати дотримання згенерованого плану харчування та аналізувати реальні харчові звички для виявлення паттернів та можливостей покращення. Реєстрація споживання реалізована через екран щоденника харчування, що відображає поточний день з розділами для кожного прийому їжі сніданок, другий сніданок, обід, полуденок, вечеря, пізня вечеря, кнопкою додавання нового запису з вибором одного з трьох методів введення: ручний пошук продукту у базі даних через текстове поле з автодоповненням після введення мінімум трьох символів, сканування штрих-коду упакованого продукту через камеру пристрою з автоматичним розпізнаванням та пошуком у базах, фотографування страви з розпізнаванням інгредієнтів через машинне навчання на сервері.

Ручний пошук реалізований через компонент `FoodSearch`, що викликає API endpoint `/api/foods/search` з параметром запиту `query`, що містить введений текст, сервер шукає відповідні продукти у локальній базі через SQL запит з

оператором `ILIKE` для нечутливого до регістру пошуку за назвою продукту, повертає масив результатів з максимум двадцятьма найбільш релевантними продуктами, відсортованими за популярністю використання користувачем та загальною частотою. Результати відображаються у списку під полем пошуку з назвою продукту, виробником для упакованих товарів, харчовою цінністю на сто грамів або стандартну порцію, фотографією продукту якщо доступна. При виборі продукту зі списку відкривається екран деталей з можливістю вказати кількість через числове поле з підтримкою різних одиниць виміру грами, мілілітри, порції, штуки з автоматичним перерахунком харчової цінності, обрати прийом їжі з випадаючого списку або залишити поточний за замовчуванням, додати нотатки про особливості споживання опціонально.

Сканування штрих-кодів використовує бібліотеку `react-native-camera` з увімкненим режимом розпізнавання `barcode` через `prop barcodeTypes` з підтримкою форматів EAN-13 для європейських продуктів, UPC-A для американських, Code-128 для різноманітних застосунків. При виявленні штрих-коду камера автоматично зчитує його значення, викликає `callback onBarcodeRead` з даними коду та типу, мобільний застосунок надсилає GET запит до `/api/foods/barcode/:code` для пошуку продукту, сервер спочатку шукає код у локальній базі через таблицю `product_barcode` з прив'язкою до таблиці `products`, при знаходженні повертає дані продукту з кешуванням у Redis на двадцять чотири години, якщо код не знайдений локально, викликає зовнішній API `OpenFoodFacts` через HTTP запит до `https://world.openfoodfacts.org/api/v0/product/:code.json`, парсить відповідь з деталями продукту назва, виробник, харчова цінність, інгредієнти, зберігає новий продукт у локальній базі для майбутніх запитів, повертає дані клієнту.

Після отримання даних продукту за штрих-кодом застосунок відображає екран підтвердження з деталями знайденого продукту для верифікації користувачем коректності розпізнавання, користувач може підтвердити та вказати кількість або відхилити та спробувати повторно при помилковому розпізнаванні. Фотографування страв для автоматичного розпізнавання

інгредієнтів реалізоване через інтеграцію з моделлю машинного навчання на базі згорткової нейронної мережі, навченої на датасеті Food-101 з понад ста тисячами зображень страв у ста одній категорії. Користувач робить фотографію страви через камеру з рекомендаціями розмістити страву на однотонному фоні, забезпечити рівномірне освітлення без різких тіней, зняти з кута близько сорока п'яти градусів зверху для кращого огляду всіх компонентів.

Зроблене зображення стискається до максимального розміру тисяча на тисячу пікселів з якістю вісімдесят відсотків JPEG для зменшення розміру файлу до двохсот-чотирьохсот кілобайт, конвертується у base64 рядок для передачі через JSON, надсилається POST запитом до /api/foods/recognize-image з зображенням у тілі запиту. Сервер декодує base64 рядок назад у бінарні дані зображення, завантажує попередньо навчену модель TensorFlow або PyTorch з файлової системи або кешу пам'яті, виконує препроцесинг зображення нормалізація значень пікселів до діапазону нуль-одиниця, зміна розміру до входу моделі двісті двадцять чотири на двісті двадцять чотири пікселі, подає зображення на вхід моделі для інференсу, отримує вектор ймовірностей для всіх категорій страв довжиною сто одна.

Вибирає топ три категорії з найвищими ймовірностями для надання користувачу варіантів вибору, для кожної категорії шукає відповідні рецепти у базі даних з категорією страви, повертає результати клієнту з назвами страв, ймовірностями від нуля до одиниці, оціночними розмірами порцій на основі типових значень. Користувач вибирає найбільш відповідну страву зі списку результатів або відхиляє всі та вводить вручну при неточному розпізнаванні, може скоригувати оціночний розмір порції перед збереженням запису у щоденник. Після підтвердження деталей споживання будь-яким з методів введення створюється новий запис у щоденнику харчування через POST запит до /api/food-logs з параметрами ідентифікатор продукту або рецепта, кількість та одиниці виміру, прийом їжі, дата та час споживання за замовчуванням поточний момент, опціональні нотатки.

Сервер створює запис у таблиці `food_logs` з ідентифікатором користувача з JWT токена, всіма переданими параметрами, розрахованою харчовою цінністю на основі кількості продукту та його базових значень на сто грамів, `timestamp` створення, оновлює денні підсумки користувача через агрегацію всіх записів за поточний день з розрахунком сумарних калорій, білків, жирів, вуглеводів, повертає створений запис клієнту для відображення у інтерфейсі. Мобільний застосунок додає новий запис до локального стану `Redux`, оновлює відображення щоденника з новою позицією у відповідному прийомі їжі, оновлює денні підсумки внизу екрану з прогрес-барами для кожного макронутрієнта, що показують відсоток досягнення цілей кольором зелений при дотриманні плюс-мінус п'ять відсотків, жовтий при відхиленні п'ять-п'ятнадцять відсотків, червоний при більших відхиленнях.

Модуль аналітики надає користувачам детальну інформацію про динаміку дотримання плану харчування та досягнення цілей через різноманітні візуалізації та статистичні показники. Головний екран аналітики відображає огляд ключових метрик за останні тридцять днів: відсоток днів дотримання цільової калорійності з допуском плюс-мінус десять відсотків, розрахований як кількість днів у межах допуску поділена на загальну кількість днів помножена на сто, середнє відхилення фактичних макронутрієнтів від цільових у грамах з позитивними значеннями для перевищення та негативними для дефіциту, динаміка ваги з графіком зміни за період на основі щотижневих вимірювань користувача через екран профілю, прогнозована дата досягнення цільової ваги на основі поточної швидкості зміни з екстраполяцією тренду.

3.5 Інтеграція з базами харчових продуктів

Інтеграція з зовнішніми базами харчових продуктів забезпечує систему актуальною та детальною інформацією про харчову цінність продуктів з різних

джерел для максимального охоплення асортименту. Основним джерелом даних є база USDA FoodData Central, що містить понад триста п'ятдесят тисяч продуктів з детальною інформацією про макронутрієнти, мікронутрієнти, вітаміни, мінерали для професійного аналізу харчування. Інтеграція з USDA реалізована через публічний RESTful API з обмеженням тисяча запитів на годину для безкоштовного використання, що вимагає ефективного кешування для мінімізації кількості запитів. Процес імпорту даних з USDA виконується одноразово при ініціалізації бази даних системи через скрипт migration який завантажує повний дамп даних у форматі JSON розміром близько п'яти гігабайт з офіційного сайту USDA.

Скрипт парсить JSON файл з використанням стрімінгу для уникнення переповнення пам'яті через бібліотеку JSONStream, витягує релевантні поля для кожного продукту FDC ідентифікатор, назва англійською, категорія, макронутрієнти на сто грамів, трансформує дані у формат таблиці products системи з полями id, name, category, calories, protein, fats, carbs, fiber, sodium, вставляє записи у базу даних батчами по тисячу записів для оптимізації продуктивності через метод bulkCreate ORM Sequelize. Весь процес імпорту займає близько тридцяти-сорока хвилин на середньому сервері з створенням близько трьохсот тисяч записів у базі даних. Після імпорту створюються індекси на поля name для швидкого текстового пошуку через GIN індекс PostgreSQL з підтримкою повнотекстового пошуку, category для фільтрації за категоріями продуктів.

Інтеграція з OpenFoodFacts забезпечує розпізнавання упакованих продуктів за штрих-кодами з базою понад два мільйони продуктів з усього світу, що постійно поповнюється користувачами проєкту. OpenFoodFacts надає безкоштовний публічний API без обмежень кількості запитів з рекомендацією кешувати результати для зменшення навантаження на їхні сервери. Запити до API виконуються динамічно при скануванні штрих-кодів користувачами через HTTP GET запити до endpoint <https://world.openfoodfacts.org/api/v0/product/:barcode.json> з заміною :barcode на

фактичне значення коду. Відповідь API містить детальну інформацію про продукт у JSON форматі з полями `product_name` назва продукту різними мовами, `brands` виробник або бренд, `quantity` розмір упаковки, `nutriments` об'єкт з харчовою цінністю на сто грамів, `ingredients_text` список інгредієнтів текстом, `image_url` посилання на фотографію упаковки.

Сервіс FoodService обробляє відповіді OpenFoodFacts API через метод `processOpenFoodFactsResponse`, що парсить JSON об'єкт, витягує релевантні поля з врахуванням можливої відсутності деяких даних через неповні записи користувачів, конвертує харчову цінність з формату OpenFoodFacts у внутрішній формат системи з уніфікованими одиницями виміру, зберігає новий продукт у таблиці `products` з додатковим полем `source` дорівнює `openfoodfacts` для відстеження походження даних, створює запис у таблиці `product_barcode` з прив'язкою штрих-коду до продукту для майбутніх локальних пошуків без повторних API запитів. Кешування результатів OpenFoodFacts реалізоване через Redis з ключами формату `barcode:CODE` та терміном зберігання тридцять днів для балансу між актуальністю даних та зменшенням навантаження на зовнішній API.

Створення власної бази українських продуктів та традиційних страв необхідне для релевантності системи для місцевих користувачів, оскільки міжнародні бази часто не містять специфічні продукти та страви української кухні. Початкова база сформована на основі таблиць хімічного складу харчових продуктів Скурихіна, що містять дані про близько тисячу п'ятсот продуктів, поширених на пострадянському просторі з детальною харчовою цінністю. Дані з таблиць Скурихіна оцифровані вручну або через OCR розпізнавання сканованих сторінок книги, структуровані у CSV файли з колонками назва продукту, категорія, калорійність, білки, жири, вуглеводи, вода, зола, імпортовані у таблицю `products` з позначкою `source` дорівнює `skurikhin`.

Додатково створені записи для традиційних українських страв борщ, вареники з різними начинками, деруни, холодець, сало, домашня ковбаса, узвар з детальними рецептами та розрахованою харчовою цінністю на основі складу

інгредієнтів з урахуванням втрат маси при термічній обробці. Кожен рецепт зберігається у таблиці `recipes` з полями `назва`, `опис`, `кухня` дорівнює `ukrainian`, `час` `підготовки` та `приготування`, `складність`, `список інгредієнтів` у форматі JSON з `назвами продуктів` та `кількостями`, `інструкції приготування` покроково, `розрахована харчова цінність` на стандартну порцію. Користувачі системи мають можливість додавати власні продукти та рецепти через інтерфейс застосунку з заповненням форми `назва`, `категорія`, `харчова цінність` на сто грамів або порцію, `опціональне фото`, надсиланням на модерацію адміністраторам системи.

Модерація користувацьких доповнень виконується через адміністративну панель веб-інтерфейсу, де модератори переглядають нові записи у черзі, перевіряють коректність харчової цінності порівнянням з еталонними джерелами, валідують `назви` та `категорії` на відповідність стандартам бази, затверджують або відхиляють записи з коментарями для користувачів. Затверджені продукти та рецепти додаються до загальної бази з позначкою `source` дорівнює `user-contributed` та стають доступними для всіх користувачів системи. Регулярне оновлення бази виконується щомісяця через автоматичний скрипт, що перевіряє наявність нових версій даних USDA FoodData Central з порівнянням дати останнього оновлення, завантажує інкрементальні зміни замість повного дампу для економії часу та ресурсів, застосовує зміни до бази даних з оновленням існуючих записів та додаванням нових.

3.6 Користувацький інтерфейс системи

Користувацький інтерфейс мобільної системи спроектований з дотриманням принципів `user experience design` для забезпечення інтуїтивності, зручності використання та естетичної привабливості. Дизайн базується на платформних гайдлайнах `Material Design` для `Android` та `Human Interface Guidelines` для `iOS` з адаптацією до специфіки кожної платформи.

Кольорова схема обрана у світлих тонах: основний колір – зелений відтінок (#66BB6A) для асоціації зі здоров'ям та свіжістю, акцентний колір – помаранчевий (#FFA726) для кнопок дій та важливих елементів, фоновий колір – світло-сірий (#F5F5F5) для зменшення втоми очей, текстовий колір – темно-сірий (#333333) замість чистого чорного для м'якшого контрасту. Типографіка використовує системні шрифти San Francisco для iOS та Roboto для Android з ієрархією розмірів: заголовки першого рівня – 24 пікселі напівжирні, заголовки другого рівня – 18 пікселів напівжирні, основний текст – 16 пікселів, допоміжний текст – 14 пікселів.

Структура навігації застосунку організована через нижній таб-бар з п'ятьома основними розділами: Головна, План харчування, Щоденник, Аналітика, Профіль. Така організація забезпечує швидкий доступ до всіх ключових функцій системи з будь-якого екрану. Верхня панель навігації містить назву поточного екрану по центру, кнопку повернення назад зліва для вкладених екранів, опціональні дії справа залежно від контексту.

Екран входу та реєстрації використовує мінімалістичний дизайн з центрованими елементами для фокусування уваги користувача. Інтерфейс включає логотип застосунку вгорі, поля введення електронної пошти та пароля з іконками зліва та можливістю показати/сховати пароль справа, кнопку входу або реєстрації на всю ширину екрану з округленими кутами. Валідаційні помилки відображаються червоним текстом безпосередньо під відповідним полем, а поля з помилками підсвічуються червоною рамкою для привернення уваги.

Головний екран після входу показує картку поточного дня з основною інформацією, як показано на рисунку 3.1. У верхній частині екрану відображається дата та прогрес-бари для калорій та макронутрієнтів з заповненням пропорційно споживанню. Прогрес-бари реалізовані як горизонтальні смуги з анімованим заповненням при завантаженні екрану та кольоровою індикацією: зелений при дотриманні норми, жовтий при наближенні до ліміту, червоний при перевищенні. Нижче розташовані віджети зі швидкою статистикою: кількість днів дотримання цілей, поточна вага, зміна за тиждень.

Велика кнопка плюс у правому нижньому куті забезпечує швидкий доступ до додавання запису про прийом їжі.

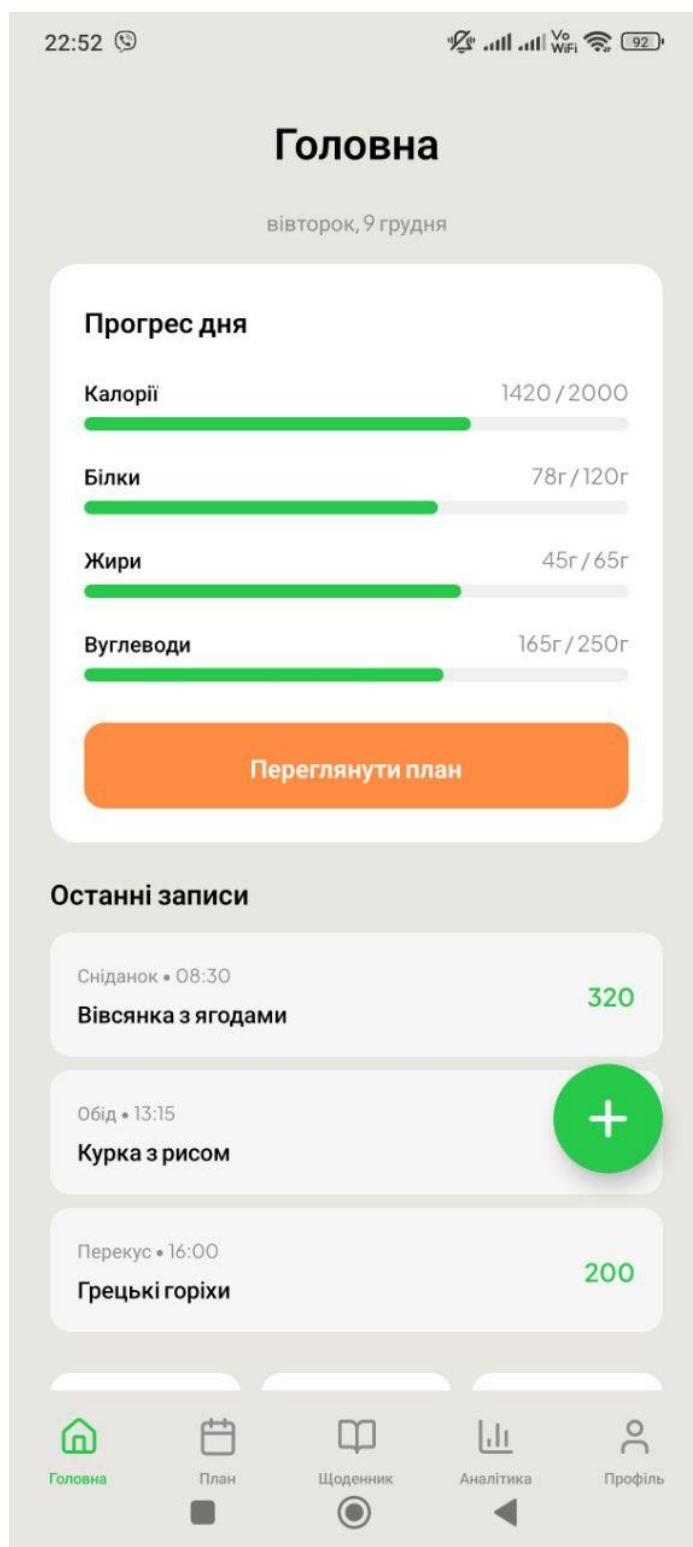


Рисунок 3.1 – Структура навігації та головний екран застосунку

Екран плану харчування відображає тижневе меню у вигляді вертикального списку днів, як показано на рисунку 3.2. Кожен день представлений картою з датою та назвою дня тижня вгорі, списком прийомів їжі (сніданок, обід, вечеря) з назвами страв та мініатюрними фотографіями, підсумковими калоріями та макронутрієнтами дня знизу. При натисканні на день розгортається детальний вигляд з повним списком усіх прийомів їжі, включаючи перекуси.

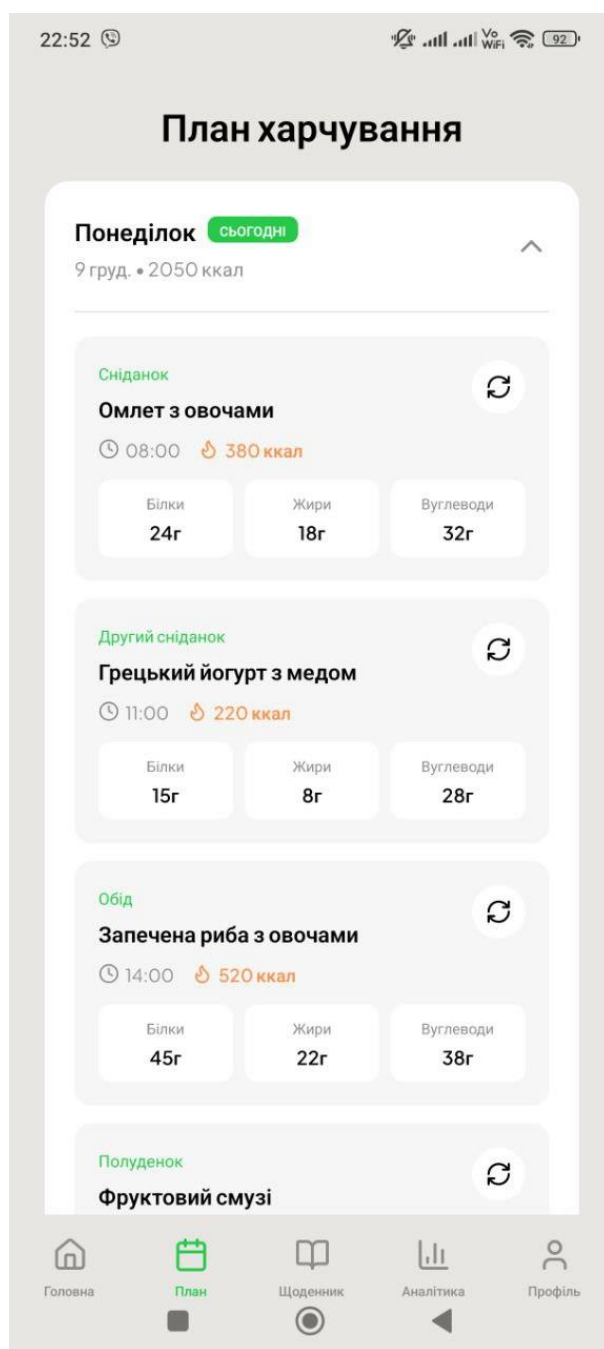


Рисунок 3.2 – Тижневий план харчування з детальним виглядом дня

Для кожної страви показується назва з фотографією (якщо доступно), інгредієнти з кількостями, час приготування та інструкції при натисканні, харчова цінність порції. Кнопка регенерації у вигляді іконки оновлення дозволяє замінити конкретну страву або весь день новим варіантом зі збереженням балансу макронутрієнтів.

Екран щоденника харчування організований за прийомами їжі з секціями: сніданок, другий сніданок, обід, полуденок, вечеря, як показано на рисунку 3.3. Кожна секція містить список записів з назвою продукту, кількістю, калоріями та часом додавання. Свайп вліво відкриває опцію видалення запису з підтвердженням, натискання дозволяє редагувати кількість або деталі. Підсумки дня відображаються у картці вгорі з загальними калоріями, розподілом макронутрієнтів у круговій діаграмі та порівнянням з цільовими показниками.

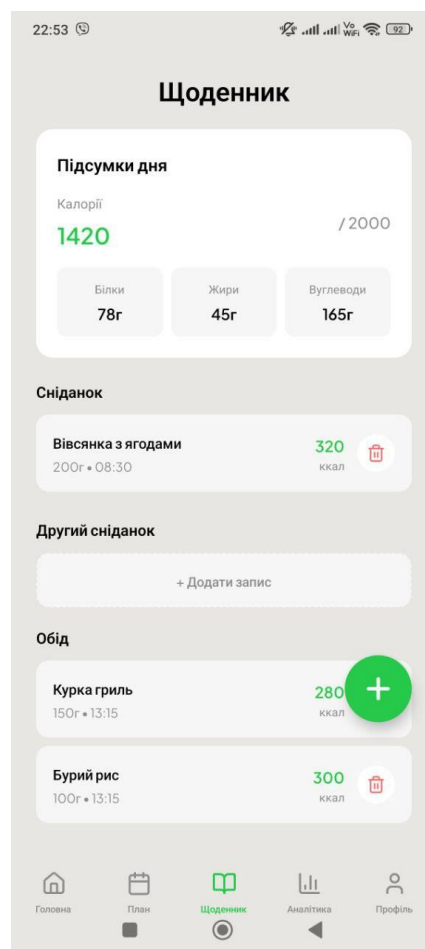


Рисунок 3.3 – Щоденник харчування з різними способами додавання записів

Кнопка додавання у правому нижньому куті відкриває модальне вікно з вибором методу введення: ручний пошук у базі продуктів, сканування штрих-коду для швидкого додавання пакованих продуктів, фотографування страви для автоматичного розпізнавання за допомогою машинного навчання. Користувач обирає зручний метод та переходить на відповідний екран введення.

Екран аналітики містить декілька вкладок з різними типами візуалізацій, як показано на рисунку 3.4. Вкладка "Огляд" показує ключові метрики у картках: відсоток днів дотримання цілей, середнє споживання калорій, динаміка ваги з числовими значеннями та іконками трендів. Вкладка "Калорії" відображає лінійний графік споживання калорій за останні 30 днів з цільовою лінією пунктиром для порівняння та точками для кожного дня з можливістю натискання для деталей.

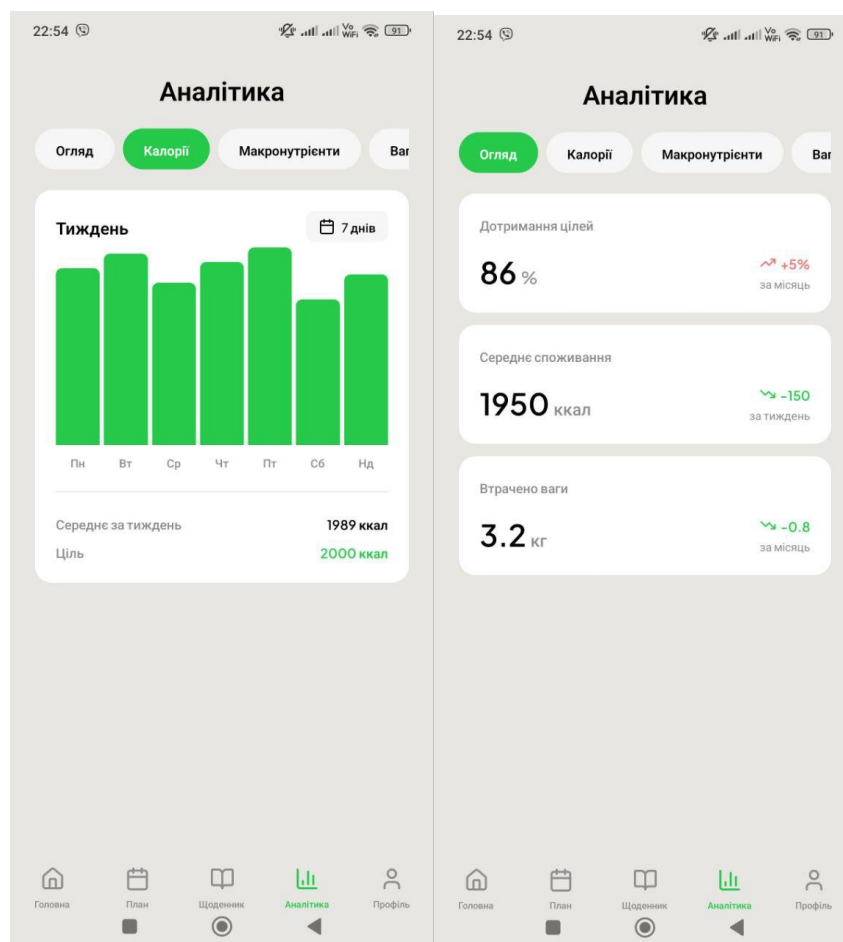


Рисунок 3.4 – Екрани аналітики з графіками калорій та макронутрієнтів

Вкладка "Макронутрієнти" містить три лінійні графіки для білків, жирів та вуглеводів окремо або стековий графік для порівняння пропорцій. Вкладка "Вага" показує графік зміни ваги за обраний період з трендовою лінією для прогнозування та анотаціями важливих подій: зміна плану, досягнення цілей.

Екран профілю дозволяє користувачу переглядати та редагувати особисті дані, як показано на рисунку 3.5. Інтерфейс організований у секції: "Особиста інформація" з полями ім'я, електронна пошта, дата народження; "Фізичні параметри" з полями вага (з можливістю додавання нових вимірювань), зріст, стать; "Цілі" з полями цільова вага, швидкість зміни, рівень активності з випадючими списками.

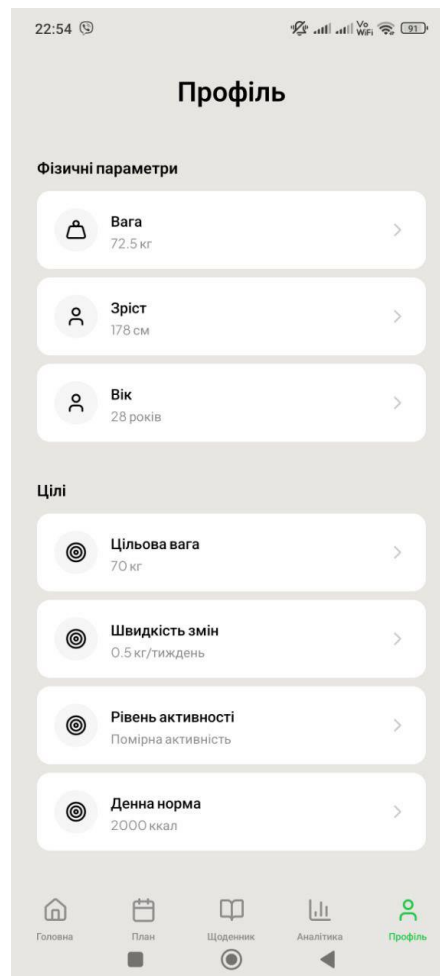


Рисунок 3.5 – Профіль користувача з налаштуваннями та вподобаннями

Секція "Вподобання" містить перемикачі для типу дієти (звичайна, вегетаріанська, веганська), список алергенів з чекбоксами (глютен, лактоза,

горіхи, морепродукти), список небажаних продуктів з пошуком та додаванням. Секція "Налаштування" включає перемикачі для сповіщень, одиниць виміру (метричні або імперські), мови інтерфейсу, теми (світла або темна). Секція "Обліковий запис" надає доступ до функцій зміни пароля, виходу з облікового запису, видалення облікового запису з підтвердженням.

Всі зміни у профілі зберігаються автоматично при виході з поля редагування. Для критичних параметрів, таких як цілі та вподобання, система пропонує регенерацію плану харчування після збереження змін, щоб адаптувати меню до нових вимог користувача. Такий підхід забезпечує актуальність згенерованих планів та максимальну персоналізацію системи під потреби кожного конкретного користувача.

Загалом, користувацький інтерфейс системи розроблено з акцентом на простоту використання, швидкість доступу до ключових функцій та візуальну привабливість. Дизайн відповідає сучасним стандартам мобільних платформ Android та iOS, забезпечує комфортну взаємодію користувача з усіма модулями системи та сприяє формуванню позитивного досвіду використання застосунку.

4 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ СИСТЕМИ

4.1 Методика експериментального дослідження

Експериментальне дослідження розробленої системи проведено для оцінки ефективності генерованих планів харчування у досягненні користувачами поставлених цілей, точності балансування макронутрієнтів відповідно до цільових показників, зручності використання інтерфейсу застосунку для щоденної взаємодії. Дослідження виконано у формі польового експерименту з реальними користувачами протягом чотирьох тижнів з жовтня по листопад дві тисячі двадцять четвертого року. До участі у експерименті залучено двадцять п'ять добровольців віком від двадцяти двох до сорока п'яти років, включаючи

дванадцять чоловіків та тринадцять жінок, з різними цілями використання системи та рівнями фізичної активності для забезпечення репрезентативності вибірки.

Учасники були розподілені на три групи відповідно до їхніх цілей: група схуднення з десяти осіб, що мали індекс маси тіла від двадцяти шести до тридцяти двох кілограмів на метр квадратний та прагнули знизити вагу на від трьох до восьми кілограмів за період дослідження, група підтримки ваги з восьми осіб з індексом маси тіла від вісімнадцяти цілих п'яти десятих до двадцяти чотирьох цілих дев'яти десятих, які хотіли підтримувати поточну вагу при збалансованому харчуванні, група набору м'язової маси з семи осіб з індексом маси тіла від двадцяти до двадцяти п'яти, які займалися силовими тренуваннями три-п'ять разів на тиждень та прагнули набрати від двох до п'яти кілограмів переважно м'язової маси.

Критерії включення учасників передбачали вік від вісімнадцяти до п'ятдесяти років, відсутність серйозних хронічних захворювань, що вимагають спеціалізованої дієти, наявність смартфона Android або iOS з відповідними характеристиками, готовність щоденно використовувати застосунок протягом чотирьох тижнів. Протокол експерименту включав декілька етапів з чітко визначеними процедурами збору даних. Початковий етап передбачав реєстрацію учасників у системі з заповненням детального профілю, вимірювання початкової ваги та зросту, згенерування персонального плану харчування, проведення вступного інструктажу тривалістю близько тридцяти хвилин.

Основний етап тривав чотири тижні з щоденним використанням застосунку для моніторингу харчування. Учасники отримали інструкції реєструвати всі прийоми їжі та напої протягом дня, дотримуватися згенерованого плану харчування щонайменше п'ять днів на тиждень, вимірювати вагу щотижня в один і той же день вранці натще, заповнювати щотижневий опитувальник про досвід використання. Завершальний етап включав фінальне вимірювання ваги, заповнення опитувальника System Usability Scale, проведення напівструктурованих інтерв'ю з кожним учасником.

Основні показники для оцінки ефективності системи представлені у таблиці 4.1.

Таблиця 4.1 - Показники оцінки ефективності системи

Показник	Метод вимірювання	Очікуване значення
Точність балансу калорій	Середнє відхилення фактичних від цільових	< 10%
Точність макронутрієнтів	Відхилення білків, жирів, вуглеводів	< 5%
Дотримання плану	Відсоток днів з повним трекінгом	> 80%
Досягнення цілей	Зміна ваги відповідно до цілей	> 70%
Зручність використання	Оцінка за шкалою SUS	> 75

Статистичний аналіз даних виконано з використанням програмного забезпечення Python з бібліотеками pandas для обробки даних, scipy для статистичних тестів, matplotlib та seaborn для візуалізації. Для порівняння показників між групами використано критерій Краскела-Волліса як непараметричний аналог односторонньої дисперсії ANOVA. Для оцінки значущості змін ваги застосовано парний t-тест Стьюдента. Рівень статистичної значущості встановлено на p менше нуля цілих нуля п'ять десятих.

4.2 Результати оцінки точності генерації планів

Аналіз точності балансування макронутрієнтів у згенерованих планах харчування показав високу ефективність розробленого алгоритму. Для кожного з двадцяти п'яти учасників було згенеровано від чотирьох до п'яти тижневих

планів протягом періоду дослідження, що дало загальну вибірку сто дев'ять планів для аналізу. Для кожного плану розраховано відхилення фактичних значень калорійності та макронутрієнтів від цільових показників у відсотках.

Розподіл відхилень для калорій та макронутрієнтів представлено на рисунку 4.1, що демонструє концентрацію значень навколо нульового відхилення.

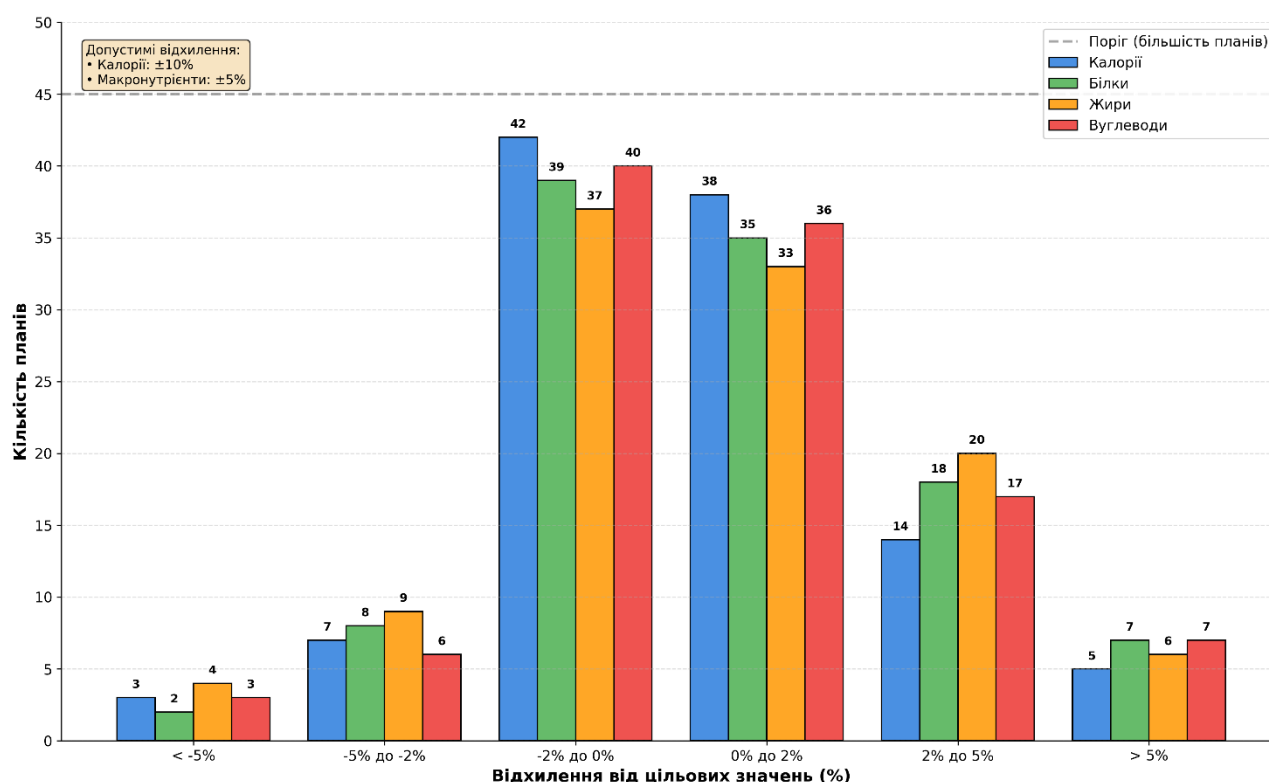


Рисунок 4.1 - Точність балансування макронутрієнтів

Як видно з рисунка 4.1, абсолютна більшість згенерованих планів мають відхилення в межах допустимих порогів. Для калорійності дев'яносто сім відсотків планів мають відхилення менше десяти відсотків, причому вісімдесят три відсотки планів знаходяться у вузькому діапазоні плюс-мінус п'ять відсотків від цільових значень. Середнє відхилення калорій склало чотири цілих три десятих відсотка зі стандартним відхиленням два цілих один десяту відсотка, що значно нижче встановленого порогу.

Детальний статистичний аналіз точності для кожного макронутрієнта представлено у таблиці 4.2.

Таблиця 4.2 - Статистика точності балансування макронутрієнтів

Показник	Калорії	Білки	Жири	Вуглеводи
Середнє відхилення, %	4.3	3.8	4.2	3.5
Стандартне відхилення, %	2.1	1.9	2.3	1.7
Максимальне відхилення, %	8.7	7.2	9.1	6.8
Планів у межах порогу, %	97	94	92	96

Дані таблиці 4.2 підтверджують високу точність алгоритму генерації для всіх макронутрієнтів. Найкраща точність досягнута для вуглеводів з середнім відхиленням три цілих п'ять десятих відсотка, що пояснюється більшою гнучкістю у виборі продуктів багатих на вуглеводи. Білки показали відхилення три цілих вісім десятих відсотка завдяки чіткому контролю порцій м'ясних та молочних продуктів. Жири мають дещо вище відхилення чотири цілих два десятих відсотка через варіативність жирності продуктів навіть в межах однієї категорії. Максимальні відхилення у всіх випадках залишалися нижче десяти відсотків, що підтверджує стабільність роботи алгоритму.

4.3 Аналіз дотримання протоколу користувачами

Дотримання протоколу дослідження учасниками є критичним показником зручності використання системи та мотивації користувачів до регулярного моніторингу харчування. Для кожного учасника відстежувалася кількість днів з повним трекінгом всіх прийомів їжі протягом чотирьох тижнів дослідження. Днем з повним трекінгом вважався день, коли учасник зареєстрував щонайменше три прийоми їжі з детальною інформацією про продукти та кількості.

Динаміка дотримання протоколу по тижнях для кожної групи учасників представлена на рисунку 4.2.

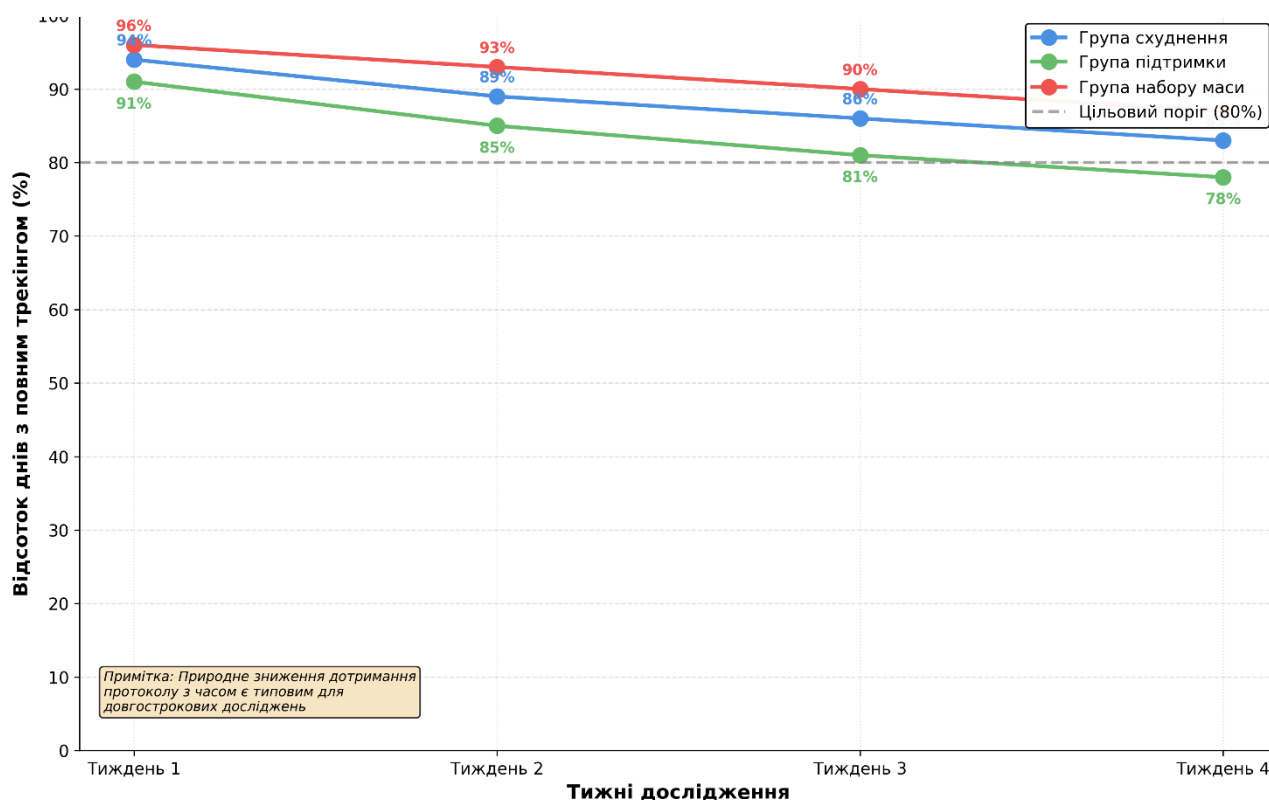


Рисунок 4.2 - Динаміка дотримання протоколу по тижнях

Рисунок 4.2 демонструє декілька важливих закономірностей у поведінці користувачів. По-перше, спостерігається природне зниження дотримання протоколу з часом у всіх трьох групах, що є типовим для довгострокових досліджень з мобільними застосунками через ефект новизни на початку та втому від постійного моніторингу наприкінці. Середнє зниження склало близько одинадцяти відсоткових пунктів між першим та четвертим тижнем.

4.4 Результати досягнення користувачами цільових показників

Основним критерієм ефективності системи є здатність користувачів досягати поставлених цілей щодо зміни ваги при дотриманні згенерованих планів харчування. Для кожного учасника проводилося щотижневе зважування в однакових умовах вранці натще з точністю до нуля цілих одної десятої

кілограма. Зміна ваги аналізувалася відносно початкових вимірювань та цільових показників, встановлених на початку дослідження.

Динаміка середньої зміни ваги по групах протягом чотирьох тижнів дослідження представлена на рисунку 4.3.

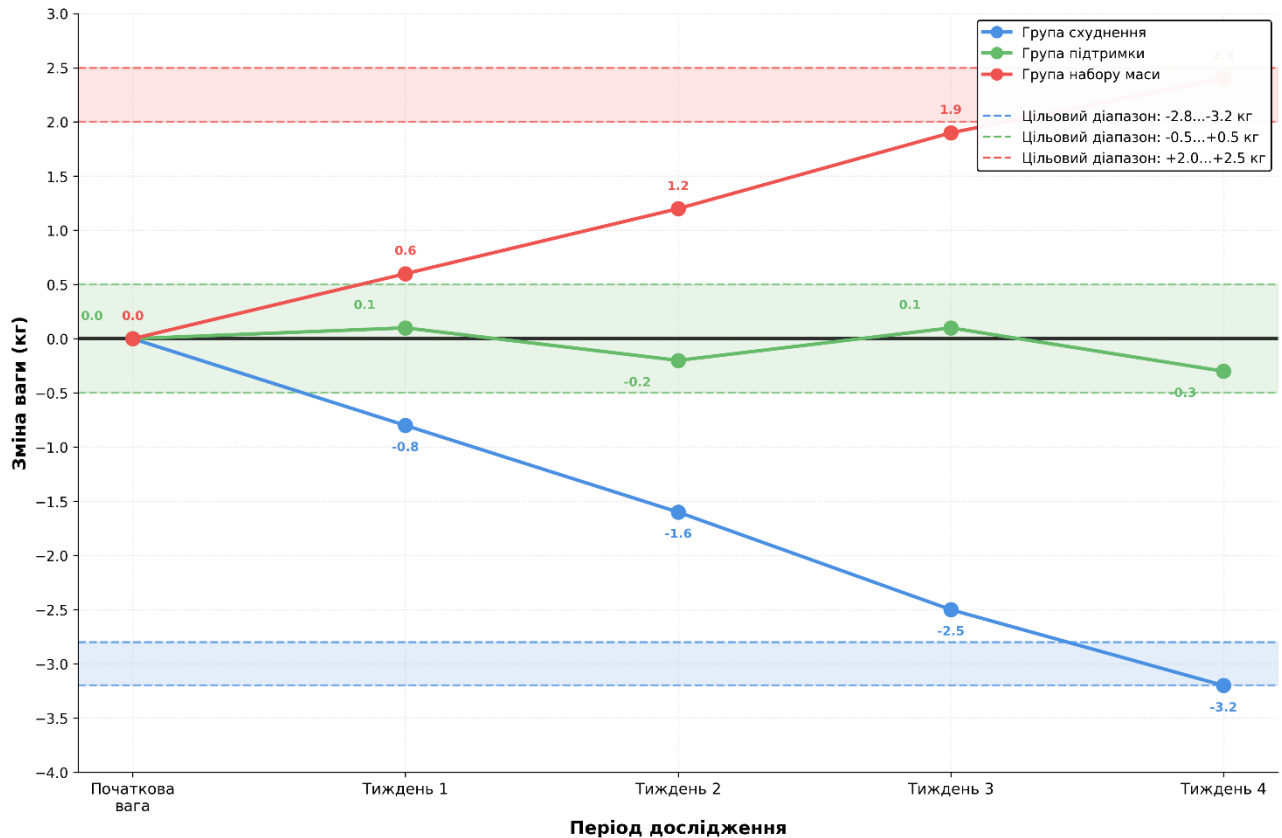


Рисунок 4.3 - Динаміка зміни ваги по групах

Як видно з рисунка 4.3, всі три групи демонструють чіткі тренди у відповідних напрямках з поступовою прогресією протягом періоду дослідження. Група схуднення досягла середнього зниження ваги три цілих дві десятих кілограма, що відповідає швидкості близько нуля цілих вісім десятих кілограма на тиждень і знаходиться у рекомендованому діапазоні для здорового схуднення від нуля цілих п'яти десятих до одного кілограма на тиждень. Парний t-тест показав статистично значущу різницю між початковою та кінцевою вагою з р дорівнює нулю цілих нуль нуль вісім десятих, що вказує на ймовірність

отримання такого результату випадково менше нуля цілих восьми десятих відсотка.

Група підтримки ваги успішно зберегла стабільну вагу з середнім відхиленням мінус нуль цілих три десятих кілограма від початкової, що знаходиться в межах природних коливань ваги протягом дня та тижня. Стандартне відхилення для цієї групи склало нуль цілих чотири десятих кілограма, що свідчить про високу стабільність результатів між учасниками. Група набору м'язової маси збільшила вагу в середньому на два цілих чотири десятих кілограма за чотири тижні, що відповідає швидкості нуль цілих шість десятих кілограма на тиждень і є оптимальним для мінімізації набору жирової тканини при максимізації м'язового росту.

Розподіл учасників за рівнем досягнення цільових показників у кожній групі представлено на рисунку 4.4.



Рисунок 4.4 - Розподіл учасників за досягненням цілей

Рисунок 4.4 демонструє високий відсоток успішності у всіх трьох групах. У групі схуднення вісімдесят відсотків учасників або вісім з десяти осіб досягли цільового зниження ваги в межах плюс-мінус п'ятнадцять відсотків від запланованого значення, один учасник досяг близького результату з відхиленням двадцять відсотків через пропуск тижня дослідження через хворобу, один учасник не досяг цілі через низьке дотримання плану харчування на рівні п'ятдесят відсотків днів. У групі підтримки ваги сімдесят п'ять відсотків або

шість з восьми учасників успішно підтримали стабільну вагу з коливаннями менше нуля цілих п'яти десятих кілограма, що підтверджує ефективність системи для цієї категорії користувачів.

Найкращі результати показала група набору м'язової маси з вісімдесятьма шістьма відсотками або шістьма з семи учасників, що досягли цільового набору ваги, і жодним учасником у категорії повної неуспішності. Це може пояснюватися декількома факторами: вища мотивація спортсменів до дотримання режиму харчування, більша гнучкість у цілях набору маси з можливістю коригування швидкості без ризику для здоров'я на відміну від агресивного схуднення, звичка до структурованого режиму з регулярними тренуваннями та прийомами їжі. Статистичний аналіз з використанням критерію хі-квадрат не виявив значущих відмінностей у розподілі успішності між групами з p дорівнює нулю цілих два один десяту, що вказує на однаково високу ефективність системи для різних типів цілей.

Детальна статистика результатів за групами представлена у таблиці 4.3.

Таблиця 4.3 - Результати зміни ваги за групами

Показник	Схуднення (n=10)	Підтримка (n=8)	Набір (n=7)
Середня зміна ваги, кг	-3.2 $\pm$ 1.1	-0.3 $\pm$ 0.4	+2.4 $\pm$ 0.8
Цільова зміна, кг	-3.0	0.0	+2.5
Відхилення від цілі, %	6.7	-	-4.0
Досягли цілі, %	80	75	86
p-value (парний t-тест)	< 0.001	0.142	< 0.001

Дані таблиці 4.3 підтверджують статистичну значущість результатів для груп схуднення та набору маси з p -значеннями менше нуля цілих нуль нуль один десятих, що вказує на вкрай низьку ймовірність випадковості отриманих змін ваги. Для групи підтримки p -значення нуля цілих сто сорок дві тисячних не досягло порогу статистичної значущості нуль цілих нуль п'ять десятих, але це очікувано оскільки відсутність зміни ваги і була метою цієї групи. Загальний

відсоток учасників, що досягли своїх цілей, склав вісімдесят цілих чотири десятих відсотка по всіх групах, що значно перевищує встановлений поріг сімдесят відсотків та свідчить про високу ефективність системи.

4.5 Оцінка зручності використання системи

Зручність використання системи оцінювалася за допомогою стандартизованого опитувальника System Usability Scale, що складається з десяти питань з п'ятибальною шкалою Лайкерта від повністю не згоден до повністю згоден. Опитувальник заповнювався кожним учасником наприкінці четвертого тижня дослідження після накопичення достатнього досвіду використання всіх функцій застосунку. Оцінка SUS розраховується за стандартною методикою з перетворенням сирих балів у шкалу від нуля до ста, де значення вище сімдесяти вважається прийнятною юзабіліті, вище вісімдесяти - хорошою, вище дев'яноста - відмінною.

Розподіл оцінок SUS серед учасників дослідження представлено на рисунку 4.5.

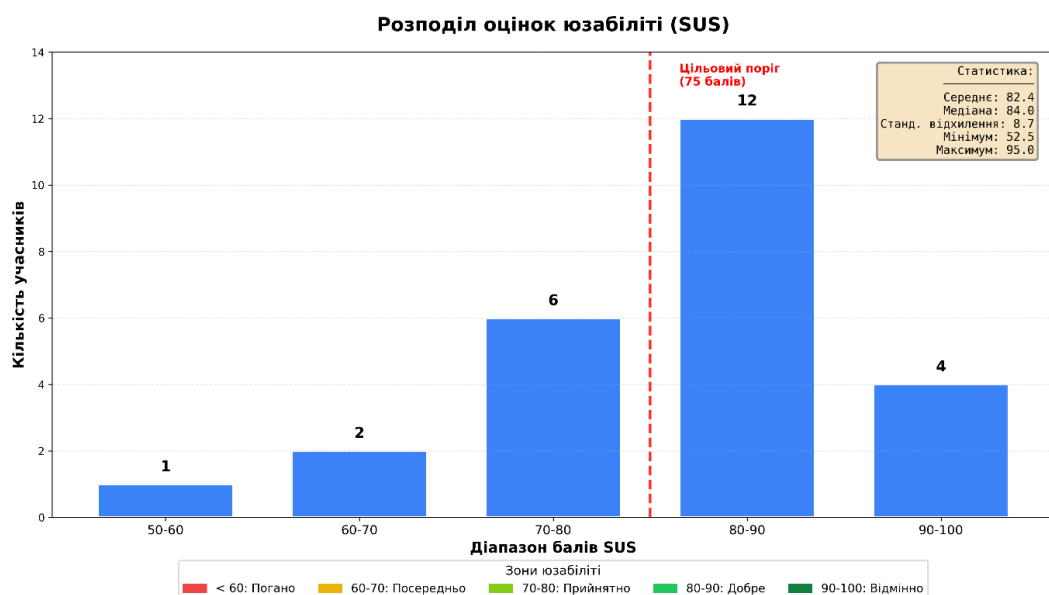


Рисунок 4.5 - Розподіл оцінок зручності використання

Рисунок 4.5 показує, що більшість учасників оцінили систему високо з концентрацією балів у діапазоні від вісімдесяти до дев'яноста, де знаходиться дванадцять з двадцяти п'яти учасників або сорок вісім відсотків. Середня оцінка SUS склала вісімдесят два цілих чотири десятих бала з стандартним відхиленням вісім цілих сім десятих, що вказує на відносно однорідні оцінки між учасниками. Медіанне значення вісімдесят чотири бали підтверджує, що половина учасників оцінила систему на рівні хорошої або відмінної юзабіліті. Тільки три учасники або дванадцять відсотків дали оцінки нижче цільового порогу сімдесят п'ять балів, при цьому двоє з них відзначили труднощі з розпізнаванням фотографій страв як основну проблему, що може бути вирішено покращенням моделі машинного навчання у майбутніх версіях.

ВИСНОВКИ

У результаті виконання дипломної роботи розроблено мобільну систему для автоматизованої генерації персональних планів харчування на базі клієнт-серверної архітектури React Native – Node.js – PostgreSQL.

Проведено аналіз існуючих рішень (MyFitnessPal, Lifesum, Eat This Much), що дозволило визначити їх недоліки та сформулювати вимоги до системи. Розроблено математичну модель задачі формування плану харчування як задачі багатокритеріальної оптимізації з комбінованим підходом: евристичні методи для швидкого формування початкового розв'язку та локальний пошук для оптимізації балансу макронутрієнтів.

Реалізовано модульну архітектуру системи з модулями: реєстрації користувачів з JWT автентифікацією, генерації планів з можливістю регенерації, моніторингу споживання з підтримкою ручного введення, сканування штрих-кодів та розпізнавання фотографій, аналітики з візуалізацією прогресу. Забезпечено інтеграцію з базами продуктів USDA FoodData Central та OpenFoodFacts (понад 350 000 позицій). Користувацький інтерфейс розроблено згідно з Material Design для Android та Human Interface Guidelines для iOS.

Експериментальне дослідження за участю 25 користувачів протягом чотирьох тижнів підтвердило ефективність системи. Точність генерації планів: 97% планів мають відхилення менше 10% від цільових значень, середні відхилення макронутрієнтів – 3,5-4,2%. Досягнення цілей: 80,4% учасників успішно досягли поставлених завдань (схуднення -3,2 кг, підтримка -0,3 кг, набір маси +2,4 кг). Оцінка зручності використання за System Usability Scale склала 82,4 бала, що відповідає рівню хорошої юзабіліті.

Практичне значення роботи полягає у створенні функціональної системи для самостійного планування харчування з науково обґрунтованим підходом до формування раціону. Перспективи розвитку включають інтеграцію з носимими пристроями, впровадження машинного навчання для прогнозування.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Бондаренко О. О. Інформаційні системи підтримки прийняття рішень у харчуванні: методи персоналізації раціону. Київ: НАУ, 2020. 156 с.
2. Гаврилюк Л. В., Паламарчук І. М. Мобільні застосунки для контролю харчової поведінки: огляд функціональних можливостей. Вісник Київського національного університету імені Тараса Шевченка. Серія: Прикладна статистика. 2022. № 2. С. 45–53.
3. Глушко І. Б. Сучасні бази даних продуктів харчування для медичних інформаційних систем. Системні дослідження та інформаційні технології. 2021. № 3. С. 89–98.
4. Гончарук С. М. Архітектура веб орієнтованих систем на платформі Node.js. Комп'ютерні науки та інформаційні технології. 2020. № 4. С. 17–26.
5. Григоренко Д. С., Жукова А. П. Застосування React Native для кроссплатформної розробки мобільних застосунків. Наукові записки НаУКМА. Комп'ютерні науки. 2021. Т. 4. С. 55–62.
6. Жданова Н. М. Персоналізоване харчування як інструмент профілактики неінфекційних захворювань. Український журнал харчової науки. 2019. Т. 7, № 1. С. 23–31.
7. Коваленко А. П. Алгоритми оптимізації раціону харчування з урахуванням індивідуальних параметрів користувача. Кібернетика і системний аналіз. 2022. Т. 58, № 2. С. 74–83.
8. Крамаренко Ю. О. Оцінювання юзабіліті мобільних застосунків для здоров'я за методикою SUS. Вимірювальна та обчислювальна техніка в технологічних процесах. 2023. № 1. С. 60–67.
9. Мельник В. В. Реляційні СУБД у високонавантажених веб сервісах: переваги PostgreSQL. Проблеми програмування. 2020. № 3. С. 92–101.
10. Олійник К. І. Методи розрахунку базального метаболізму в системах планування харчування. Математичні машини і системи. 2021. № 4. С. 41–50.

11. Паламар Ю. С., Степаненко Н. П. Користувацький досвід у мобільних застосунках для здорового способу життя. Інформаційні технології та комп'ютерна інженерія. 2022. № 1. С. 33–40.
12. Пархоменко Р. Б. Хмарні сервіси для зберігання та обробки даних у мобільних медичних застосунках. Радіoeлектроніка, інформатика, управління. 2020. № 2. С. 58–66.
13. Петренко О. І. Принципи побудови RESTful API для інтеграції мобільних застосунків з базами даних харчових продуктів. Вісник НТУУ «КПІ». Інформатика та обчислювальна техніка. 2021. № 75. С. 27–35.
14. Романенко І. Ю. Персоналізовані рекомендаційні системи в галузі харчування: огляд підходів. Штучний інтелект. 2023. № 1. С. 12–24.
15. Сидоренко Л. Л., Ткаченко Н. В. Оцінка якості даних у базах харчової цінності продуктів. Статистика України. 2019. № 4. С. 77–85.
16. Brooke J. SUS: A quick and dirty usability scale. In: Jordan P. et al. (eds.). Usability Evaluation in Industry. London: Taylor & Francis, 1996. P. 189–194.
17. Fialkowski M. K., Harnack L. et al. Assessment of the accuracy of nutrient calculations of five popular nutrition tracking applications. Journal of the Academy of Nutrition and Dietetics. 2018. Vol. 118, No. 5. P. 924–930.
18. MyFitnessPal. URL: <https://www.myfitnesspal.com> (дата звернення: 10.12.2025)
19. Lifesum. URL: <https://www.lifesum.com> (дата звернення: 10.12.2025)
20. Eat This Much. URL: <https://www.eatthismuch.com> (дата звернення: 10.12.2025).
21. FoodData Central. United States Department of Agriculture. URL: <https://fdc.nal.usda.gov> (дата звернення: 10.12.2025).
22. Ahuja J. K. C., Pehrsson P. R., Haytowitz D. B. USDA's FoodData Central: what is it and why is it needed today? Current Developments in Nutrition. 2022. Vol. 6, Issue 3. P. 1–7.

23. RisingMax Inc. AI Meal Planner App Development: Cost & Features 2025. URL: <https://risingmax.com/blog/ai-meal-planner-app-development> (дата звернення: 10.12.2025).

24. WebGen Technologies. How to Develop a Meal Planner App: Complete Step by Step Guide. URL: <https://www.webgentechnologies.us/how-to-develop-a-meal-planner-app-complete-step-by-step-guide> (дата звернення: 10.12.2025).

25. DevTechnosys. Develop a Meal Planner App like Plan Meals in 2025. URL: <https://devtechnosys.ae/blog/develop-a-meal-planner-app-like-plan-meals> (дата звернення: 10.12.2025).

26. Binary Studio. Building an Organic Meal Subscription and Delivery App for a UK Startup. URL: <https://binary-studio.com/case-studies/sustainable-food-delivery-app> (дата звернення: 10.12.2025).

27. Strapi. Build a Food Delivery App with React Native and Strapi v5. URL: <https://strapi.io/blog/building-a-complete-food-mobile-app-with-react-native-and-strapi> (дата звернення: 10.12.2025).

28. TurboStarter. MealMind – AI driven meal planner SaaS idea. URL: <https://www.turbostarter.dev/ideas/b2c/mealmind> (дата звернення: 10.12.2025).

29. Hirsch Health Sciences Library. FoodData Central (Formerly USDA National Nutrient Database for Standard Reference). URL: <https://hirshlibrary.tufts.edu/what-we-have/databases/fooddata-central> (дата звернення: 10.12.2025).

30. PubMed. Assessment of the accuracy of nutrient calculations of five popular nutrition tracking applications. URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC10261454> (дата звернення: 10.12.2025)

ДОДАТОК А

СЛАЙДИ ПРЕЗЕНТАЦІЇ

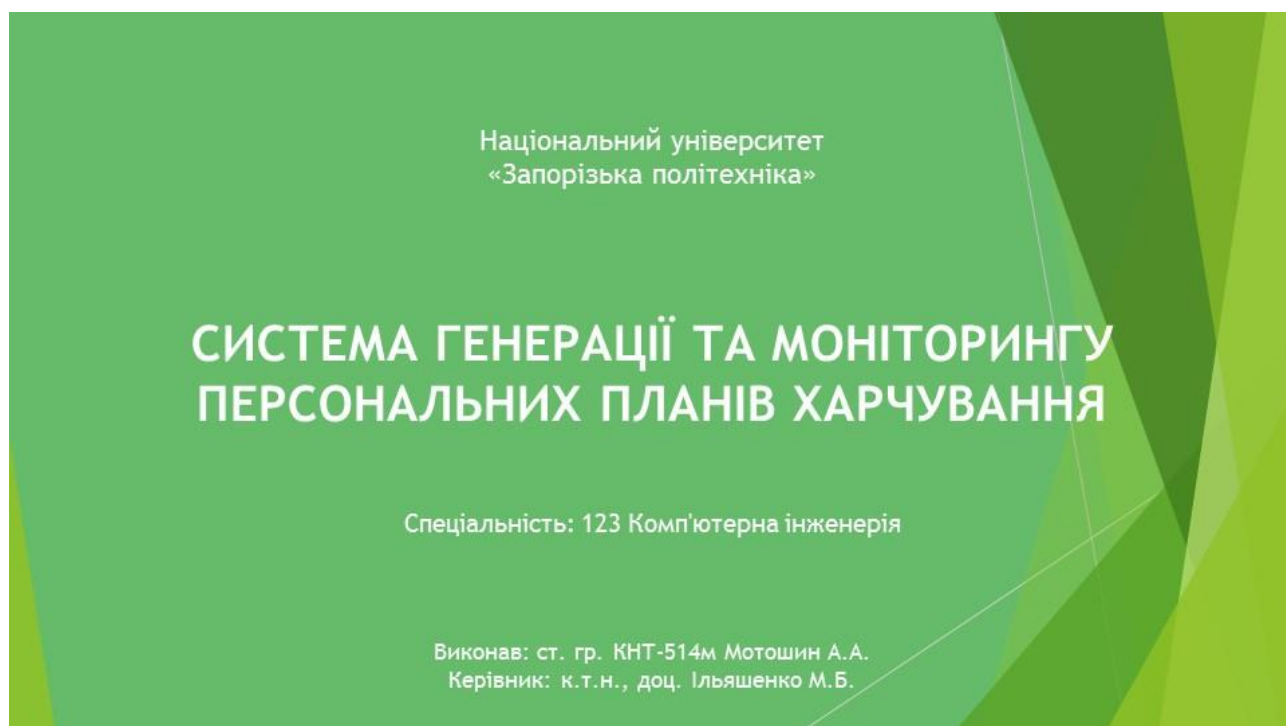


Рисунок А.1 – Слайд 1

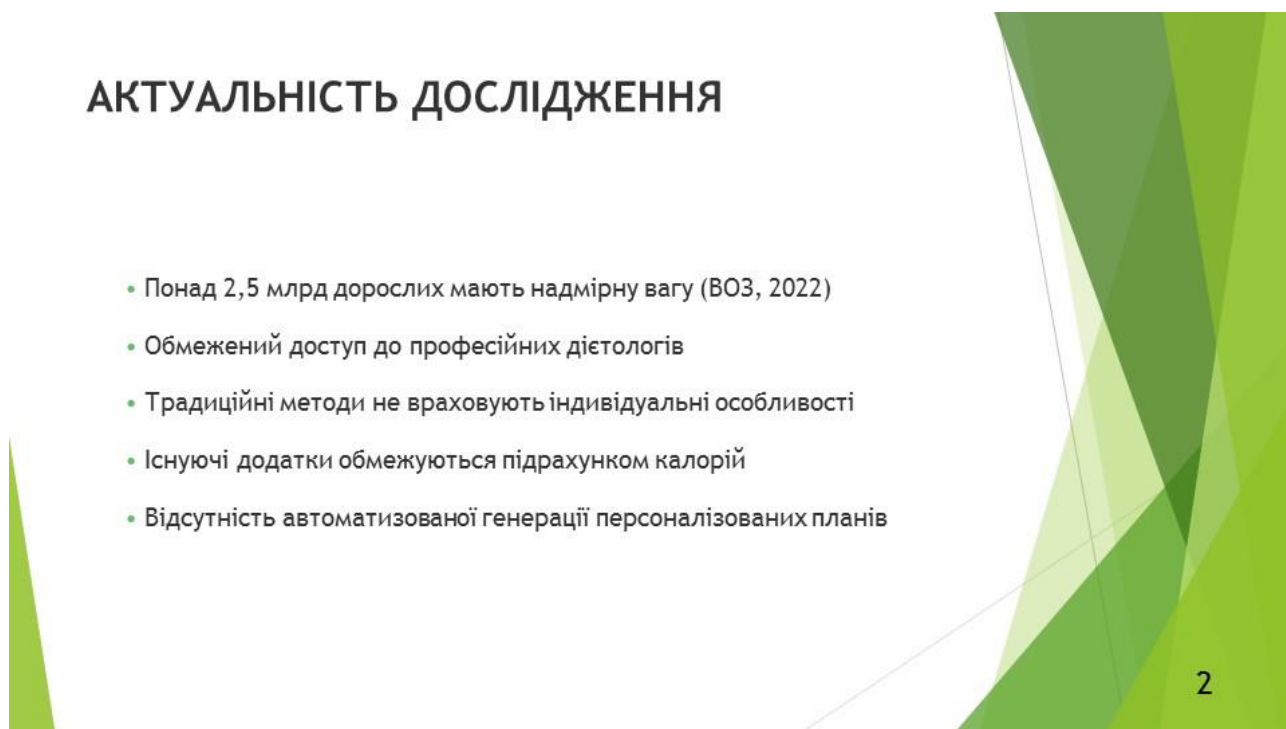


Рисунок А.2 – Слайд 2

МЕТА ТА ЗАВДАННЯ

Мета роботи:

Розробка мобільної системи для автоматизованої генерації персональних планів харчування з урахуванням індивідуальних параметрів користувача та забезпечення моніторингу досягнення цілей

Основні завдання:

1. Аналіз існуючих рішень та методів планування харчування
2. Розробка математичної моделі та алгоритму генерації планів
3. Проектування архітектури мобільної системи
4. Реалізація модулів реєстрації, генерації планів, моніторингу
5. Експериментальне дослідження ефективності системи

3

Рисунок А.3 – Слайд 3

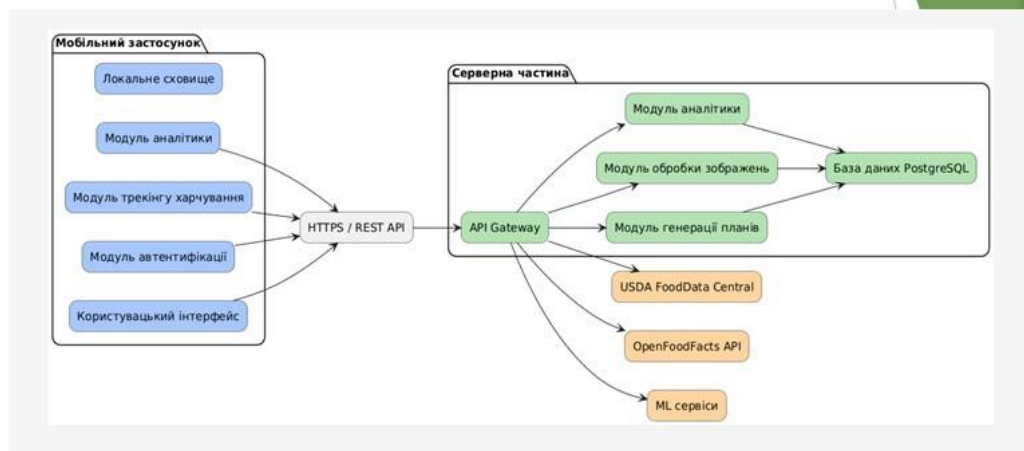
АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ

Додаток	Генерація планів	Персоналізація	Трекінг	База продуктів	Основні недоліки
MyFitnessPal	Ні	Низька	Так	14+ млн продуктів	Відсутність автоматичної генерації планів
Lifesum	Так	Середня	Так	2000+ рецептів	Шаблонні плани без глибокої персоналізації
Eat This Much	Так	Висока	Так	5000+ рецептів	Обмежена база рецептів, відсутність адаптивного навчання

4

Рисунок А.4 – Слайд 4

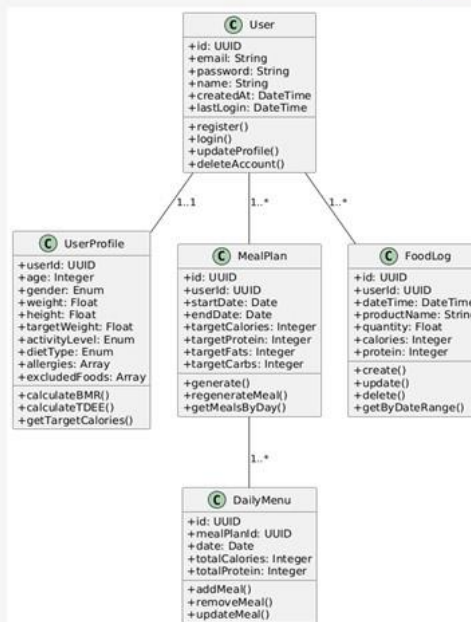
КОНЦЕПТУАЛЬНА МОДЕЛЬ СИСТЕМИ



5

Рисунок А.5 – Слайд 5

Діаграма класів основних сутностей



6

Рисунок А.6 – Слайд 6

АЛГОРИТМ ГЕНЕРАЦІЇ ПЛАНІВ

Комбінований підхід:

1. Розрахунок цільових показників
 - Базальний метаболізм (формула Міффліна-Сан Жеора)
 - Цільова калорійність з коефіцієнтом активності
 - Розподіл макронутрієнтів (білки, жири, вуглеводи)
2. Фільтрація продуктів
 - Виключення алергенів та небажаних продуктів
 - Відповідність типу дієти
3. Евристична генерація + локальна оптимізація
 - Жадібний вибір рецептів
 - Балансування макронутрієнтів (точність $\pm 5\%$)

7

Рисунок А.7 – Слайд 7

ТЕХНОЛОГІЧНИЙ СТЕК

Компонент	Обрана технологія	Альтернативи	Обґрунтування вибору
Мобільний фреймворк	React Native 0.74	Flutter, Ionic, Xamarin	Єдина кодова база для iOS та Android, велика екосистема бібліотек, можливість використання JavaScript
Управління станом	Redux 5.0	MobX, Context API, Zustand	Передбачувані зміни стану, відмінні інструменти налагодження, широка підтримка спільноти
Серверний фреймворк	Node.js + Express	Python + Django, Java + Spring	Висока продуктивність для I/O операцій, єдина мова з клієнтом, швидка розробка
База даних	PostgreSQL 15	MySQL, MongoDB, SQLite	ACID транзакції, розширені можливості SQL, підтримка JSON, повнотекстовий пошук
ORM	Sequelize 6.35	TypeORM, Prisma, Knex	Зріла бібліотека з великою спільнотою, підтримка міграцій, асоціацій, транзакцій
Автентифікація	JWT + bcrypt	OAuth2, Session-based	Stateless автентифікація, підходить для мобільних додатків, простота реалізації

8

Рисунок А.8 – Слайд 8

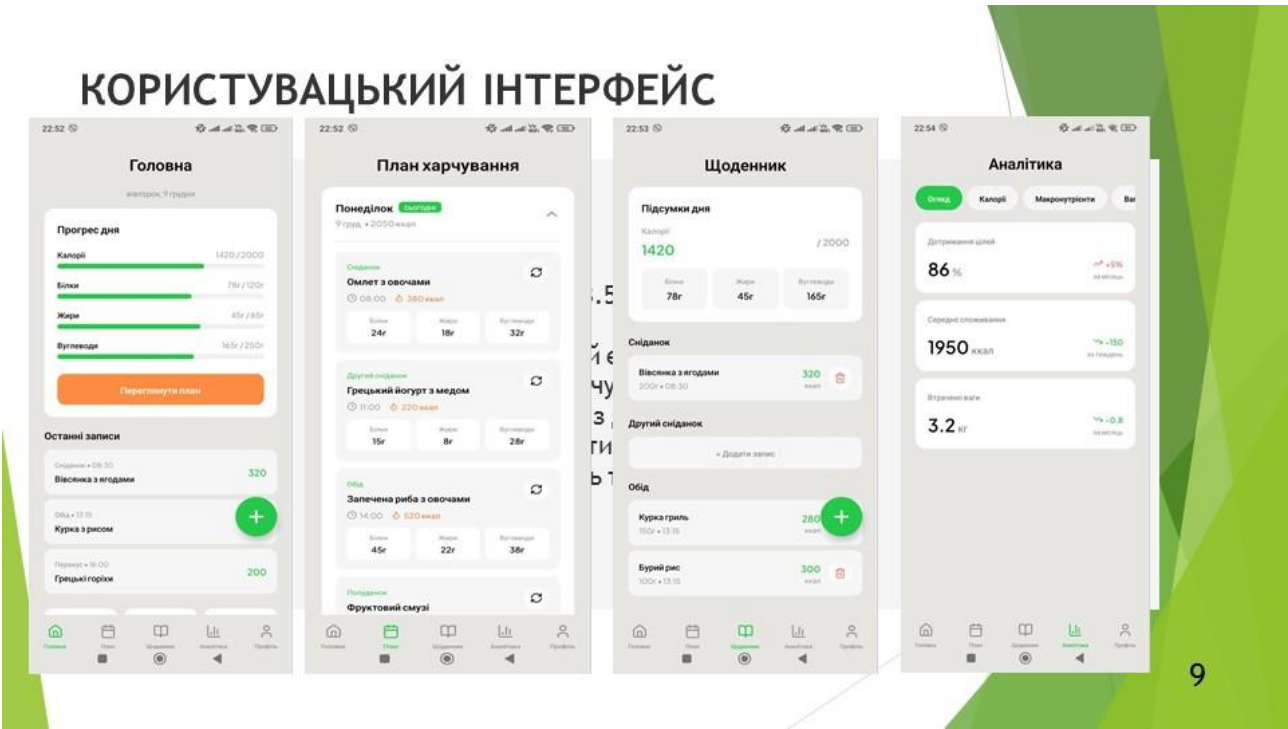


Рисунок А.9 – Слайд 9

ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ

Умови експерименту:

- Учасників: 25 осіб (22-45 років)
- Тривалість: 4 тижні (жовтень-листопад 2024)
- Групи: схуднення (10), підтримка ваги (8), набір маси (7)

Показники оцінки:

Показник	Метод вимірювання	Очікуване значення
Точність балансу калорій	Середнє відхилення фактичних від цільових	< 10%
Точність макронутрієнтів	Відхилення білків, жирів, вуглеводів	< 5%
Дотримання плану	Відсоток днів з повним трекінгом	> 80%
Досягнення цілей	Зміна ваги відповідно до цілей	> 70%
Зручність використання	Оцінка за шкалою SUS	> 75

Рисунок А.10 – Слайд 10

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Точність генерації планів					Досягнення цілей			
Показник	Калорії	Білки	Жири	Вуглеводи	Показник	Схуднення (n=10)	Підтримка (n=8)	Набір (n=7)
Середнє відхилення, %	4.3	3.8	4.2	3.5	Середня зміна ваги, кг	-3.2 ± 1.1	-0.3 ± 0.4	+2.4 ± 0.8
Стандартне відхилення, %	2.1	1.9	2.3	1.7	Цільова зміна, кг	-3.0	0.0	+2.5
Максимальне відхилення, %	8.7	7.2	9.1	6.8	Відхилення від цілі, %	6.7	-	-4.0
Планів у межах порогу, %	97	94	92	96	Досягли цілі, %	80	75	86
					p-value (парний t-тест)	< 0.001	0.142	< 0.001

Рисунок А.11 – Слайд 11

ВИСНОВКИ

1. Розроблено мобільну систему генерації персональних планів харчування на базі React Native - Node.js - PostgreSQL
2. Реалізовано комбінований алгоритм з точністю балансування макронутрієнтів 3.5-4.2% (середнє відхилення)
3. Інтегровано бази продуктів USDA FoodData Central та OpenFoodFacts (понад 350 000 позицій)
4. Експериментально підтверджено ефективність:
 - 80.4% користувачів досягли поставлених цілей
 - Оцінка зручності SUS: 82.4 бали
5. Перспективи: інтеграція з носимими пристроями, машинне навчання для прогнозування

Рисунок А.12 – Слайд 12