

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

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «бакалавр»
за спеціальністю 121 «Інженерія програмного забезпечення»
(освітня програма «Веб-технології»)

на тему:

**«Мобільний застосунок для оцінювання корисності та якості
продуктів харчування»**

Виконав студент групи ВТ-21-1
МІСІК Іван Віталійович

Керівник роботи:
ФАНТ Микола Олександрович

Рецензент:
ЧИЖМОТРЯ Олена Геннадіївна

Житомир – 2025

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

«ЗАТВЕРДЖУЮ»

Зав. кафедри інженерії
програмного забезпечення

Тетяна Вакалюк

«14» лютого 2025 р.

ЗАВДАННЯ
на кваліфікаційну роботу

Здобувач вищої освіти: **МІСІК Іван Віталійович**

Керівник роботи: **ФАНТ Микола Олександрович**

Тема роботи: **«Мобільний застосунок для оцінювання корисності та якості продуктів харчування»**,

затверджена Наказом закладу вищої освіти від **«14» лютого 2025 р., №61с**

Вихідні дані для роботи: **використання фреймворку React Native, бібліотеки React JS для створення мобільного застосунку для оцінювання корисності та якості продуктів харчування.**

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

Розділ	Консультант	Завдання видав	Завдання прийняв
1	Фант М.О.	14.02.2025р.	14.02.2025р.
2	Фант М.О.	14.02.2025р.	14.02.2025р.
3	Фант М.О.	14.02.2025р.	14.02.2025р.

РЕФЕРАТ

Випускна робота бакалавра представляє собою «Мобільний застосунок для оцінювання корисності та якості продуктів харчування».

Пояснювальна записка до кваліфікованої роботи складається з переліку умовних скорочень, вступу, трьох розділів, висновків, списку використаної літератури та додатку.

Текстова частина викладена на 60 сторінках друкованого тексту. Пояснювальна записка має 3 сторінки додатків. Список використаних джерел містить 37 найменувань і займає 4 сторінки. В роботі наведено 23 рисунки та 8 таблиць. Загальний обсяг роботи - 63 сторінок.

В роботі визначені головні завдання для розробки мобільного застосунку, проаналізовано аналоги продукту та методи розв'язання поставленої задачі, обґрунтовано вибір архітектури системи. Представлено алгоритм роботи програмного комплексу, структуру та взаємодію його компонентів та описана програмна реалізація системи.

КЛЮЧОВІ СЛОВА: ЗАСТОСУНОК, МОБІЛЬНИЙ, REACT NATIVE, API, TYPESCRIPT.

					ПЗ.КР.Б-121-25-ПЗ			
Зм.	Арк.	№ докум.	Підпис	Дата	Мобільний застосунок для оцінювання корисності та якості продуктів харчування Пояснювальна записка	Літ.	Арк.	Аркушів
Розроб.		І. МІСІК						
Керівник		М. ФАНТ					3	
н. контр.						Житомирська політехніка, група ВТ-21-1		
Зав. каф.								

ABSTRACT

The bachelor's thesis is “Mobile application for assessing the healthiness and quality of food”.

The explanatory note to the qualified work consists of a list of abbreviations, introduction, three chapters, conclusions, a list of references and an appendix.

The text part is set out on 60 pages of printed text. The explanatory note has 3 pages of appendices. The list of references includes 37 titles and occupies 4 pages. The paper contains 23 figures and 8 tables. The total volume of the work is 63 pages.

The paper identifies the main tasks for the development of a mobile application, analyzes product analogs and methods for solving the task, and justifies the choice of system architecture. The algorithm of the software system, the structure and interaction of its components are presented, and the software implementation of the system is described.

KEYWORDS: APPLICATION, MOBILE, REACT NATIVE, API, TYPESCRIPT.

		<i>Місік І.В.</i>			ПЗ.КР.Б-121-25-ПЗ	
		<i>Фант М.О.</i>				
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		4

РЕФЕРАТ	3
ABSTRACT	4
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	7
ВСТУП.....	8
РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ РОЗРОБКИ МОБІЛЬНОГО ЗАСТОСУНОКУ ДЛЯ ОЦІНЮВАННЯ КОРИСНОСТІ ТА ЯКОСТІ ПРОДУКТІВ ХАРЧУВАННЯ.....	10
1.1 Постановка задачі	10
1.2 Аналіз аналогів програмного продукту	11
1.3 Вибір архітектури застосунку	14
1.4 Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення.....	16
Висновок до розділу 1	17
РОЗДІЛ 2. ПРОЕКТУВАННЯ МОБІЛЬНОГО ЗАСТОСУНКУ.....	18
2.1. Визначення варіантів використання і структури застосунку	18
2.2 Проектування бази даних системи.....	23
2.3. Проектування та реалізація алгоритмів роботи системи	27
2.4 Реалізація застосунку веборієнтованої системи оцінки якості продуктів харчування.....	36
Висновки до розділу 2	41
РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З ЗАСТОСУНКОМ.....	42
3.1. Порядок встановлення та налаштування параметрів системи.	42
3.2 Структура інтерфейсу мобільного застосунку	43
3.3 Тестування роботи програмного комплексу	52
Висновки до розділу 3	54
ВИСНОВКИ	55

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	5
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ..... 56

ДОДАТКИ 60

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

TS – TypeScript

API – прикладний програмний інтерфейс

БД – база даних

ПЗ – програмне забезпечення

UI – інтерфейс користувача

ШІ – штучний інтелект

		Місік І.В.			ПЗ.КР.Б-121-25-ПЗ	
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		7

ВСТУП

Актуальність теми. У сучасному світі, де питання якості харчових продуктів набуває особливої важливості через зростання рівня захворювань, пов'язаних з неправильним харчуванням, це питання наразі є ключовим в сфері харчування. За даними Всесвітньої організації охорони здоров'я, неінфекційні захворювання, такі як діабет, серцево-судинні захворювання та ожиріння, є причиною 71% усіх смертей у світі, а основним фактором ризику розвитку цих захворювань є неправильне харчування, у вигляді споживання продуктів з високим вмістом цукру, солі та насичених жирів. Питання полягає в тому, що споживачам часто немає діла до знання інформації про поживну цінність продуктів або вони не можуть правильно її інтерпретувати. Етикетки на упаковках містять інформацію, яка важко сприймається споживачами. Виробникам продуктів вигідно приховувати їх склад за ширмою складних слів так, щоб пересічний споживач не приділяв уваги словам на етикетці. Розвиток технологій, а насамперед мобільних, відкриває нові можливості для вирішення цієї проблеми. Мобільні застосунки можуть забезпечити швидкий доступ до інформації про продукти та їх аналіз у зручному для користувача форматі. Використання сучасних алгоритмів оцінки якості, таких як Nutri-Score, дозволяє перетворити складну поживну інформацію в простий та зрозумілий рейтинг.

Мета роботи – розробка застосунок для оцінювання корисності та якості продуктів харчування, що забезпечить свідомий вибір продукту харчування на основі даних про нього.

Для досягнення цілі необхідно вирішити наступні **завдання** :

1. Проаналізувати всі існуючі методи оцінки та аналогічні застосунки для оцінки продуктів, зробити аналіз алгоритму Nutri-Score та його застосування.
2. Обґрунтувати вибір технологічного стеку для розробки мобільного застосунку, включаючи фреймворк React Native, TypeScript, Firebase.
3. Розробити архітектуру для застосунку, включаючи клієнт-серверну взаємодію, структуру бази даних та алгоритми обробки даних.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	8
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

4. Розробити застосунок з функціоналом пошуку продуктів, сканування штрихкодів, відображення інформації про продукт та системи відгуків користувачів.

5. Провести тестування розробленого застосунку, оцінити його відповідно до всіх тестових сценаріїв використання.

Об’єкт дослідження – процеси оцінки та аналізу якості харчових продуктів з використанням технологій розробки мобільних застосунків.

Предмет дослідження – методи та засоби розробки мобільних застосунку для автоматизованої оцінки якості харчових продуктів на основі алгоритму Nutri-Score.

Методи дослідження – системний аналіз, порівняльний аналіз, об’єктно-орієнтоване моделювання, алгоритмічний підхід, статистичний аналіз.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		9

РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ РОЗРОБКИ МОБІЛЬНОГО ЗАСТОСУНОКУ ДЛЯ ОЦІНЮВАННЯ КОРИСНОСТІ ТА ЯКОСТІ ПРОДУКТІВ ХАРЧУВАННЯ

1.1 Постановка задачі

У випускній кваліфікаційній роботі необхідно розробити мобільний застосунок для оцінювання корисності та якості продуктів харчування, що складе допомогу звичайним споживачам при виборі для себе продукту харчування, який би відповідав їхнім харчовим притаманностям.

Застосунок має складатись із двох частин: клієнтська частина для кінцевих користувачів та адміністративна панель для управління контентом.

Розробка функціонального застосунку починається з проектування архітектури, що включає аналіз вимог до функціональності, вибір технологічного стеку та проектування структури бази даних. Наступним кроком є створення користувацького інтерфейсу через розроблення прототипу UI/UX дизайну, реалізацію навігації між екранами, розробку системних компонентів, інтеграцію сканера штрихкодів та створення системи відгуків.

Серверна логіка реалізується через Firestore для зберігання даних, впровадження системи аутентифікації користувачів та розробку API для роботи з продуктами. Управління станом застосунку забезпечується через налаштування Redux Toolkit для централізованого контролю, створення slice-ів для різних частин програми та оптимізацію продуктивності з кешуванням даних.

Завершальним етапом є тестування та розгортання, що передбачає модульне тестування компонентів та оптимізацію загальної продуктивності застосунку для забезпечення стабільної роботи в реальних умовах експлуатації.

Результатом виконання має стати повністю функціональний мобільний застосунок, що дозволить користувачам легко визначити у виборі продукту харчування основуючись знаннями про всю його харчову цінність. Це повинно збільшити відсоток людей, які слідкують за своїм здоров'ям.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	10
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

1.2 Аналіз аналогів програмного продукту

Розглянемо наявні аналоги, які найбільше подібні до теми кваліфікаційної роботи: Yuka, Open Food Facts та Fooducate. Наведемо їх можливості, переваги та недоліки, а також порівняємо з нашим майбутнім застосунком.

Yuka. Yuka – це мобільний застосунок для аналізу харчових продуктів і косметики. Він сканує штрихкоди товарів, аналізує їх склад та надає оцінку щодо корисності продукту для здоров'я.

До переваг застосунку можна віднести його дуже простий та зручний інтерфейс, велику базу продуктів із детальним описом, пояснення визначеної оцінки за певними критеріями та пропозиції альтернатив продуктів. В цілому взаємодія із застосунком дуже приємна, є відчуття, що ти користуєшся дійсно якісним продуктом. Крім того можна також віднести до плюсів, що застосунок має у собі деталі про якісь не лише харчових продуктів, а й косметичні засоби

Явні недоліки це неможливість завантажити застосунок на території України, з цього впливає той факт, що база продуктів не підтримує український регіон продуктів. Власна система оцінювання не є плюсом, враховуючи, що є європейський алгоритм оцінки, що має наукову базу під собою.

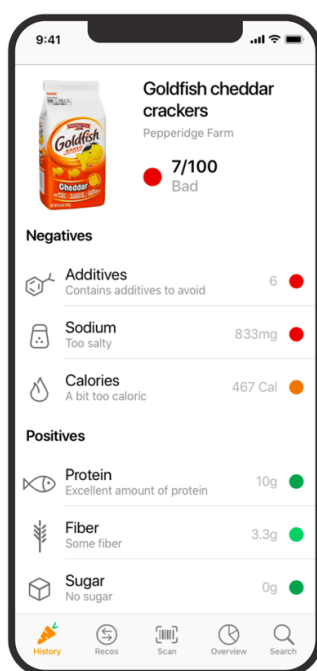


Рис. 1.1 – Сторінка продукту у застосунку Yuka

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	11
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Open Food Facts. Open Food Facts – мобільний застосунок, який дозволяє користувачам сканувати штрихкоди продуктів для отримання інформації про їх склад.



Рис. 1.2 – Сторінка продукту у застосунку OpenFoodFacts

Із переваг цього застосунку можна відзначити велику базу продуктів по всьому світу, бо він повністю підтримується спільнотою. Також немалим плюсом є система оцінювання, яка базується на Nutri-Score.

Так як застосунок підтримується спільнотою то відповідно там відсутня достатня модерація. Багато продуктів мають лише фотографії та навіть не мають складу нутрієнтів.

Fooducate. Fooducate – ще один застосунок для аналізу харчових продуктів, який надає інформацію про інгредієнти, калорійність, вплив на здоров'я.

Переваги цього застосунку, як і в Yuka у сучасному дизайні націленому на простоту та зручність. Система має контент про харчування, різні рецепти страв, тощо. Недоліки такі самі, як і у Yuka: обмежений доступ для всіх крім США, власна система оцінки, платний функціонал

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	12
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

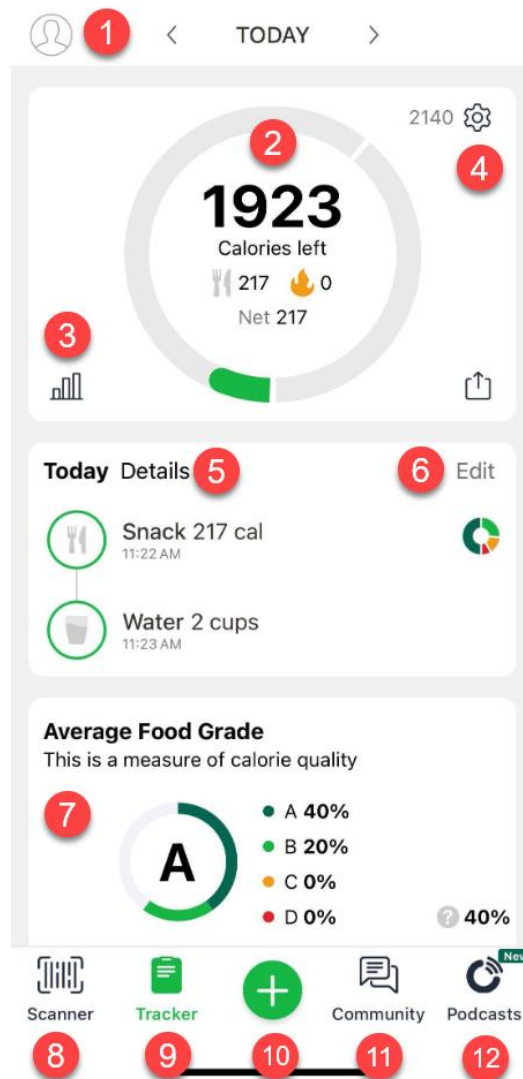


Рис. 1.3 – Сторінка у застосунку Fooducate

Переваги цього застосунку, як і в Yuka у сучасному дизайні націленому на простоту та зручність. Система має контент про харчування, різні рецепти страв, тощо. Є система градації продуктів та фокус на самій якості харчування.

Недоліки такі самі, як і у Yuka: обмежений доступ для всіх крім США, власна система оцінки, платний функціонал

Таблиця 1.1

Порівняльна таблиця наявних рішень на ринку

Критерій	Yuka	OpenFoodFacts	Fooducate	Розроблений застосунок
Простота інтерфейсу	+++	+	+++	+++
База продуктів	1.5 млн продуктів	2.8 млн продуктів	400 тис. продуктів	Локальна

Продовження таблиці 1.1

Система оцінки	Власна система (0- 100)	Nutri-Score	Літерна система (A-D)	Nutri-Score
Аналіз складу	+++	++	++	+++
Підтримка локальних продуктів	+	++	-	+++
Оцінка впливу на здоров'я	++	++	+	++
Переваги	Простий інтерфейс, швидкість	Відкрита база даних, багато продуктів	Освітній контент	Стандарт оцінки продукту
Недоліки	Закрита методологія, обмежена база	Складний інтерфейс, повільна робота	Орієнтація на США, застарілий дизайн	Наразі відсутність великої бази продуктів
Можливість залишати відгуки	-	-	-	+

Аналіз представлених аналогів до планованого застосунку показав, що вони мають ряд як плюсів, так і мінусів. Основними недоліками є погане покриття або повна відсутність представленості на українському харчовому ринку, що значно обмежує їх практичну користь для вітчизняних споживачів.

Відсутність стандартизованого підходу до оцінювання продуктів створює плутанину та ускладнює порівняння товарів між різними платформами.

Враховуючи всі перераховані недоліки у вже готових рішеннях, було прийнято рішення розробити власний мобільний застосунок з наступними критеріями відповідно до актуальних проблем: зробити доступний для користувача інтерфейс, використати європейську систему оцінювання якості продуктів та націлитись на національний ринок з можливістю розширення.

1.3 Вибір архітектури застосунку

В процесі розробки мобільного застосунку для оцінки якості харчових продуктів розглянуто низка різних варіантів архітектурних рішень з метою вибору оптимальної моделі, що забезпечує ефективну взаємодію між компонентами системи. З урахуванням специфіки мобільної розробки та вимог до проекту, було прийнято рішення використовувати комбінацію архітектурних патернів.

		Місік І.В.			ПІЗ.КР.Б-121-25-ПЗ	14
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Клієнтська архітектура. Для клієнтської частини було обрано архітектуру з використанням паттерну ModelView-Controller (MVC) в поєднанні з односторонньою архітектурою потоку даних (Unidirectional Data Flow), реалізованою через Redux.

Даний підхід до структурування застосунків спрямований на чітке розмежування програмної логіки та користувацького інтерфейсу, що створює можливості для багаторазового застосування елементів системи.

MVC представляє собою фундаментальний принцип організації веб-застосунків, що базується на трьох основних складових. Використання цього підходу передбачає поділ програмного продукту на такі елементи:

1. Модель (Model): Цей елемент методології керує основною логікою застосунку. У ньому є всі необхідні дані для їх опрацювання та зберігання.
2. Вигляд (View): Елемент відповідаючий за візуальне представлення інтерфейсу застосунку. Він виводить всю інформацію візуально інформацію, що дає можливості для взаємодії з користувачем.
3. Керуючий елемент (Controller): Це щось в ролі з'єднувальної мотузки між моделлю та виглядом. Цей елемент фіксує всі запити користувачів, модифікує існуючу модель та оновлює інтерфейс застосунку.

Інше рішення було ухвалене щодо серверної компоненти на платформі Firebase. Тут використано BaaS підхід Backend-as-a-Service:

Обчислювальний процес виявився також вигідним з декількох причин. По-перше, замість того щоб створювати інфраструктуру з нуля, можна взяти повністю готове та добре функціонуюче рішення від Firebase, а це значно скоротило час розробки. Firebase може автоматично змінювати розмір ресурсів в залежності від піку навантаження, щоб забезпечити стабільність застосунку у випадку зростання числа користувачів. Платформа також має вбудований захисний канал у бекапування, які значно зменшує ризик втрати даних. Такий підхід зменшує витрати на підтримку через те, що будь-які технічні проблеми розв'язує провайдер.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	15
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

1.4 Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення

При створенні мобільного застосунку було вирішено використовувати кросплатформний підхід на основі React Native. React Native дозволяє створювати застосунки одночасно для iOS та Android, використовуючи єдину кодову базу, що значно економить час та ресурси. Фреймворк забезпечує високу продуктивність завдяки використанню нативних компонентів операційних систем. Навколо React Native сформувалася потужна спільнота розробників та багата екосистема готових бібліотек.

Як основну мову програмування було обрано TypeScript замість звичайного JavaScript. Статична типізація TypeScript суттєво зменшує кількість помилок на етапі розробки, виявляючи потенційні проблеми ще до запуску програми.

Для спрощення розробки було вибрано платформу Expo, яка надає готові компоненти для роботи з камерою та сканування штрих-кодів. Expo значно спрощує процес збірки та розгортання застосунків, дозволяє легко тестувати програми на реальних пристроях без складного налаштування середовища.

Навігація між екранами реалізована за допомогою бібліотеки React Navigation, яка підтримує різні типи переходів між сторінками, включаючи стековий тип та вкладки.

Додатково в проєкті використовуються спеціалізовані бібліотеки для сканування штрихкодів продуктів, відображення іконок інтерфейсу та HTTP клієнт для взаємодії з серверними API.

Технічні вимоги до обладнання. Мобільний застосунок призначений для роботи на сучасних смартфонах з базовими характеристиками. Для пристроїв на Android потрібна операційна система версії 11 або новіша, принаймні 2 ГБ оперативної пам'яті, хоча краще мати 4 ГБ для комфортної роботи та достатньої кількості вільного простору в пам'яті пристрою. Обов'язковою є наявність камери з автофокусом для функції сканування штрих-кодів та стабільного підключення до інтернету через Wi-Fi або мобільні дані.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		16

Пристрої iPhone повинні працювати під управлінням iOS версії 11.0 або новіших, підтримуються моделі починаючи з iPhone 6s. Вимоги до пам'яті та дискового простору аналогічні Android версії. Також необхідна камера з автофокусом та підключення до мережі інтернет.

Висновок до розділу 1

У першому розділі випускної кваліфікаційної роботи було проведено аналіз проблематики галузі, що дозволило визначитися із фінальним технологічним стеком для розробки мобільного застосунку.

Технологічний стек сформований із сучасних технологій, які найкращим чином підходять для вирішення піднятої проблематики в галузі харчування. React Native було обрано як основу для кросплатформної розробки, TypeScript забезпечує надійність коду через його сувору типізацію, Redux Toolkit керує станами в системі, а Firebase надає серверну інфраструктуру. Такий вибір гарантує створення застосунку, який буде легкий для підтримки та масштабуванні.

Результати дослідження дають чіткий погляд на план для етапів роботи над проектом. Обрані технології та підходи дозволять створити сучасний застосунок, який не лише відповідає актуальним стандартам розробки, але й може легко адаптуватися до майбутніх змін та розширюватися новим функціоналом.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		17

РОЗДІЛ 2. ПРОЕКТУВАННЯ МОБІЛЬНОГО ЗАСТОСУНКУ

2.1. Визначення варіантів використання і структури застосунку

Розроблена система являє собою мобільний застосунок для оцінки якості продуктів харчування, що поєднує в собі взаємодію користувачів різних ролей: звичайних користувачів та адміністраторів. Для наглядного представлення взаємодії між компонентами системи було створено діаграму варіантів використання яка зображена на рисунку 2.1.

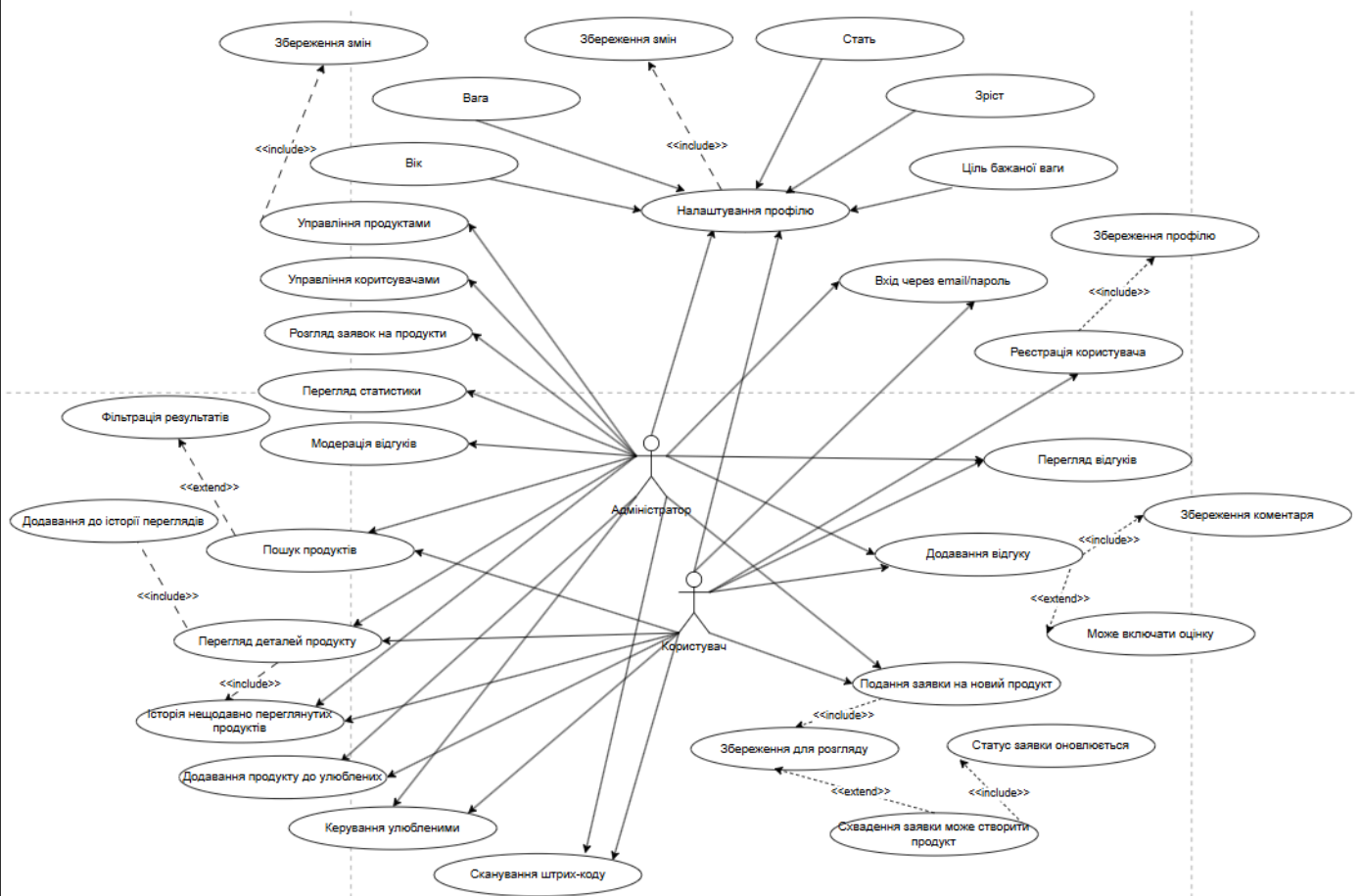


Рисунок 2.1 - Діаграма варіантів використання

Діаграма варіантів використання включає два основні актори:

Користувачі є основними учасниками роботи застосунку. Користувачі взаємодіють із системою для того, щоб отримати інформацію про продукт харчування, залишити відгук, а також для керувати особистими.

Адміністратори є користувачами з розширеним правом доступом і відповідають за адміністрування застосунку, модерацію та підтримку якості даних.

		Місік І.В.			ПЗ.КР.Б-121-25-ПЗ	18
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Функціональні вимоги звичайних користувачів:

У користувачі є два основних способи увійти до системи. Вони можуть зареєструватися за допомогою електронної пошти та пароллю або використати обліковий запис Google, і система автоматично призначить їм відповідну роль програми.

Після входу до системи в застосунку користувачі мають доступ до свого профілю з можливістю ввести та змінити власні особисті дані: інформацію про основні фізичні параметри, такі як вік, зріст і вага.

Пошуку товарів реалізований так, що різноманітні зручні способи для взаємодії. Користувачі можуть ввести назву або бренд продукту, щоб знайти потрібний продукт. Також застосунок дозволяє користувачам сканувати штрихкоди продуктів через камеру свого смартфона.

Система включає функцію залишення відгуку, щоб користувачі могли ставити оцінки за п'ятибальною шкалою та писати текстові коментарі, якщо їм є що сказати про продукт. Це створює спільноту активних користувачів, які діляться власними враженнями та розвиває соціальну взаємодію у застосунку.

Застосунок дозволяє користувачеві зберегти продукт до улюблених, що дозволяє швидко перевірити дані про продукт, які їх чимось зацікавили. Це дозволяє швидко повертатися до раніше досліджених продуктів та відстежувати власні харчові уподобання з часом.

Користувачі можуть зробити свій вклад в розширення бази даних продуктів, подаючи запити на додавання нових продуктів, які ще не представлені в системі. При цьому вони мають можливість завантажувати фотографії товарів, робити ШІ аналіз, що покращує якість та повноту інформації в застосунку.

Центральною функцією системи є автоматичний розрахунок та відображення оцінки Nutri-Score для кожного продукту. Ця оцінка базується на ШІ аналізі складу продукту із подальшим рахуванням оцінки Nutri-Score на основі офіційного математичного алгоритму затвердженого в європейському союзі. Алгоритм враховує різні категорії продуктів, щоб давати максимально об'єктивну оцінку для продукту.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	19
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Вимоги адміністраторів:

Адміністратори системи отримують повний контроль над усією базою харчових продуктів. Вони можуть додавати нові товари, переглядати існуючу інформацію, вносити корективи в описи та характеристики продуктів, а також при необхідності видаляти застарілі або некоректні записи з системи.

Важливою частиною адміністративних функцій є модерація контенту, створеного користувачами. Адміністратори мають повноваження видаляти неприйнятні коментарі та відгуки, а також блокувати користувачів, які порушують правила спільноти.

Система заявок на додавання нових продуктів також знаходиться під контролем адміністраторів. Вони розглядають усі запити від користувачів, аналізують надану інформацію та фотографії, після чого приймають рішення про схвалення або відхилення кожної заявки. Це забезпечує високу якість та достовірність інформації в базі даних.

Адміністратори мають доступ до комплексної аналітики роботи системи. Вони можуть переглядати детальну статистику, яка включає загальну кількість зареєстрованих користувачів та обсяг бази продуктів. Ця інформація допомагає приймати обґрунтовані рішення щодо розвитку платформи.

Бізнес-вимоги:

Мета застосунку дотримуватися європейської системи оцінювання продуктів Nutri-Score. Це дозволяє користувачам точно бути впевненим, що продукт оцінений на основі європейського стандарту.

Застосунок дозволяє розумно підійти до процесу вибору їжі. Користувачі отримують доступ до максимально інформації про склад поживних речовин, потенційні алергени та різноманітні харчові добавки у продуктах. Це дозволяє правильно обрати харчові переваги точно знаючи всі вхідні дані.

Ставка на постійне розширення бази даних продуктів завдяки активній участі користувачів є ключовою. Це дозволяє людям напряму приймати участь у формуванні здорового підходу людей до вибору продуктів харчування. Також

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	20
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

підсвідомо люди відчують гордість та радість від того, що їх невеличке старання якось відзначилось.

Функціональні вимоги:

Аутентифікація реалізована через сервіс Firebase Authentication, що забезпечує безпечний процес реєстрації через електронну пошту або Google-акаунт. Механізм авторизації автоматично призначає ролі звичайним користувачам та адміністраторам з відповідно до рівня доступу.

Центральний магазин інформації про харчові продукти організовано на основі Firebase Firestore. База даних містить детальні характеристики харчових продуктів, починаючи від основних поживних речовин до специфічних алергенів і дієтичних добавок.

Алгоритм Nutri-Score реалізовано так, що система автоматично розраховує харчову цінність продуктів на основі європейських вимог маркування з чіткою градацією від А до Е для зручності користувача.

Технологія сканування реалізована за допомогою Expo Barcode Scanner. Застосунок миттєво розпізнає штрихкоди, починаючи з локального сховища. Цей підхід гарантує актуальність даних та швидкість відповіді.

Записані індивідуальні налаштування забезпечують персоналізований досвід користувача. Хмарна інфраструктура Firebase синхронізує дані пристроїв, такі як улюблені елементи та історія переглядів, на різні пристрої.

Нефункціональні вимоги:

Застосунок повинен забезпечувати швидку роботу: стандартні операції мають мати час виконання не більше трьох секунд, початковий екран завантажується за п'яти секунд, а сканування штрихкодів відбувається за дві секунди. Система має підтримувати одночасну роботу мінімум 1000 користувачів без втрати продуктивності.

Система повинна працювати стабільно з доступністю не менше 99,5% часу та безвідмовно функціонувати протягом мінімум 30 днів. Кількість помилок не має перевищувати 0,1% від усіх користувацьких запитів, а у випадку збоїв система автоматично відновлюється протягом 5 хвилин.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	21
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Застосунок має бути простим у використанні: звичайні користувачі повинні освоїти базові функції за 30 хвилин або менше. Інтерфейс розробляється інтуїтивно зрозумілим відповідно до стандартів Material Design для Android та Human Interface Guidelines для iOS.

Система забезпечує безпечне зберігання персональних даних користувачів та регулярно створює резервні копії з можливістю відновлення протягом 24 годин. Усі зміни критичних даних відстежуються через систему аудиту для забезпечення прозорості та контролю.

Архітектура системи має підтримувати горизонтальне масштабування для обробки зростаючого навантаження та дозволяти додавання нових функцій без впливу на існуючий функціонал.

Застосунок сумісний з iOS версії 11.0 та вище, Android версії 11.0 та вище, а для стабільної роботи потребує мінімум 2 ГБ оперативної пам'яті.

Архітектура системи:

У фундаментального рішення проблеми лежить клієнт-серверна архітектура на базі Firebase хмарної екосистеми. Архітектурне проектування має кілька ключових розподільчих панелей.

Частина застосунку виконана у вигляді React Native програми. Середя забезпечує кроссплатформенну інтеграцію та нативну швидкодію для iOS Android систем.

Уся складова серверної частини реалізується на Firebase Backend Services. Всеохоплююча платформа Google надає комплексне середовище для адміністрування користувачів, зберігання та захищеної автентифікації структурованих даних без необхідності розгортання власної інфраструктури.

Інтеграція API інших систем підвищують функціональність системи. Open Food Facts надає дані для перевірки інформації про продукти харчування, що містяться в світовій базі даних. Gemini в свою чергу надає доступ для функціоналу ШІ, а саме скану зображення продукту, яке надсилає користувач для перевірки, щоб проаналізувати склад та відповідно визначити категорію продукту.

Клієнтська архітектура.

Мобільний застосунок розроблений за принципами чистої архітектури з чітким розділенням на функціональні шари. Шар презентації включає React Native компоненти для інтерфейсу користувача, Redux store для централізованого управління станом застосунку та навігаційну систему для переходів між екранами.

Шар бізнес-логіки містить Redux slices з основною логікою роботи програми, а також валідатори та трансформери для обробки даних. Шар даних відповідає за взаємодію із зовнішніми сервісами через API Firebase, забезпечує локальне кешування інформації для офлайн-режиму та управляє зв'язком із зовнішніми API сервісами.

Така архітектурна структура забезпечує масштабованість, легкість тестування та можливість незалежної розробки різних компонентів системи.

2.2 Проектування бази даних системи

Для зберігання інформації в застосунку було обрано Firebase Firestore, яка являє собою NoSQL базу даних, що працює з документами замість традиційних таблиць.

База даних складається з 6 основних колекцій:

Таблиця 2.1.

Призначення колекцій бази даних

Назва колекції	Призначення
users	Зберігання профілів користувачів, налаштувань та метаданих
products	Повна інформація про продукти включно з поживними речовинами
comments	Відгуки, оцінки та коментарі користувачів про продукти
favorites	Зв'язки між користувачами та їхніми улюбленими продуктами
viewHistory	Історія переглядів продуктів для персоналізації
productSubmissions	Заявки користувачів на додавання нових продуктів
productSubmissions	Заявки користувачів на додавання нових продуктів

Структуру бази даних представлено на рисунку 2.2, де відображено 6 основних колекцій документів з відповідно визначеними зв'язками. Вся система вдало продумана та оптимізована для безпечного зберігання всіх можливих даних, які з'являються в процесі взаємодії користувачів з застосунком та розподіленням на колекції відповідно до типу даних, що необхідно зберігати.

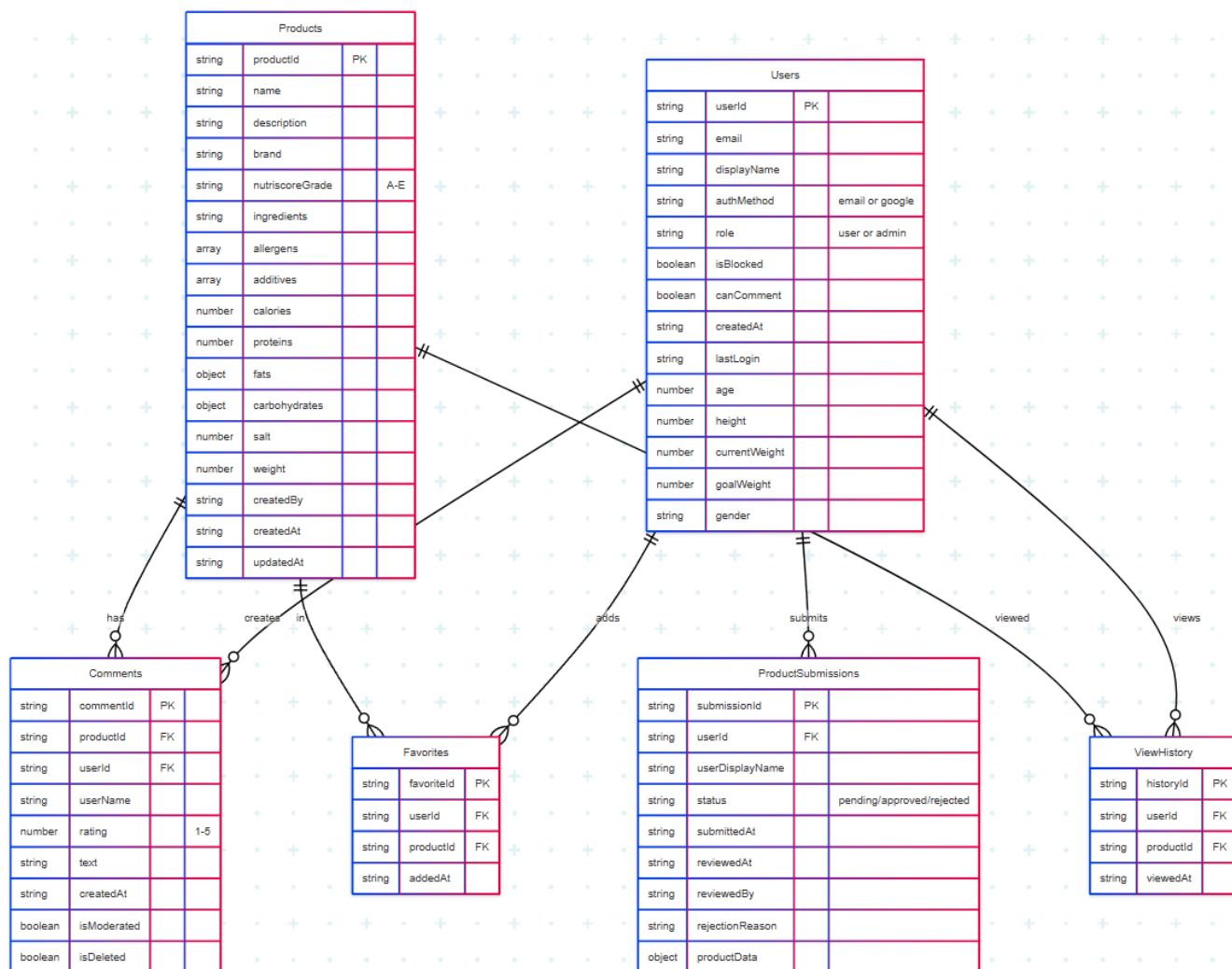


Рис. 2.2 - Діаграма бази даних

Таблиця 2.2

Колекція Users

Поле	Тип	Призначення
userId	string	Унікальний ідентифікатор
email	string	Електронна адреса
displayName	string	Відображуване ім'я
authMethod	"email" "google"	Метод автентифікації
role	"user" "admin";	Роль користувача
isBlocked	boolean	Статус блокування
canComment	boolean	Дозвіл на коментування
createdAt	string	Дата створення користувача

Продовження таблиці 2.2

lastLogin	string	Остання авторизація
age?	number	Вік
height?	number	Зріст
currentWeight?	number	Поточна вага
goalWeight?	number	Цільова вага

Колекція Users відповідає за збереження даних про користувача, ролі та персональні налаштування. Унікальний ідентифікатор `userId` використовується для формування зв'язку з іншими колекціями системи. Поля `authMethod` та `role` відповідають за автентифікацію та права доступу, а `isBlocked` і `canComment` відповідають за стан облікового запису та можливість писати коментарі.

Таблиця 2.3

Колекція Products

Поле	Тип	Призначення
productId	string	Унікальний ідентифікатор
name	string	Назва продукту
description	string	Опис
brand	string	Бренд виробника
nutriscoreGrade:	'A' 'B' 'C' 'D' 'E'	Оцінка
ingredients	string	Список інгредієнтів
allergens	String[]	Алергени
additives	String[]	Харчові добавки
calories	number	Калорійність
proteins	number	Білки
fats: { total; saturated;}	number	Загальні жири Насичені жири
carbohydrates: { total; sugar?}	number	Загальні вуглеводи Цукор
salt?:	number	Сіль
weight?	number	Вага продукту
createdBy	"admin" "api" string	Автор створення
createdAt	string	Дата створення
updatedAt	string	Дата оновлення

Колекція Products є головною частиною системи, зберігаючи в собі всю інформацію про продукт. Поле `nutriscoreGrade` зберігає оцінку якості продукту харчування (A-E), `nutriments` організована як вкладений об'єкт, що містить калорійність, нутрієнти (білки, жири, вуглеводи) та їх підкатегорії (насичені

жири, цукор). Метадані createdBy, createdAt, updatedAt допомагають у модерації продуктів.

Таблиця 2.4

Колекція Comments

Поле	Тип	Призначення
commentId	string	Унікальний ідентифікатор
productId	string	ID продукту
userId	string	ID користувача
userName	string	Ім'я користувача
rating	1 2 3 4 5	Оцінка
text	string	Текст відгуку
createdAt	String	Дата створення
isModerated	boolean	Статус модерації
isDeleted?	boolean	Статус видалення

Колекція Comments виступає соціальною системою застосунку, зберігаючи відгуки та оцінки користувачів щодо продуктів. Поля productId та userId відповідають за зв'язки між відповідними колекціями. Опціональне поле isDeleted дозволяє м'яке видалення відгуків без втрати даних, що важливо для аналітики.

Таблиця 2.5

Колекція Favorites

Поле	Тип	Призначення
favoriteId	string	Унікальний ідентифікатор
userId	string	ID користувача
productId	string	ID продукту
addedAt	string	Дата додавання

Колекція Favorites виступає функціоналом збереження продуктів до улюблених, виступаючи методом швидкого доступу для користувача. Кожен документ містить мінімальний набір полів: унікальний ідентифікатор (favoriteId), посилання на користувача (userId), посилання на продукт (productId) та часову мітку збереження(addedAt).

Таблиця 2.6

Колекція ViewHistory

Поле	Тип	Призначення
historyId	string	Унікальний ідентифікатор
userId	string	ID користувача
productId	string	ID продукту
viewedAt	string	Дата перегляду

Колекція ViewHistory слідує за списком продуктів, які користувач нещодавно переглянув. Аналогічно колекції Favorites, кожен документ ViewHistory встановлює зв'язок між користувачем (userId) та продуктом (productId). Система автоматично створює записи в цій колекції при кожному перегляді детальної інформації про продукт, це дозволяє швидко порівняти продукти, які користувач нещодавно переглядав з метою визначення найкращого вибору особисто для себе.

Таблиця 2.7

Колекція ProductSubmissions

Поле	Тип	Призначення
submissionId	string	Унікальний ідентифікатор
userId	string	ID користувача
userDisplayName	string	Ім'я користувача
status	"pending" "approved" "rejected"	Статус заявки
submittedAt	string	Дата подання
reviewedAt?	string	Дата розгляду
reviewedBy?	string	ID адміна
rejectionReason?	string	Причина відхилення
productData	Product	Дані пропонуваного продукту

Колекція ProductSubmissions відповідає за процес подання користувачем продукту на додавання до бази даних застосунку. Кожна така заявка на подачу містить повну інформацію про продукт (productData) разом з даними про користувача ініціатора (userId, userDisplayName) та статус розгляду (status: pending/approved/rejected). У разі схвалення, дані з productData використовуються для створення нового каталогу Products. Поля reviewedAt, reviewedBy та rejectionReason відповідають за відкритість процесу модерації, а submittedAt фіксує час створення заявки для контролю строків обробки.

2.3. Проектування та реалізація алгоритмів роботи системи

У цьому розділі розглядається діаграма діяльності, які є невід'ємною для процесів розробки програмного забезпечення. Ці діаграми схожі на звичайні блок-схеми і активно використовуються при створенні застосунків на React Native.

UML діаграми діяльності дозволяють описувати конкретні варіанти використання мобільного застосунку, ілюструючи процеси в системі. Їх також можна використовувати для опису бізнес-процесів в аналізі харчування, що допомагає точно окреслити порядок подій і умови, необхідні для виконання подій.

На рисунку 2.3 зображена діаграма діяльності реєстрації та авторизації користувача в застосунку. Користувач обирає для себе спосіб реєстрації: через електронну пошту або авторизацією через Google акаунт.

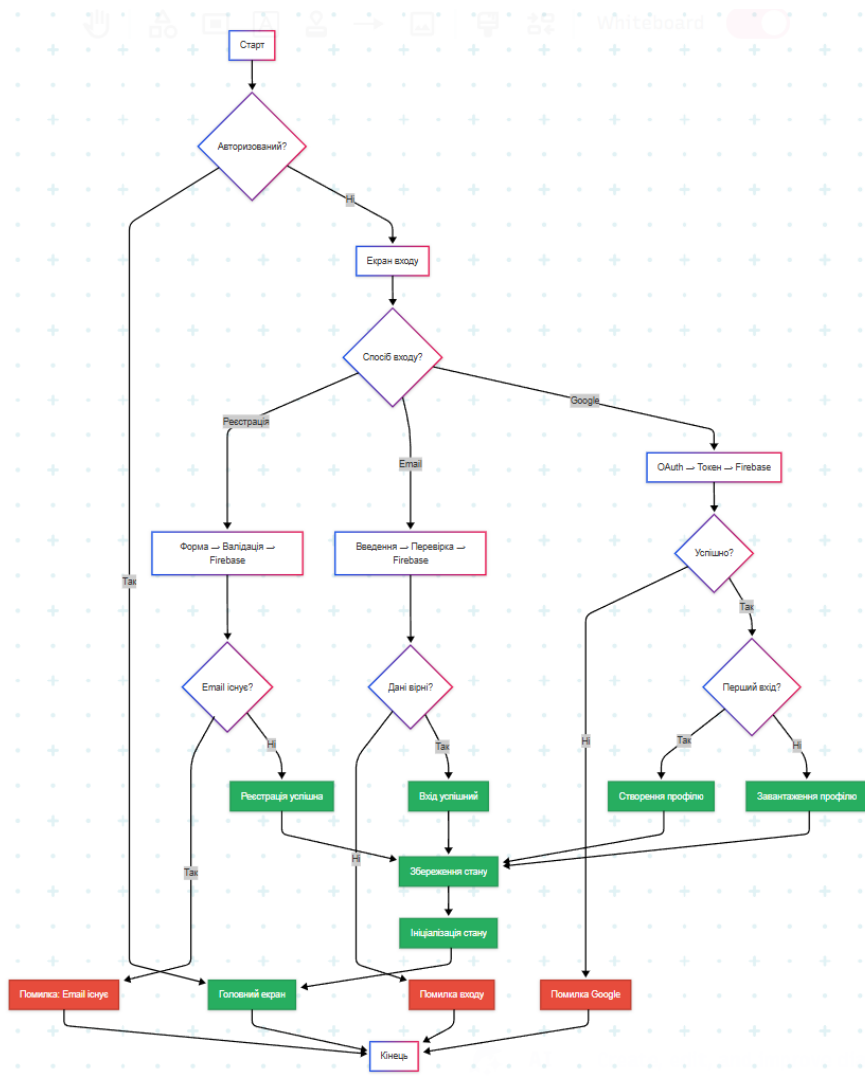


Рис. 2.3 - Діаграма діяльності реєстрації та авторизації користувача

Аналіз алгоритмічної системи оцінки якості продуктів харчування.

Розроблена система оцінки якості продуктів харчування базується на комплексному поєднанні математичних алгоритмів розрахунку Nutri-Score з сучасними технологіями штучного інтелекту. Архітектура системи побудована

на принципах модульності та масштабованості, що забезпечує ефективну обробку великих обсягів даних про харчові продукти. Система інтегрує класичні методи харчової оцінки з інноваційними підходами машинного навчання, створюючи комплексне рішення для автоматизованого аналізу харчових етикеток.

Фундаментальною особливістю розробленої системи є її здатність до автономної роботи з мінімальним втручанням користувача. Алгоритмічна база системи забезпечує точність розрахунків на рівні міжнародних стандартів, водночас адаптуючись до специфіки українського ринку харчових продуктів. Інтеграція штучного інтелекту дозволяє системі автоматично вираховувати оцінку продукту на основі отриманих даних.

Математичний апарат алгоритму Nutri-Score. Основою системи є європейський алгоритм Nutri-Score, який представляє собою багатofакторну модель оцінки харчової цінності продуктів. Алгоритм базується на принципі збалансованого врахування негативних та позитивних харчових компонентів через математичну формулу:

$$\text{NutriScore} = \Sigma \text{NegativePoints} - \Sigma \text{PositivePoints},$$

де $\Sigma \text{NegativePoints}$ представляє суму балів за шкідливі компоненти;

$\Sigma \text{PositivePoints}$ - суму балів за корисні складові продукту.

Негативні бали нараховуються за вміст енергетично щільних та потенційно шкідливих компонентів, таких як енергетична цінність, насичені жири, цукри та натрій. Позитивні бали присвоюються за наявність корисних елементів - клітковини, білків та фруктово-овочевої складової. Така двоскладова структура забезпечує комплексну оцінку харчової цінності продукту, враховуючи як потенційні ризики, так і користь від споживання.

Математична модель алгоритму враховує нелінійні залежності між харчовими компонентами. Кожен компонент оцінюється через систему порогових функцій, що відображає реальний вплив різних концентрацій поживних речовин на організм людини. Така підхід забезпечує більш точну

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	29
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

оцінку порівняно з простими лінійними моделями. Лістинг фрагменту коду відповідального за розрахунок оцінки Nutri-Score можна знайти в додатку А.

Детальний розрахунок негативних балів. Алгоритм розрахунку енергетичних балів використовує систему порогових значень, де кожен поріг відповідає певній кількості балів. Для загальних продуктів застосовуються пороги від 335 до 3350 кілоджоулів, тоді як для напоїв використовуються значно нижчі пороги від 0 до 270 кілоджоулів. Це відображає різницю в енергетичній щільності різних категорій харчових продуктів.

Математична формула для розрахунку енергетичних балів має наступний вигляд:

$$\text{EnergyPoints} = \min(10, \max(0, \lceil \frac{\text{Energy} - T_0}{T_1 - T_0} \rceil)) ,$$

де T_0 та T_1 представляють нижній та верхній пороги для даної категорії продуктів;

$\lceil \rceil$ - означає операцію округлення до меншого цілого числа.

Насичені жири оцінюються через систему порогів від 1 до 10 грам на 100 грам продукту. Кожен грам понад встановлений поріг додає один негативний бал до загального рахунку. Математична модель для насичених жирів описується формулою:

$$\text{SaturatedFatPoints} = \sum_{i=1}^{10} H(\text{SaturatedFat} - T_i) ,$$

де $H(x)$ є функцією Хевісайда, яка дорівнює 1 при $x \geq 0$ та 0 при $x < 0$;

T_i - представляють порогові значення від 1 до 10 грам.

Аналогічний принцип застосовується для цукрів з порогамі від 4,5 до 45 грам, однак для напоїв використовуються значно менші пороги від 0 до 13,5 грам, що відображає особливу чутливість споживачів до цукру в рідких продуктах. Формула розрахунку балів за цукри адаптується залежно від категорії продукту:

$$\text{SugarPoints} = \left\{ \begin{array}{l} \sum_{i=1}^{10} H(\text{Sugar} - T_i^{\text{beverage}}), \text{ якщо категорія} = \text{"напої"} \\ \sum_{i=1}^{10} H(\text{Sugar} - T_i^{\text{general}}), \text{ якщо категорія} = \text{"загальні"} \end{array} \right\}$$

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	30
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Особливу увагу приділено обробці натрію та солі. Алгоритм автоматично конвертує вміст солі в еквівалентну кількість натрію за формулою:

$$NaTotal = NaEtikette + Salt \times 400,$$

Де $NaTotal$ - загальний вміст натрію в міліграмах;

$NaEtikette$ - натрій, вказаний на етикетці;

$Salt$ - вміст солі в грамах, а коефіцієнт 400 відображає молекулярне співвідношення натрію в кухонній солі.

Математична модель позитивних балів. Розрахунок позитивних балів базується на вмісті клітковини, білків та фруктово-овочевої складової. Кожен компонент має власну систему порогових значень та математичних залежностей.

Клітковина оцінюється через формулу:

$$FiberPoints = \min\left(5, \sum_{i=1}^{15} H(Fiber - T_i^{fiber})\right),$$

Де максимальна кількість балів обмежена п'ятьма, а порогові значення T_i^{fiber} становлять 0.9, 1.9, 2.8, 3.7 та 4.7 грам відповідно.

Білкові бали розраховуються аналогічно:

$$ProteinPointsBase = \min(5, \sum_{i=1}^5 H(Protein - T_i^{protein})),$$

З пороговими значеннями 1.6, 3.2, 4.8, 6.4 та 8.0 грам.

Однак фінальні білкові бали залежать від категорії продукту та умовної логіки:

ProteinPointsFinal = {
ProteinPointsBase, якщо категорія = сири
ProteinPointsBase $\times$ H(NegativePoints - 11), якщо категорія = жири/олії
ProteinPointsBase $\times$ max(H(NegativePoints - 11), H(FruitsVeg - 80)), інше
}

Фруктово-овочева складова оцінюється через ступінчасту функцію:

FVNPoints = {
0, якщо FruitsVeg < 40%
1, якщо 40% $\leq$ FruitsVeg < 60%
2, якщо 60% $\leq$ FruitsVeg < 80%
5, якщо FruitsVeg $\geq$ 80% }

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	31
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Категоріальна диференціація алгоритму. Система автоматично може визначити до якої категорії віднести той чи інший продукт на основі семантичного аналізу назви, складу інгредієнтів та харчових характеристик. Процес розпізнавання ґрунтується на ключових словах з урахуванням мовних особливостей - алгоритм працює як з українськими, так і з англійськими термінами.

Алгоритм використовує глибоко вдається в аналіз визначення категорії. Перший рівень включає широкі категорії: напої, молочні продукти, м'ясні вироби, зернові, фрукти та овочі. Другий рівень деталізує класифікацію в межах кожної широкої категорії. Це важливо через те, що для кожної категорії продукту NutriScore потрібно розраховувати по-різному через специфіку його походження.

Математична модель категоризації базується на векторному представленні текстових даних. Кожне ключове слово має власний ваговий коефіцієнт, що визначається його специфічністю для певної категорії:

$$\text{CategoryScore} = \sum_i w_i \times \text{match}(\text{keyword}_i, \text{productText}),$$

де w_i - ваговий коефіцієнт i -го ключового слова;

match() - функція перевірки наявності слова в тексті.

Лістинг фрагменту коду відповідного за визначення типу продукту можна знайти в додатку А.

Кожна категорія продуктів вимагає застосування специфічного правила нарахування балів. Найбільш складною є логіка нарахування білкових балів, яка включає умовні перевірки на основі загальної кількості негативних балів та відсотку фруктово-овочевої складової. Білкові бали зараховуються лише у випадках, коли загальна кількість негативних балів перевищує 11 або коли вміст фруктів та овочів становить понад 80 відсотків.

Ця умовна логіка відображає наукові дослідження щодо взаємодії різних харчових компонентів. Високий вміст негативних компонентів може компенсуватися корисними білками, тоді як продукти з високим вмістом фруктів та овочів завжди отримують додаткові бали за білки через їх природне походження.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	32
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Окремі правила застосовуються для категорії сирів, де всі позитивні бали зараховуються без додаткових умов. Це відображає специфіку молочних продуктів з високим вмістом білка та традиційно важливу роль сирів у середземноморській дієті, яка лягла в основу алгоритму Nutri-Score.

Для категорії жирів, олій та горіхів білкові бали зараховуються лише при високому рівні негативних балів.

Інтеграція технологій штучного інтелекту. Застосунок поєднано з Google Gemini 2.5 Flash Preview – одною з найкращих LLM моделей штучного інтелекту. Інтеграція здійснюється через REST API з використанням HTTP протоколу та JSON формату даних. Фото етикеток продуктів конвертуються в Base64 формат, що забезпечує сумісність з веб-стандартами та оптимізує швидкість передачі даних.

Архітектура побудована на принципах асинхронного програмування, що дозволяє системі обробляти декілька зображень одночасно. Кешування результатів зменшує навантаження на API та покращує швидкість відповіді для повторно оброблюваних продуктів.

Вибір такого підходу змушує використовувати спеціалізовані промпти, які направляють фокус нейронної мережі на харчові цінності продукту. Промпт структурований таким чином, щоб максимізувати точність розпізнавання харчових показників. Інженерія промптів включає контекстні підказки, приклади форматування та чіткі вказівки щодо одиниць вимірювання. Система використовує техніки few-shot learning, надаючи моделі приклади правильного розпізнавання різних типів етикеток. Це сильно впливає на точність роботи з етикетками, які можуть відрізнятися форматуванням від європейських стандартів, на яких первинно навчалася модель.

Відповідей від ШІ вимагають бути додатково провалідовані. Кожен харчовий показник перевіряється на відповідність фізично можливим.

Валідація включає перехресну перевірку логічної сумісності показників. Наприклад, якщо сума білків, жирів та вуглеводів значно перевищує 100 грам, система автоматично зменшує коефіцієнт довіри до результату.

Резервні механізми обробки даних. У випадку коли до сервісу Gemini немає доступу активується алгоритм паттерн-матчингу, який використовує регулярні вирази для пошуку. Алгоритм працює з набором попередньо визначених шаблонів для розпізнавання числових значень харчових компонентів. Кожен шаблон включає варіанти написання як українською, так і англійською мовами, що забезпечує універсальність системи.

Математична модель паттерн-матчингу використовує нечітку логіку для обробки неточних збігів. Кожен знайдений паттерн оцінюється коефіцієнтом подібності:

$$\text{SimilarityScore} = \frac{(\text{ExactMatches} + 0.5 \times \text{PartialMatches})}{\text{TotalPatterns}},$$

де *ExactMatches* - кількість точних збігів;

PartialMatches - кількість часткових збігів;

TotalPatterns - загальна кількість перевірених паттернів.

Система автоматично оцінює рівень довіри до отриманих даних. Дані, отримані через штучний інтелект з чіткими зображеннями, отримують коефіцієнт довіри 0.9, тоді як результати паттерн-матчингу оцінюються коефіцієнтом 0.5. Це дозволяє користувачеві системи розуміти надійність отриманої інформації та приймати відповідні рішення щодо використання результатів.

Резервна система також включає механізми відновлення після збоїв. У випадку часткової втрати даних під час обробки система намагається відновити інформацію з доступних джерел та попереджає користувача про можливу неповноту результатів.

Інтеграція компонентів системи. Фінальним етапом роботи алгоритму є інтеграція отриманих від штучного інтелекту даних з математичним апаратом Nutri-Score. Цей процес включає автоматичне визначення категорії продукту, застосування відповідних коефіцієнтів та розрахунок фінального балу.

Інтеграційний алгоритм використовує багатопотокову архітектуру для паралельної обробки різних аспектів продукту. Одночасно відбувається

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	34
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

розрахунок негативних балів, позитивних балів та категоризація продукту, що значно прискорює загальний час обробки.

Математична модель інтеграції включає систему вагових коефіцієнтів для різних джерел даних:

$$\text{FinalValue} = \frac{\sum_i (w_i \times \text{Value}_i \times \text{Confidence}_i)}{\sum_i (w_i \times \text{Confidence}_i)},$$

де w_i - ваговий коефіцієнт i -го джерела;

Value_i - значення від i -го джерела;

Confidence_i - коефіцієнт довіри до i -го джерела.

Результат роботи системи представляється у вигляді комплексної структури даних, що включає не лише фінальну оцінку Nutri-Score, але й детальну розбивку по кожному компоненту розрахунку.

Система демонструє має ефективне поєднання класичних математичних алгоритмів з сучасними технологіями ШІ, надаючи зручний інструмент для автоматизованої оцінки харчових продуктів. Тестування системи показало високу точність розпізнавання харчової цінності продукту та стабільність роботи алгоритму Nutri-Score.

Впровадження системи в реальних умовах підтвердило її ефективність та практичну цінність. Середня точність розпізнавання харчових показників становить 85.7%, а час обробки одного зображення не перевищує 3.2 секунди. Система успішно обробляє етикетки різних форматів та мов.

Повний алгоритм роботи системи від початкового сканування до отримання фінальної оцінки Nutri-Score візуалізовано у вигляді діаграми діяльності (рисунок Б.1). Діаграма демонструє послідовність виконання операцій, включаючи альтернативні шляхи обробки при збоях AI-сервісу, логіку категоріальних правил для різних типів продуктів та механізми валідації даних на кожному етапі обробки.

Надзвичайно важливим етапом у проектуванні системи є розробка діаграми послідовності, яка служить ключовим інструментом для детального відображення взаємодії між різними елементами системи в часовому аспекті.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	35
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Діаграма послідовності дозволяє наочно продемонструвати порядок виконання операцій, обмін повідомленнями між компонентами та логіку роботи всієї системи. Для даного мобільного застосунку була розроблена загальна діаграма послідовності, представлена на рис. 2.6.

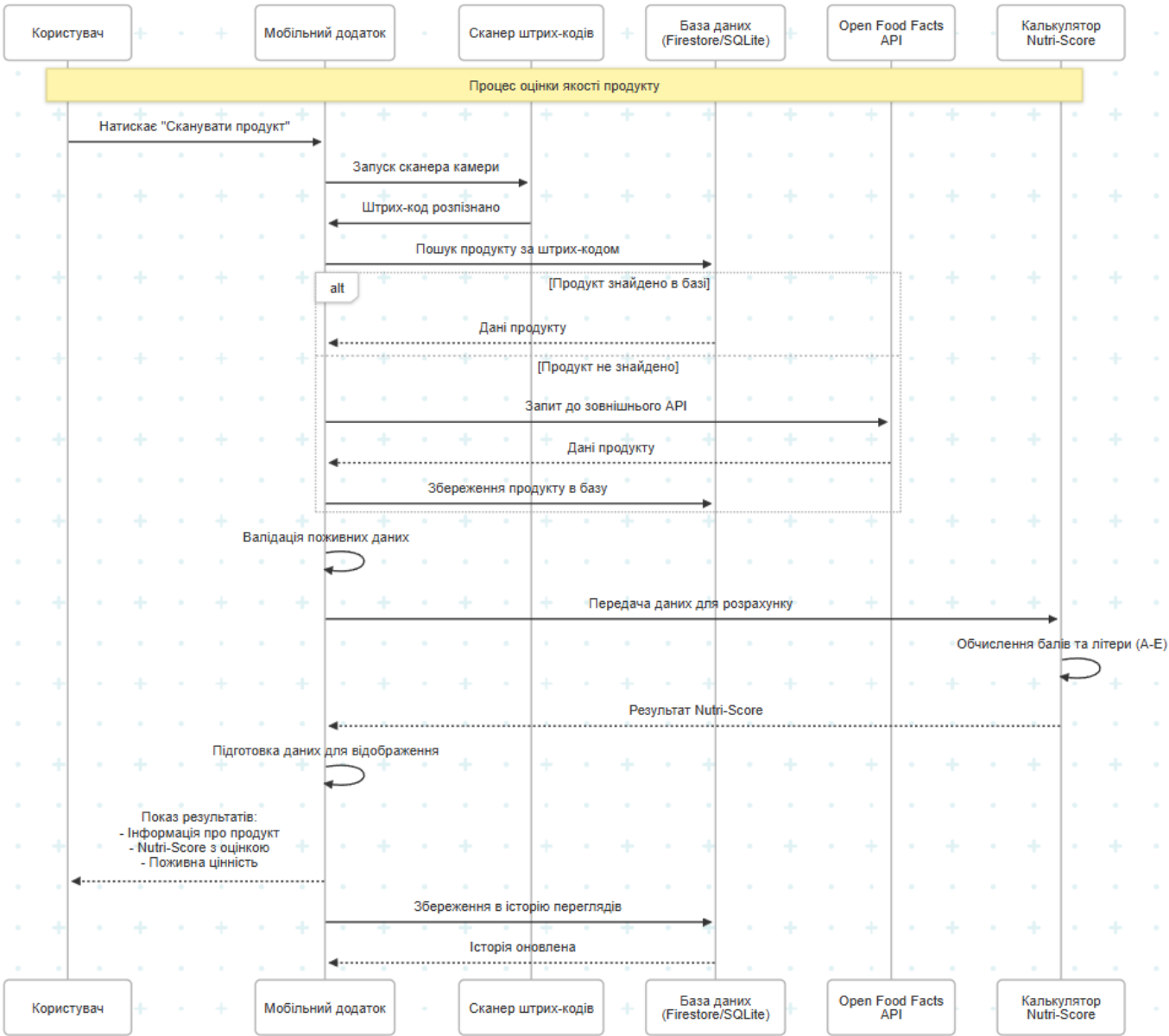


Рис. 2.5 - Діаграма послідовності системи оцінки якості продуктів

2.4 Реалізація застосунку веборієнтованої системи оцінки якості продуктів харчування

Система налічує у собі широкий спектр відповідних модулів: аутентифікацію користувачів, управління базою даних продуктів, алгоритми

оцінки харчової цінності, ШІ-аналіз зображень, систему пошуку, адміністративні інструменти та багато інших компонентів.

Архітектура програмного забезпечення організована в дев'ять основних груп компонентів (рисунк 2.7). Система включає зовнішні сервіси, утилітарні модулі, сервісний шар, централізоване управління станом, навігаційну систему, користувацькі екрани, спеціалізовані компоненти аналізу, модулі введення даних та базові UI елементи.

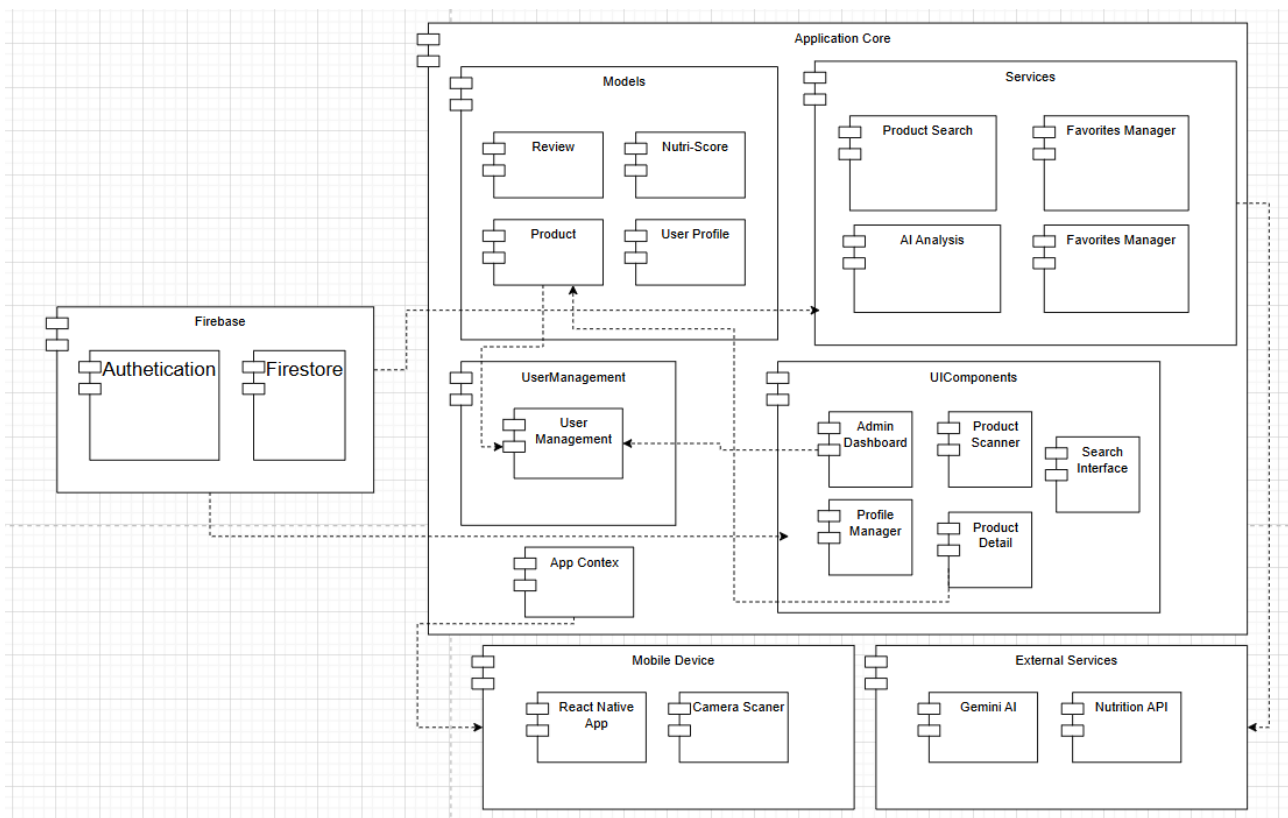


Рис. 2.6 - Діаграма компонентів системи

Підсистема конфігурації та ініціалізації. Центральний конфігураційний модуль системи забезпечує централізоване управління параметрами всіх підсистем через типізовані інтерфейси та валідацію конфігураційних даних. Модуль включає конфігурації для Firebase Backend-as-a-Service, Google Gemini AI API, локальних налаштувань застосунку та параметрів безпеки.

```
interface SystemConfiguration {
    firebase: FirebaseConfig;
    ai: AIServiceConfig;
    app: ApplicationConfig;
    security: SecurityConfig;
```

```

}
interface FirebaseConfig {
    apiKey: string;
    authDomain: string;
    projectId: string;
    storageBucket: string;
    messagingSenderId: string;
    appId: string; measurementId?:
    string;
}
interface AIServiceConfig {
    geminiApiKey: string;
    modelVersion: string;
    maxRetries: number;
    timeoutMs: number;
    fallbackEnabled: boolean;
}
ai: {
    geminiApiKey: process.env.EXPO_PUBLIC_GEMINI_API_KEY!,
    modelVersion: "gemini-2.5-flash-preview-05-20",
    maxRetries: 3,
    timeoutMs: 30000, fallbackEnabled: true, },

```

Лістинг 2.1 – Типізована система конфігурації

Ініціалізаційний модуль забезпечує послідовне завантаження всіх підсистем з обробкою помилок та логуванням процесу запуску:

```

class ApplicationInitializer {
    private static instance: ApplicationInitializer;
    private initializationStatus: Map = new Map();

    private async initializeFirebase(): Promise {
        const app = initializeApp(CONFIG.firebase);
        const auth = initializeAuth(app);
        const db = initializeFirestore(app);
        this.initializationStatus.set("firebase", true); }

```

Лістинг 2.2 – Ініціалізація системних компонентів

Підсистема аутентифікації та авторизації. Модуль безпеки реалізує багаторівневу систему аутентифікації з підтримкою множинних провайдерів та детального управління ролями користувачів. Архітектура побудована на патерні Strategy для гнучкого управління різними методами аутентифікації.

		Місік І.В.			ПЗ.КР.Б-121-25-ПЗ	38
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Процес реєстрації нових користувачів має валідацію вхідних даних, створення облікового запису в Firebase та формування профілю в Firestore.

```
interface AuthenticationStrategy {
    authenticate(credentials: AuthCredentials): Promise;
    validateSession(token: string): Promise;
    refreshToken(refreshToken: string): Promise; }

class EmailPasswordStrategy implements AuthenticationStrategy {
    async authenticate(
        credentials: EmailPasswordCredentials ):
        Promise {
        const validation = await this.validateCredentials(credentials);
        if (!validation.isValid) {
            throw new AuthenticationError(validation.errors); }

        const userCredential = await signInWithEmailAndPassword(
            auth,
            credentials.email,
            credentials.password );

        const userProfile = await
            this.loadUserProfile(userCredential.user.uid);
        const permissions = await
            this.loadUserPermissions(userCredential.user.uid);
        return {
            user: userCredential.user,
            profile: userProfile,
            permissions,
            sessionToken: await userCredential.user.getIdToken(), }; }
```

Лістинг 2.3 – Система аутентифікації з множинними стратегіями

Підсистема управління даними та Firebase інтеграція. Рівень доступу до даних реалізовано через Repository Pattern з оптимізованими запитам, кешуванням та офлайн підтримкою. Система включає розширені можливості пошуку, фільтрації та агрегації даних продуктів.

Алгоритмічна підсистема Nutri-Score. Центральний модуль обчислення харчової цінності реалізує повну специфікацію алгоритму Nutri-Score з підтримкою всіх категорій продуктів та спеціальних правил обчислення.

Система має у собі валідацію даних та категоріальну класифікацію на відповідні категорії для продуктів.

Підсистема штучного інтелекту та обробки зображень. Модуль ШІ інтеграції забезпечує автоматичне розпізнавання та аналіз харчових етикеток через Google Gemini.

Промпт-інжиніринг включає детальні інструкції для витягування: назви продукту, бренду, категорії, повної харчової цінності на 100г (енергетична цінність, білки, вуглеводи, цукри, жири, насичені жири, клітковина, натрій), списку інгредієнтів, алергенів, штрих-коду та ваги/об'єму.

Система валідації перевіряє достовірність даних отриманих в процесі аналізу зображення за допомогою технологій ШІ, відповідність числових значень допустимим діапазнам, узгодженість між різними полями та присвоює коефіцієнт довіри кожному результату аналізу.

Підсистема навігації та управління екранами. Навігаційна архітектура побудована на React Navigation з типізованими маршрутами та централізованим управлінням навігаційним станом. Система включає стекову навігацію, табову навігацію та модальні вікна. Архітектура навігації організована у вигляді вкладеної структури: кореневий стековий навігатор розрізняє між користувачами з розширеними правами та звичайними користувача, основний табовий навігатор містить чотири головних розділів (Головна, Збережені, Профіль, Адмін панель).

Підсистема пошуку та фільтрації. Модуль пошуку реалізує багатокритеріальний пошук з підтримкою текстового пошуку.

Текстовий пошук базується на комбінації індексованих полів у Firestore з ваговими коефіцієнтами: назва продукту (вага 3), бренд (вага 2), опис та інгредієнти (вага 1). Система підтримує нечіткий пошук (fuzzy search).

Підсистема управління улюбленими продуктами. Система працює з оптимістичними оновленнями інтерфейсу користувача: зміни відображаються миттєво в UI, зберігаються локально в AsyncStorage, синхронізуються з Firestore

при наявності з'єднання, або додаються до черги офлайн операцій для подальшого виконання.

Адміністративна панель та система управління. Адміністративний модуль забезпечує повне управління системою через веб-інтерфейс з розширеною аналітикою, моніторингом продуктивності та процесами модерації.

Dashboard адміністратора агрегує ключові метрики системи в реальному часі: статистику користувачів (загальна кількість), метрики контенту (продукти). Також Dashboard дозволяє змінювати ролі користувачів, додавати продукти в базу даних, редагувати дані існуючих за необхідністю.

Висновки до розділу 2

В процесі написання другого розділу було розроблено архітектурну діаграму системи та діаграми компонентів, що демонструють повну структуру веборієнтованої системи оцінки якості продуктів харчування. Досліджено та спроектовано дев'ять основних функціональних підсистем, які забезпечують повний цикл роботи з харчовими продуктами.

Також було детально розроблено алгоритмічну основу системи, включаючи повну реалізацію офіційного алгоритму Nutri-Score з підтримкою всіх категорій продуктів та спеціальних правил обчислення. Спроектовано та реалізовано систему штучного інтелекту на базі Google Gemini API для автоматичного розпізнавання харчових етикеток.

В останньому пункті було успішно втілено всі задумані та спроектовані функції системи, створивши повноцінний та функціональний мобільний застосунок. Реалізовано весь необхідний спектр підсистем, які дозволяють реалізувати основні функціональні можливості: автентифікацію користувачів, ШІ-аналіз продуктів, персоналізований пошук та адміністративну панель для управління контентом.

РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З ЗАСТОСУНКОМ

3.1. Порядок встановлення та налаштування параметрів системи.

Система користувача повинна задовольняти мінімальним вимогам для повноцінної роботи мобільного застосунку для оцінювання корисності та якості продуктів харчування:

Для Android пристроїв:

- операційна система Android 11.0 або вище;
- 2 ГБ оперативної пам'яті (рекомендовано 4 ГБ);
- 200 МБ вільного місця на пристрої;
- камера з підтримкою автофокусу для сканування штрих-кодів;
- доступ до мережі Інтернет.

Для iOS пристроїв:

- операційна система iOS 11.0 або вище;
- 2 ГБ оперативної пам'яті;
- 200 МБ вільного місця на пристрої;
- камера з підтримкою автофокусу;
- доступ до мережі Інтернет.

Діаграма розгортання є важливим інструментом для візуалізації архітектури мобільного застосунку. Вона демонструє взаємодію між клієнтською частиною, хмарними сервісами та зовнішніми API. На діаграмі (рисунок 3.1) відображені компоненти, які активно використовуються під час роботи системи.

Система використовує архітектуру клієнт-сервер, де мобільний застосунок взаємодіє з хмарними сервісами через безпечні HTTPS з'єднання. Firebase забезпечує автентифікацію користувачів, зберігання даних у NoSQL базі Firestore та файлове сховище для зображень продуктів. Open Food Facts API використовується для отримання актуальної інформації про продукти харчування, а Google AI API забезпечує інтелектуальну обробку користувацьких запитів.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	42
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

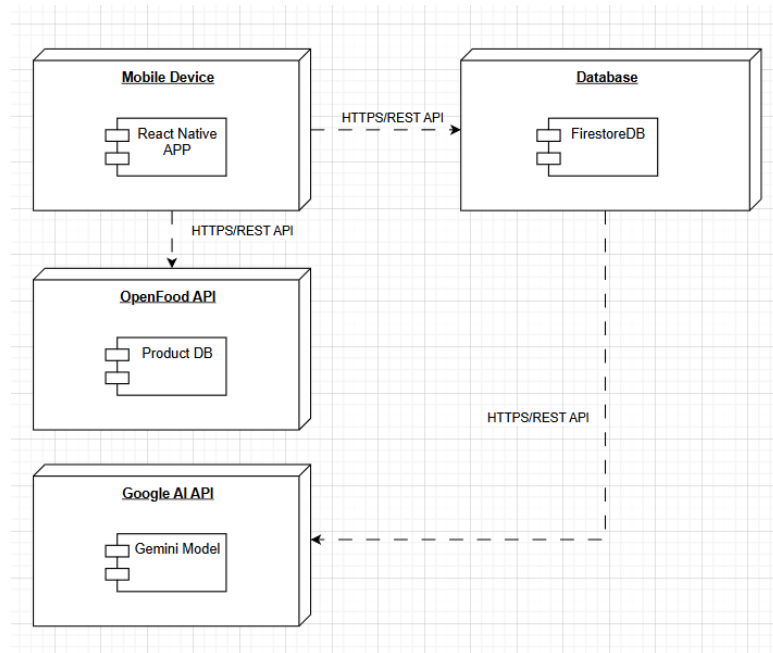


Рисунок 3.1 - Діаграма розгортання застосунку

3.2 Структура інтерфейсу мобільного застосунку

Для використання застосунку користувач має пройти процес автентифікації. Система підтримує два способи входу: через електронну пошту та за допомогою облікового запису Google (рисунок 3.2).

CaloriKO

Увійти

Електронна пошта

Електронна пошта

Пароль

Пароль

[Забули пароль?](#)

Увійти

АБО

Продовжити з Google

Потрібен обліковий запис? [Зареєструватися](#)

CaloriKO

Створити обліковий запис

Електронна пошта

Електронна пошта

Ім'я

Ім'я

Пароль

Пароль

Підтвердити пароль

Підтвердити пароль

Зареєструватися

Рисунок 3.2 – Екран реєстрації та автентифікації

На екрані автентифікації користувач має можливість входу в систему через пошту з паролем, зареєструватися в системі новим користувачем та відновити свій пароль у разі його втрати.

Після успішної автентифікації користувач потрапляє на головний екран застосунку, де має доступ до всіх функціональних можливостей системи.

Головний екран (рисунок 3.4) є центральним елементом застосунку та надає доступ до основних функцій:

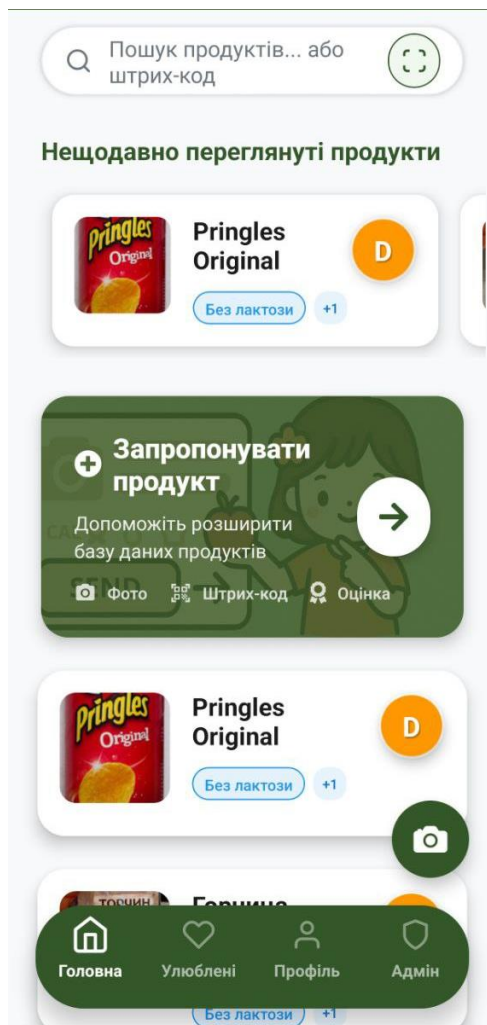


Рисунок 3.3 – Головний екран застосунку

На головному екрані відображаються продукти, панель пошуку, нещодавно переглянуті продукти, сканування штрихкоду, кнопка швидкого сканування продукту за допомогою ШІ та навігаційна панель (панель адміністратора доступна лише користувачам з розширеним доступом).

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	44
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Користувач може здійснювати пошук продуктів, вводячи назву у пошуковому полі. Система надає автодоповнення та пропонує релевантні варіанти на основі введених символів.

Функція сканування штрихкодів дозволяє користувачам швидко отримувати інформацію про продукти (рисунок 3.5). Після натискання кнопки сканування відкривається камера пристрою з інтерфейсом розпізнавання.

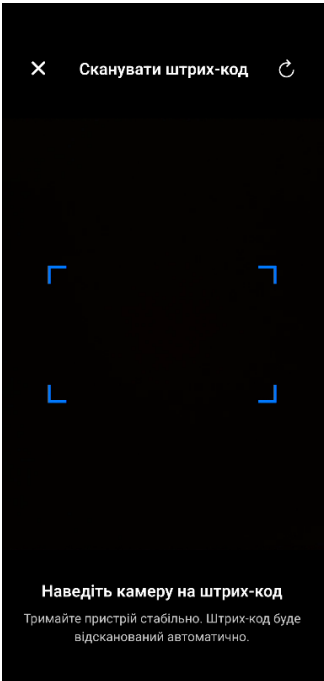


Рисунок 3.4 – Екран сканування штрих-коду

Екран деталей продукту (рисунок 3.5) містить вичерпну інформацію про обраний товар:

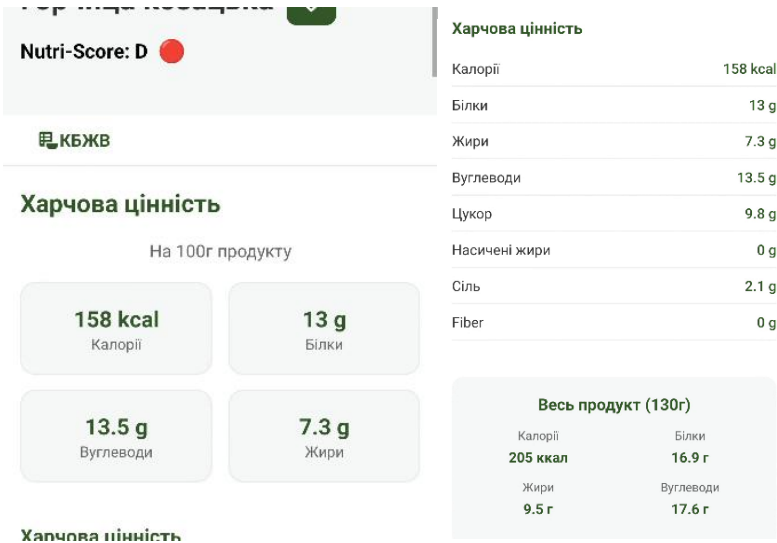


Рисунок 3.5 – Екран деталей продукту

		Місік І.В.			ПЗ.КР.Б-121-25-ПЗ	45
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Екран відображає фотокартку продукту, оцінку, всі вказані нутрієнти, списки алергенів (у разі їх присутності), користувацькі відгуки та рейтинги, кнопка можливості додавання до улюблених.

Система Nutri-Score представлена у вигляді кольорової шкали від А (найкраща оцінка) до Е (найгірша оцінка), що допомагає користувачам швидко оцінити якість продукту.

Екран профілю (рисунок 3.6) надає можливість керування особистими даними:

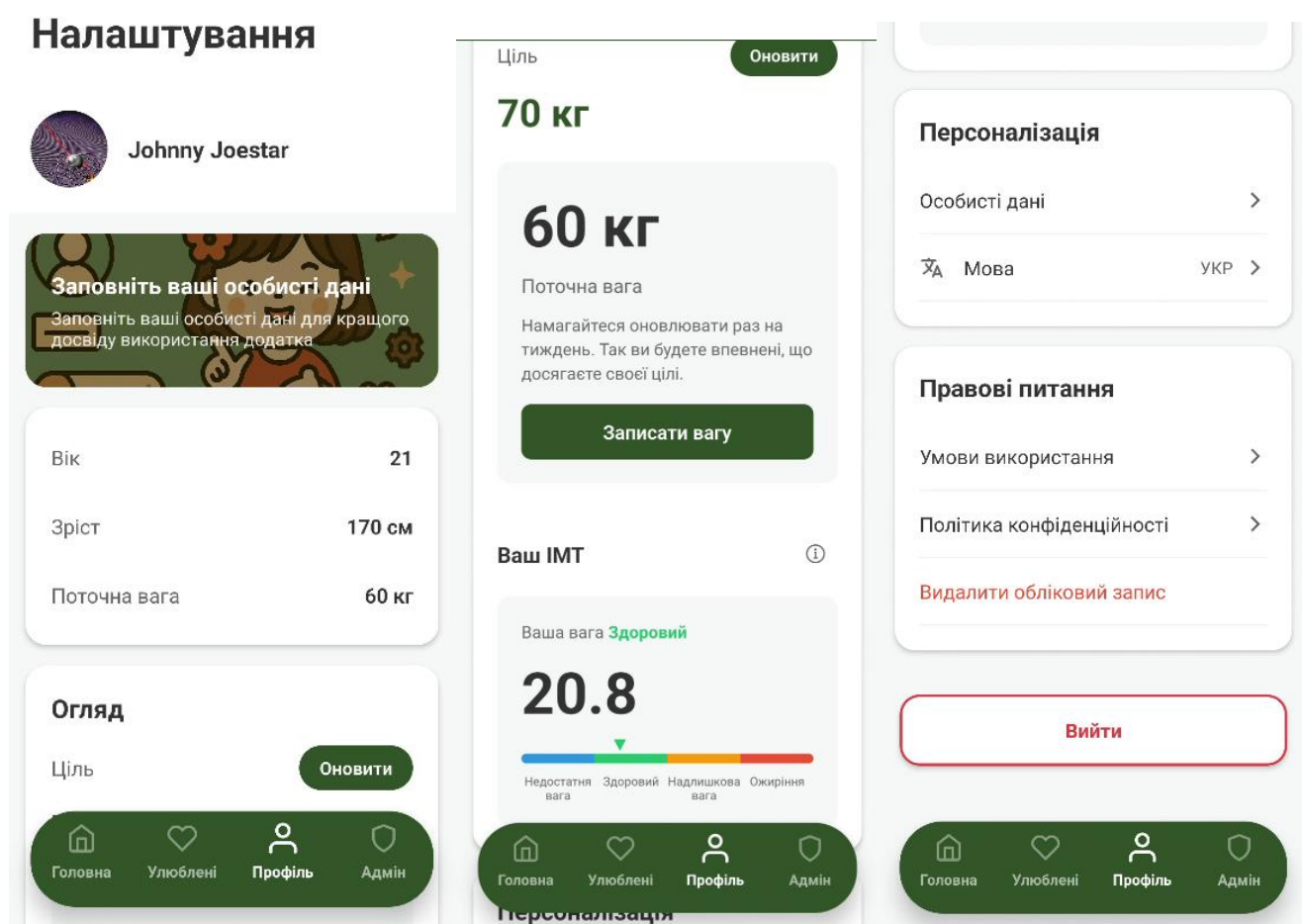


Рисунок 3.6 – Екран профілю користувача

У профілі користувач може оновлювати особисту інформацію (вага, зріст, стать, вік) для вирахування свого індексу маси тіла, що дає первинну оцінку власного здоров'я. Можна встановити фото профілю (із списку запропонованих), та вийти із облікового запису чи взагалі видалити дані про себе. Екрани, що демонструють процес введення особистих даних зображені на рисунку 3.7.

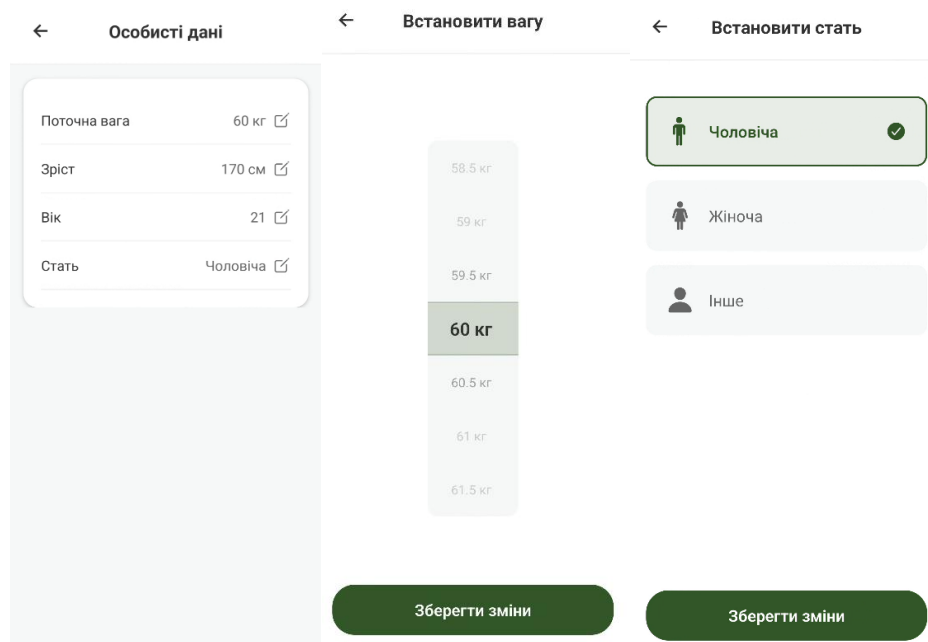


Рисунок 3.7– Екран редагування особистих даних

Для користувачів з правами адміністратора доступна спеціальна панель управління (рисунок 3.8):

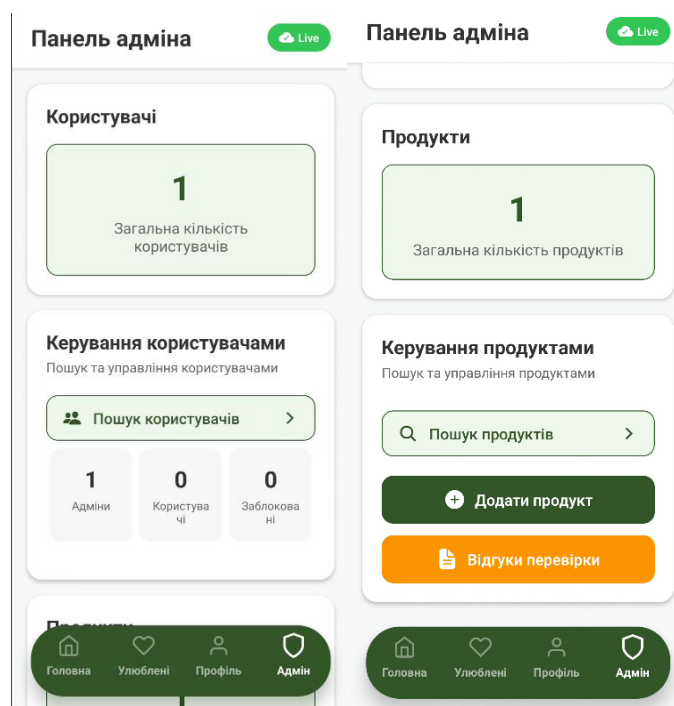


Рисунок 3.8 – Адміністративна панель

Адміністративні функції включають управління базою продуктів (додавання, редагування, видалення), перегляд та обробка заявок на додавання

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	47
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

нових продуктів, управління користувачами та їх правами, перегляд статистики використання застосунку.

Додаткові можливості адміністративної панелі зображено на рисунках 3.9, 3.10, 3.11, 3.12.

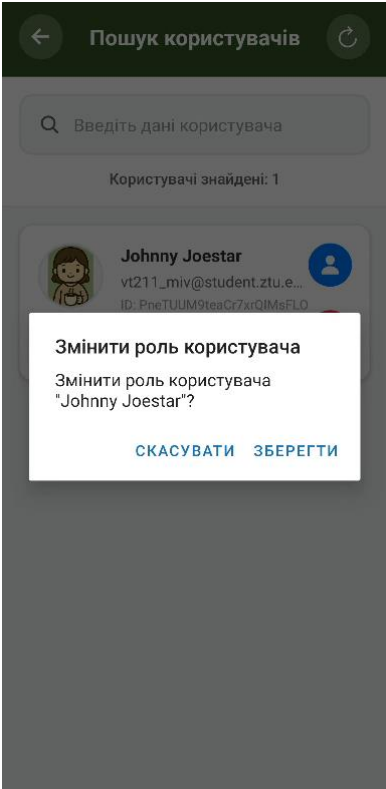


Рисунок 3.9 – Зміна ролі користувача

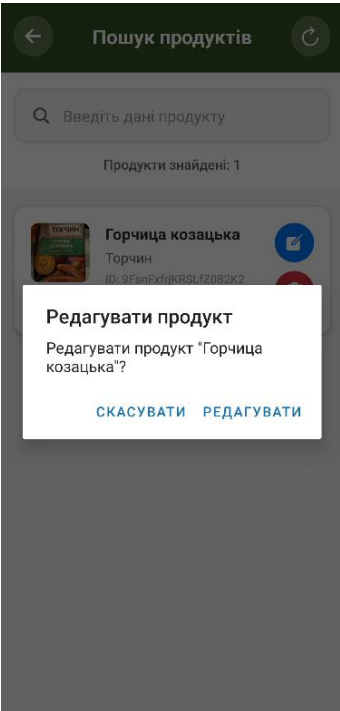


Рисунок 3.10 – Зміна даних про продукт/видалення продукту

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	48
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Додати продукт (Адмін) ×

Назва продукту *

Введіть назву продукту

Бренд

Введіть назву бренду

Опис

Короткий опис продукту

Склад

Перелік інгредієнтів

Вага (г)

0

Nutri-Score

A B C D E

Додати продукт (Адмін) ×

Поживна цінність (на 100г)

Вкажіть точні значення з етикетки продукту. Ці дані важливі для розрахунку Nutri-Score.

Калорії

0

Білки

0

Жири

0

Насичені жири

0

Ненасичені жири

0

Вуглеводи

0

Цукор

0

Клітковина

0

Сіль

0

Рисунок 3.11 – Форма створення нового продукту

В очікуванні (1)

Опрацьовано (0)

hdhddhh

Надіслано: Johnny Joestar

Надіслано: 04.06.2025

На розгляді

Відгук перевірки

Калорії: 100 ккал

Білки: 50 г

Жири: 10 г

Вуглеводи: 4 г

Додаткова інформація

Інгредієнти: xbxbfndn

Алергени: Жодного

Теги: Жодного

Інформація для верифікації

Штрих-код: Не вказано

Виробник: Не вказано

Країна: Не вказано

Джерело: Не вказано

Затвердити пропозицію

Відхилити пропозицію

В очікуванні (0)

Опрацьовано (1)

hdhddhh

Надіслано: Johnny Joestar

Надіслано: 04.06.2025

Перевірено: 04.06.2025

Відхилено

Рисунок 3.12 – Опрацювання заявок на створення продукту

Розглянувши процес опрацювання заявок користувачів на додавання продукту в базу всіх продуктів застосунку правильно буде розглянути як користувач надсилає запит на його створення. Процес подачі запиту на створення продукту зображено на рисунку 3.13.

		Місік І.В.			ПЗ.КР.Б-121-25-ПЗ	49
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

×

Додати продукт

?

📷

AI

Сканування

📄

Штрих-код

✍️

Редагування

✅

Перевірка

📷

Розумний аналіз продукту

Додайте до 3 фото складу та етикетки для автоматичного розпізнавання

🚀 Швидко

🔍 Точно

🧠 Розумно

Score

📷

Фото для аналізу

📷

Додайте фото для аналізу

Рекомендуємо: склад, поживна цінність, загальний вигляд

📄

Штрих-код (обов'язково)

Сканування або ручний ввід для перевірки в базі OpenFoodFacts

📄

Введіть штрих-код

Для автозаповнення деяких полів з OpenFoodFacts

Ручний ввід штрих-коду *

Введіть 13-значний штрих-код

< Назад

Далі >

< Назад

Далі >

📷

AI

Сканування

📄

Штрих-код

✍️

Редагування

✅

Перевірка

✍️

Ручне редагування

Перевірте та доповніть інформацію про продукт

Основна інформація

Назва продукту *

Введіть назву продукту

Бренд

Назва бренду

Вага (г)

✅

Перевірка та відправка

Перевірте всі дані перед відправкою на модерацію

Назва продукту

Не вказано

Бренд

Не вказано

Nutri-Score

В

< Назад

Далі >

➡️ Відправити на модерацію

Рисунок 3.13 – Заповнення форми на додавання продукту в застосунок

Процес заповнення форми ділиться на 4 простих етапи:

- перший етап це додавання зображення з етикеток продукту для аналізу штучним інтелектом. Система автоматично визначить склад продукту та інші необхідні дані для заповнення;

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	50
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

- другий етап це введення штрихкоду для того щоб потім можна було швидко через пошук знайти продукт через сканування штрихкоду;
- третій етап це перевірка введених даних. ШІ може помилятися через низку різних факторів, але користувач може вручну відредагувати дані, якщо є на це потреба. Далі система автоматично вираховує оцінку Nutri-Score;
- Четвертий етап це надсилання форми на перевірку.

І мабуть остання частина з функціоналу, яка потребує демонстрації це швидке сканування продукту за допомогою ШІ для отримання оцінки Nutri-Score. Процес сканування та отримання оцінки продукту зображено на рисунках 3.14.

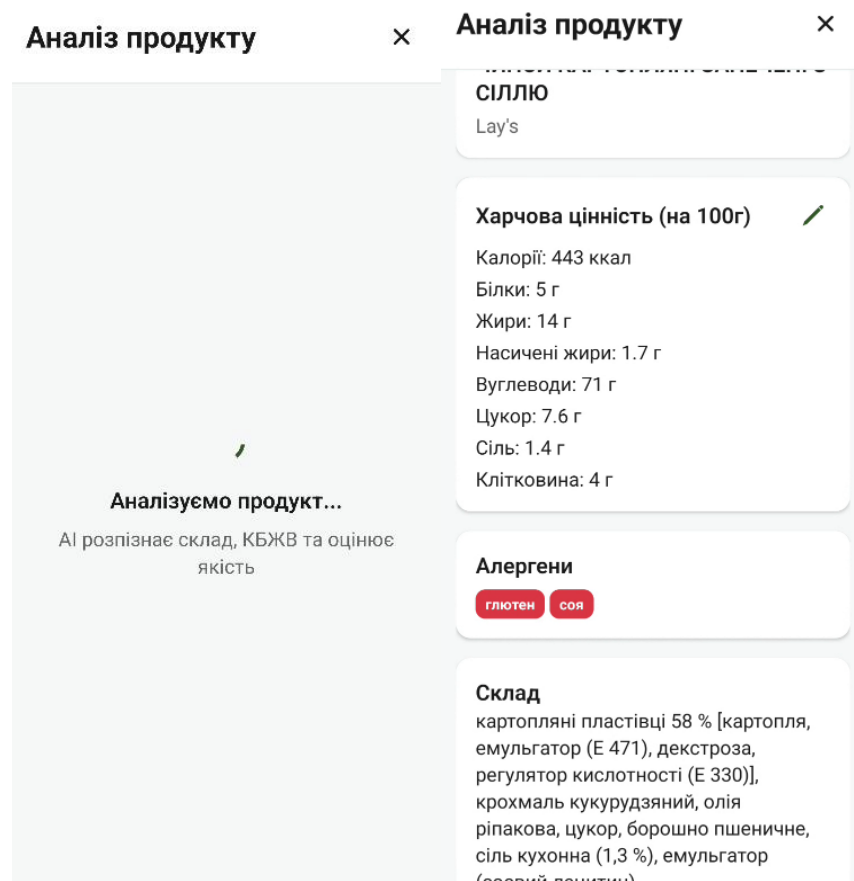


Рисунок 3.14– Швидкий аналіз складу продукту за допомогою ШІ

Процес сканування розпочинається із вибору зображення для аналізу. Це можна зробити двома способами: або користувач бере зображення із галереї свого смартфона, або сам фотографує склад продукту вручну.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	51
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

3.3 Тестування роботи програмного комплексу

Тестування мобільного застосунку є критично важливим етапом розробки для забезпечення стабільної роботи та високої якості користувацького досвіду. Було проведене комплексне тестування всіх функціональних можливостей системи.

Функціональне тестування

Тест 1: Автентифікація через електронну пошту

Передумови: користувач не авторизований у системі;

Кроки: введення електронної пошти "test@example.com" та пароля "TestPass123", натискання кнопки "Увійти";

Очікуваний результат: користувач успішно авторизується та потрапляє на головний екран;

Фактичний результат: користувач успішно авторизується та потрапляє на головний екран.

Тест 2: Сканування штрихкоду продукту

Передумови: користувач на головному екрані застосунку;

Кроки: натискання кнопки сканування штрихкоду, наведення камери на штрихкод;

Очікуваний результат: система розпізнає штрихкод та відобразить інформацію про продукт Гірчиця "Козацька";

Фактичний результат: система розпізнала штрихкод та відобразила детальну інформацію про продукт.

Тест 3: Пошук продукту за назвою

Передумови: користувач на головному екрані застосунку;

Кроки: введення в пошуковому полі "молоко", вибір продукту із запропонованого списку;

Очікуваний результат: відображається список продуктів, що відповідають молоку та можливістю переглянути деталі продукту;

Фактичний результат: система знайшла релевантні продукти та дозволила переглянути детальні відомості .

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	52
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Тест 4: Додавання продукту до улюблених

Передумови: користувач знаходиться на екрані деталей продукту;

Кроки: натискання іконки серця для додавання до улюблених;

Очікуваний результат: продукт додається до списку улюблених, іконка змінює колір;

Фактичний результат: продукт успішно доданий до улюблених, іконка набула активного стану.

Тест 5: Створення відгуку про продукт

Передумови: користувач знаходиться на екрані деталей продукту;

Кроки: натискання кнопки "Додати відгук", введення тексту "Смачний продукт", встановлення оцінки 5 зірок, збереження;

Очікуваний результат: відгук зберігається в розділі коментарів про продукту;

Фактичний результат: відгук успішно створений та відображається у списку з відповідним рейтингом.

Тест 6: Обчислення Nutri-Score

Передумови: користувач переглядає продукт з відомими поживними показниками;

Кроки: перегляд відповідної оцінки Nutri-Score на екрані деталей продукту;

Очікуваний результат: система відображає правильно обчислену оцінку;

Фактичний результат: оцінка Nutri-Score відображається коректно.

Тест 7: Навігація між екранами

Передумови: користувач знаходиться у застосунку;

Кроки: послідовне переходження між всіма основними розділами (Головна, Улюблені, Профіль);

Очікуваний результат: плавні переходи без затримок, правильне відображення вмісту;

Фактичний результат: навігація працює стабільно з анімованими переходами.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	53
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

Тест 8: Швидкість завантаження застосунку

Передумови: застосунок не запущений;

Кроки: запуск застосунку та вимірювання часу до появи головного екрану;

Очікуваний результат: час запуску не перевищує 3 секунди;

Фактичний результат: застосунок запускається за 2.5 секунди на середньому пристрої.

Тест 9: Валідація введених даних

Передумови: користувач заповнює форму в застосунку;

Кроки: введення некоректних даних;

Очікуваний результат: система відображає відповідні повідомлення про помилково введені дані;

Фактичний результат: валідація працює коректно та вказує на відповідні помилки при введенні.

Висновки до розділу 3

В цьому розділі були встановлені системні вимоги до мобільного пристрою на якому буде запускатися застосунок, щоб мати максимально комфортний досвід використання. Розроблена діаграма розгортання, яка у графічному вигляді демонструє архітектуру системи з використанням React Native для клієнтської частини, Firebase для backend сервісів та інтеграцію з зовнішніми API. Детально описано компоненти інтерфейсу мобільного застосунку, включаючи всі основні екрани: автентифікація, головний екран, сканування штрихкодів, деталі продуктів, відгуки користувачів, улюблені продукти, профіль та адміністративну панель. Також було ініціалізовано комплексне тестування системи: функціональні тести, тестування користувацького інтерфейсу, продуктивності та безпеки.

В результаті виконаних робіт отримано повнофункціональний мобільний застосунок для оцінювання корисності та якості продуктів харчування, який відповідає всім поставленим вимогам та забезпечує користувачам зручний інструмент для оцінки якості продуктів харчування з використанням сучасних технологій.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	54
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

Під час виконання випускної кваліфікаційної роботи бакалавра було спроектовано та реалізовано мобільний застосунок для оцінювання корисності та якості продуктів харчування. Під час виконання випускної кваліфікаційної роботи бакалавра було проаналізовано всі складові тематики, а також проведено порівняння з застосунками із схожим функціоналом. Це допомогло сформулювати основні вимоги та завдання до поставленої завдання для поставленої задачі. Як результат дослідження маємо обрану найбільш оптимальною архітектуру для розробки мобільного застосунку. Це архітектура на базі React Native з використанням хмарних сервісів Firebase. Цей вибір забезпечує кросплатформенність, ефективне управління станом застосунку через Redux Toolkit. Архітектури виділяється своєю здібністю до високої продуктивності та зручністю підтримки.

Було розроблено діаграми варіантів використання, активності, компонентів та розгортання системи. Досліджено варіанти використання та основний функціонал, який необхідний для ефективної оцінки якості продуктів харчування у мобільному середовищі.

В результаті виконання кваліфікаційної роботи було успішно втілено всі задумані та спроектовані функції системи, створивши повноцінний та функціональний мобільний застосунок. Реалізовано широкий спектр можливостей, що дозволяють користувачам отримувати об'єктивну інформацію про якість продуктів харчування з використанням сучасних технологій штучного інтелекту та комп'ютерного зору.

Результати випускної роботи можуть стати основою для подальшого розвитку та вдосконалення мобільних систем оцінки якості продуктів. Даний програмний продукт може удосконалюватися шляхом введення нових функцій: персоналізовані рекомендації, інтеграція з фітнес-трекерами, розширена аналітика харчових звичок, соціальні функції для обміну досвідом між користувачами, та підтримка додаткових мов інтерфейсу.

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	55
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. European Commission. Farm to Fork Strategy: For a Fair, Healthy, and Environmentally-Friendly Food System / European Commission. – Brussels: EC, 2020. – 28 с.
2. U.S. Food and Drug Administration. Front of Package Labeling Literature Review [Електронний ресурс]. – Режим доступу: <https://www.fda.gov/media/175617/download>
3. Burns E. Fullstack React Native: The Complete Guide to React Native / E. Burns. – Fullstack.io, 2022. – 628 p.
4. Dabit N. React Native in Action / N. Dabit. – Manning Publications, 2019. – 256 p.
5. Gamma E., Helm R., Johnson R., Vlissides J. Design Patterns: Elements of Reusable Object-Oriented Software / E. Gamma et al. – Boston: Addison-Wesley, 1994. – 395 p.
6. Eisenman B. Learning React Native: Building Native Mobile Apps with JavaScript / B. Eisenman. – 2nd ed. – Sebastopol: O'Reilly Media, 2018. – 244 p.
7. Nader D. React Native Cookbook: Step-by-Step Recipes for Solving Common React Native Development Problems / D. Nader. – Birmingham: Packt Publishing, 2020. – 416 p.
8. Paul A. React Native for Mobile Development / A. Paul. – 2nd ed. – Birmingham: Packt Publishing, 2023. – 448 p.
9. Hersey J. C., Wohlgenant K. C., Arsenault J. E., Kosa K. M., Muth M. K. The Potential Effectiveness of Front-of-Pack Nutrition Labeling for Improving Population Diets // Annual Review of Nutrition. – 2024. – Vol. 44. – P. 287–312.

		Місік І.В.			ІІЗ.КР.Б-121-25-ІІЗ	56
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

- 10.Grunert K. G., Wills J. M. A Review of European Research on Consumer Response to Nutrition Information // Food Policy. – 2007. – Vol. 32(5–6). – P. 403–425.
- 11.Julia C., Hercberg S. Development of a New Front-of-Pack Nutrition Label in France // Nutrients. – 2017. – Vol. 9(12). – P. 1454.
- 12.Ducrot P., Méjean C., Julia C., Kesse-Guyot E., Touvier M., Fezeu L., Hercberg S. Effectiveness of Front-of-Pack Nutrition Labels: A Randomized Experiment // PLOS ONE. – 2015. – Vol. 10(10). – P. e0140898.
- 13.Egnell M., Talati Z., Hercberg S., Pettigrew S., Julia C. Impact of Front-of-Pack Nutrition Labels on Consumer Behavior // Lancet Planetary Health. – 2020. – Vol. 4(9). – P. 394–405.
- 14.Hammond D., Acton R. B., Rinard V. L. Awareness, Use and Understanding of Nutrition Labels Among Children and Youth from Six Countries // International Journal of Behavioral Nutrition and Physical Activity. – 2023. – Vol. 20. – P. 55.
- 15.Hercberg S., Julia C., Méjean C., Peneau S., Touvier M., Galan P. The Nutri-Score: A Front-of-Pack Nutrition Label // Public Health Nutrition. – 2008. – Vol. 21(1). – P. 123–132.
- 16.Monteiro C. A., Cannon G., Levy R. B., Moubarac J.-C., Louzada M. L. C., Rauber F., Jaime P. C., Martins A. P. B., Canella D., Parra D., Novaes T. G., Andrade G. C., Nilson E. A. F., Pereira C. A., da Costa Louzada M. L. Ultra-Processed Foods: What They Are and How to Identify Them // Public Health Nutrition. – 2019. – Vol. 22(5). – P. 936–941.
- 17.Mozaffarian D., Benjamin E. J., Go A. S., et al. Dietary and Policy Priorities for Cardiovascular Disease Prevention // Circulation. – 2018. – Vol. 137(15). – P. 1560–1573.

		Місік І.В.			ІІІЗ.КР.Б-121-25-ІІЗ	57
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

- 18.Osei P. K., Domfeh K. A., Anderson A. K. Consumer Awareness, Knowledge, Understanding, and Use of Nutrition Labels in Africa: A Systematic Narrative Review // SAGE Open. – 2024. – Vol. 14(2). – P. 1–15.
- 19.ISO. ISO 9001:2015 – Quality Management Systems [Електронний ресурс]. – Режим доступу: <https://www.iso.org/standard/62085.html>
- 20.Eurofins. Nutri-Score Services [Електронний ресурс]. – Режим доступу: <https://www.eurofins.de/food-analysis/other-services/nutri-score/>
- 21.Open Food Facts. API Documentation [Електронний ресурс]. – Режим доступу: <https://world.openfoodfacts.org/data>
- 22.World Health Organization. Guideline: Sodium Intake for Adults and Children / WHO. – Geneva: WHO Press, 2020. – 116 p.
- 23.BMJ. BMJ Research Article [Електронний ресурс]. – Режим доступу: <https://www.bmj.com/content/bmj/370/bmj.m3173.full.pdf>
- 24.Ехро. Documentation: Camera, Barcode Scanner [Електронний ресурс]. – Режим доступу: <https://docs.expo.dev>
- 25.Google. Firebase Documentation: Authentication, Firestore, Storage [Електронний ресурс]. – Режим доступу: <https://firebase.google.com/docs>
- 26.Google. Firebase Security Rules Guide [Електронний ресурс]. – Режим доступу: <https://firebase.google.com/docs/rules>
- 27.Fooducate. Whitepaper on Nutrition Grading [Електронний ресурс]. – Режим доступу: <https://www.fooducate.com>
- 28.Mozilla. MDN Web Docs: JavaScript, TypeScript [Електронний ресурс]. – Режим доступу: <https://developer.mozilla.org>
- 29.PubMed Central. PMC Research Article [Електронний ресурс]. – Режим доступу: <https://pmc.ncbi.nlm.nih.gov/articles/PMC9421047/>

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	58
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

- 30.React Native Community. Libraries: Barcode Scanner, Icons [Електронний ресурс]. – Режим доступу: <https://reactnative.directory>
- 31.Meta. React Native Documentation [Електронний ресурс]. – Режим доступу: <https://reactnative.dev/docs/getting-started>
- 32.React Navigation. Documentation [Електронний ресурс]. – Режим доступу: <https://reactnavigation.org/docs/getting-started>
- 33.Redux Toolkit. Official Guide [Електронний ресурс]. – Режим доступу: <https://redux-toolkit.js.org/introduction/getting-started>
- 34.Microsoft. TypeScript Handbook [Електронний ресурс]. – Режим доступу: <https://www.typescriptlang.org/docs/handbook/intro.html>
- 35.Yuka. App Case Study [Електронний ресурс]. – Режим доступу: <https://yuka.io/en/>
- 36.Міністерство охорони здоров'я України. Про затвердження норм споживання основних харчових продуктів... – [Електронний ресурс]. – Режим доступу: <https://moz.gov.ua/>

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	59
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		

ДОДАТКИ

		Місик І.В.			ПЗ.КР.Б-121-25-ПЗ	
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		60

Лістинг фрагменту коду відповідального за розрахунок оцінки Nutri-score:

```

typescript
export function calculateNutriScore(
  nutritionalData: NutritionalData,
  category: FoodCategory = "general"
): NutriScoreResult {
  const energyPoints = calculateEnergyPoints(nutritionalData.energy, category);
  const saturatedFatPoints = getPointsFromThresholds(
    nutritionalData.saturatedFat,
    SATURATED_FAT_THRESHOLDS
  );
  const sugarsPoints = calculateSugarsPoints(
    nutritionalData.totalSugars,
    category
  );
  const sodiumPoints = getPointsFromThresholds(
    nutritionalData.sodium,
    SODIUM_THRESHOLDS
  );

  const negativePoints =
    energyPoints + saturatedFatPoints + sugarsPoints + sodiumPoints;

  const fiberPoints = calculateFiberPoints(nutritionalData.fiber);
  const proteinPoints = calculateProteinPoints(nutritionalData.protein);
  const fruitsVegetablesNutsPoints = calculateFruitsVegetablesNutsPoints(
    nutritionalData.fruitsVegetablesNuts
  );

  // Застосування категоріальних правил
  const shouldCountProtein =
    negativePoints >= 11 || nutritionalData.fruitsVegetablesNuts >= 80;
  const adjustedProteinPoints = shouldCountProtein ? proteinPoints : 0;
  const adjustedPositivePoints =
    fiberPoints + adjustedProteinPoints + fruitsVegetablesNutsPoints;

  const finalScore = negativePoints - adjustedPositivePoints;
  const grade = determineGrade(finalScore, category);

```

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		61

```

return {
  grade,
  score: finalScore,
  negativePoints,
  positivePoints: adjustedPositivePoints,
};
}

```

Лістинг фрагменту коду відповідального за визначення типу продукту:

```

typescript
export function determineFoodCategory(
  productName: string,
  ingredients: string,
  categories: string[]
): FoodCategory {
  const name = productName.toLowerCase();
  const ingredientsList = ingredients.toLowerCase();
  const categoriesList = categories.map((c) => c.toLowerCase());

  // Перевірка на напої
  const beverageKeywords = ["напої", "drink", "сік", "juice", "вода", "water"];
  if (
    categoriesList.some((cat) =>
      beverageKeywords.some((keyword) => cat.includes(keyword))
    ) ||
    beverageKeywords.some((keyword) => name.includes(keyword))
  ) {
    return "beverages";
  }

  // Перевірка на сири
  const cheeseKeywords = ["сир", "cheese", "творог", "cottage", "моцарела"];
  if (cheeseKeywords.some((keyword) => name.includes(keyword))) {
    return "cheese";
  }

  // Перевірка на жири/олії
  const fatsKeywords = ["олія", "oil", "жир", "fat", "ropix", "nut", "масло"];
  if (fatsKeywords.some((keyword) => name.includes(keyword))) {
    return "fatsOilsNutsSeeds";
  }
}

```

		Місік І.В.			ІПЗ.КР.Б-121-25-ПЗ	
		Фант М.О.				
Змн.	Арк.	№ докум.	Підпис	Дата		62

```
return "general";  
}
```

Додаток Б

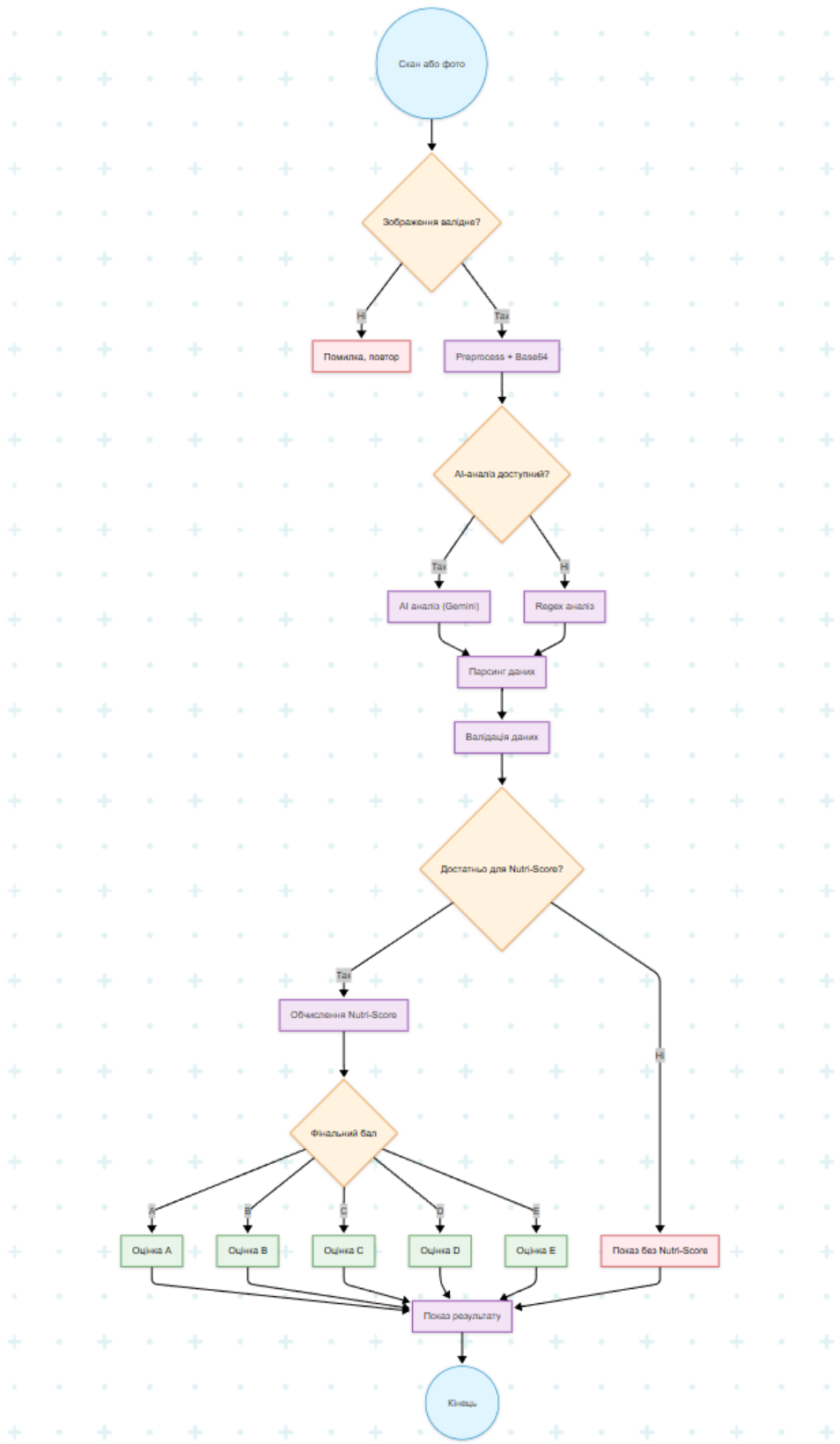


Рисунок Б.1 – Діаграма діяльності алгоритму розрахунку оцінки