

ПРОГРАМНА СИСТЕМА РОЗРАХУНКУ КАЛОРИЙ

У роботі описано поняття і суть здорового харчування, підходи щодо аналізу даних і звітності харчового раціону з урахуванням персональних даних, роль та особливості створення програмних систем розрахунку калорій із використанням різних архітектурних рішень. У результаті порівняльного аналізу вказаних систем було обрано монолітну архітектуру, на базі якої було спроектовано, реалізовано, протестовано програмну систему підрахунку калорій із використанням середовища розроблення Microsoft Visual Studio 2022; .Net Framework версії 7.0.0; технології ADO.NET; Windows Presentation Foundation (WPF); мови програмування C#; мови розмітки XAML; бази даних Microsoft SQL Server 2019 з метою покращення якості харчування користувачів.

Ключові слова: план харчування, калорійність їжі, дієта, монолітна архітектура, підрахунок калорій.

V. Pleskach, E. Vakulenko, A. Serdiuk

CALORIE CALCULATION SOFTWARE SYSTEM

This article describes the concept and essence of healthy nutrition, approaches to data analysis and reporting of food intake taking into account personal data, the role and features of creating software systems for calculating calories using various architectural solutions. As a result of the comparative analysis of the specified systems, a monolithic architecture was chosen, on the basis of which a software system for counting calories was designed, implemented, and tested using the Microsoft Visual Studio 2022 development environment; .Net Framework version 7.0.0; ADO.NET technologies; Windows Presentation Foundation (WPF); C# programming language; XAML markup languages; Microsoft SQL Server 2019 database in order to improve the quality of user experience.

Key words: nutrition plan, food calories, diet, monolithic architecture, calorie counting.

Вступ

За останні пів століття ожиріння стало серйозною проблемою охорони здоров'я. Відповідно до оцінок експертів Всесвітньої організації охорони здоров'я (ВООЗ) здоров'я громадян на 8-12% залежить від системи охорони здоров'я в країні, на 18-20% від генетичної схильності людини, на 68-74% – від способу життя, однією з найважливіших складових якого є харчування [1]. Відповідно до класифікації ВООЗ людину вважають такою, що страждає від ожиріння, якщо її індекс маси тіла (ІМТ) становить більше 30 кг/м². Тому підрахунок калорій людини є ключовим елементом в реалізації здорового харчування і способу життя. Нині немає універсальної медичної інформаційної технології, яка в режимі реального часу обчислює кількість калорій будь-якої їжі клієнта. Найбільш прагматичним рішенням цієї проблеми є проектування, розроблення та тестування такої

програмної системи підрахунку калорійності продуктів харчування з використанням новітніх технологій, щоб вона мала дружній інтерфейс, і клієнт сам міг коригувати по собі дані, створювати власне меню на день, тиждень, і спостерігати зміну своїх показників у бік покращення здоров'я.

Відомо багато досліджень, які пов'язані з розпізнаванням і ідентифікацією зображень їжі з використанням, наприклад, нейромереж, комп'ютерного зору. Зокрема, це метод, базований на оптимізованих за параметрами нейронних мереж згортки (CNN) для виявлення зображень традиційних страв за допомогою портативної камери [2]. Ці підходи, як правило, пов'язані з глибоким навчанням – наприклад, система DEEPFIC, що використовує глибоке навчання для класифікації харчових продуктів і оцінки їх калорійності за зображеннями, що передбачає попереднє об-

роблення зображень і застосування нейронних мереж CNN для точного розпізнавання їжі та оцінки калорій; мобільні застосунки та веб-системи, що вивчають створення лічильників калорій на мобільних платформах, таких як Android, щоб допомогти користувачам ефективно відстежувати щоденне споживання калорій. Ці системи часто об'єднують бази даних продуктів харчування та інформацію про їхню харчову цінність, що дозволяє користувачам вводити свої страви та отримувати розрахунки спожитих калорій, і алгоритми, такі як YOLOv5 використовують для розпізнавання їжі та оцінки калорій. Метод сегментації та оброблення зображень для розпізнавання частин їжі застосовують у дослідженні [3].

Метою цього дослідження є підвищення ефективності та зручності надання послуг у сфері харчування та здоров'я на основі розробленої програмної системи розрахунку калорійності. Зокрема, було досліджено теоретичні основи побудови програмних систем для розрахунку калорій, здійснено аналіз рішень щодо створення програмних систем розрахунку калорій і поставлено завдання спроектувати, реалізувати та протестувати програмну систему для розрахунку калорій із кращими характеристиками. Об'єктом дослідження є процеси розрахунку калорійності, предметом дослідження – засоби побудови програмної системи розрахунку калорійності. Методи дослідження охоплюють системний аналіз і синтез, узагальнення тощо.

1. Матеріали та методи

Розроблення програмних систем підрахунку калорій зазвичай передбачає використання алгоритмів для оцінки енергетичного вмісту харчових продуктів. Алгоритмізація підрахунку калорій у програмних системах харчування базована на кількох етапах. Спочатку користувач вводить інформацію про спожиті продукти або вибирає їх з бази даних. Система використовує дані про вміст поживних речовин у цих продуктах такі, як білки, вуглеводи, жири, а також енергетичний вміст.

На другому етапі враховують кількість і обсяг порцій, які користувач спожи-

ває, що може охоплювати вимірювання ваги, об'єму, вибір стандартних розмірів порцій. Після введення всіх показників система обчислює загальну кількість калорій для прийому їжі.

Програмне забезпечення для розрахунку калорій може бути реалізовано на різних платформах, мовах програмування та бібліотеках. Розрахунок кількості калорій у харчових продуктах може ґрунтуватися на різних формулах, проте основна ідея полягає у визначенні енергетичного вмісту продукту на основі його хімічного складу. Двома найпопулярнішими формулами є формула Харріса – Бенедикта та Маффліна – Сан Жеора [4].

Хоча обидві формули використовують для розрахунку кількості калорій, їх вирізняють підходи щодо визначення енергетичних потреб організму. Формула *Харріса – Бенедикта* враховує стать, вік, зріст і вагу: жінки: $BMR = 655 + (9,6 * \text{вага в кг}) + (1,8 * \text{зріст у см}) - (4,7 * \text{вік у роках})$; чоловіки: $BMR = 66 + (13,7 * \text{вага в кг}) + (5 * \text{зріст у см}) - (6,8 * \text{вік у роках})$.

Щоб дізнатися свій активний рівень метаболізму, потрібно додатково помножити цю кількість калорій на коефіцієнт, який залежить від рівня активності: 1,2 – малорухливий спосіб життя; 1,375 – легкі фізичні вправи один – три дні на тиждень; 1,55 – помірні фізичні вправи шість – сім днів на тиждень; 1,75 – тяжкі фізичні вправи щодня; 1,9 – важкі фізичні вправи два чи більше разів на день.

Переваги полягають у тому, що формулу легко обчислити та зрозуміти, це пояснює її широке застосування. Серед недоліків можна виділити те, що не враховано персональних показників клієнта та відсутня інформація про рівень активності, що може призвести до неточних результатів.

Формула *Маффліна – Сан Жеора* також враховує стать, вік, зріст і вагу, та відразу використовує показник активності: для жінок: $(10 * \text{вага (кг)} + 6,25 * \text{зріст (см)} - 5 * \text{вік (г)} - 161) * A$; для чоловіків: $(10 * \text{вага (кг)} + 6,25 * \text{зріст (см)} - 5 * \text{вік (г)} + 5) * A$; де A – це рівень активності людини, який зазвичай розрізняють за п'ятьма рівнями фізичних навантажень на добу: мінімальна активність: $A = 1,2$; слабка актив-

ність: $A = 1,375$; середня активність: $A = 1,55$; висока активність: $A = 1,725$; екстра – активність: $A = 1,9$. Труднощі розрахунку та потреба точнішої інформації – її головні недоліки.

У той час як формула Харріса-Бенедикта зазвичай використовується для загальної оцінки, формула Маффліна – Сан Жерора може надати більш персоналізовані результати, особливо у визначенні кількості калорій, потрібних для досягнення конкретної ваги.

За думкою Х. Мітчелл нутрієнти їжі можна розрахувати за допомогою системи Атуотера, основаної на таких формулах: білки: 30% калорійності раціону; жири: 30% калорійності раціону; вуглеводи: 40% калорійності раціону [5].

Систему Атуотера застосовують у харчовій промисловості для визначення загальної калорійності їжі за методом 4 – 9 – 4. Ця система використовує коефіцієнти перетворення енергії на макроелементи вуглеводів, жирів, білків і клітковини. Середні значення енергії подають як кількість калорій на 1 грам макроелемента: 1 грам білків = 4 ккал; 1 грам жирів = 9 ккал; 1 грам вуглеводів = 4 ккал. Ці значення базуються на середньому виході енергії цих поживних речовин під час травлення та метаболізму.

Найпоширеніші нині архітектури програмних систем: клієнт – сервер, монолітна, сервіс-орієнтована, мікросервісна архітектура тощо.

Прикладом застосування для підрахунку калорій, що послуговується клієнт – серверною архітектурою може бути MyFitnessPal. Адже його клієнтська частина взаємодіє з сервером, де знаходиться база даних харчових продуктів та алгоритми підрахунку калорій [6]. Однак навіть такий поширений програмний продукт мав проблеми із безпекою. Близько 150 млн користувачів MyFitnessPal від Under Armour стали жертвою витоку даних в лютому 2018 р., під час якого хакери отримали доступ до імен користувачів, паролів і адрес електронної пошти.

Багатокомпонентна архітектура – це ще один підхід до розроблення програмних систем, базованих на розподілі функціонала на невеликі автономні компоненти.

Кожен компонент може виконувати свою конкретну функцію та взаємодіяти з іншими компонентами для досягнення загальної мети системи. Головними характеристиками багатокомпонентної архітектури є модульність, що полегшує роботу над окремими компонентами, повторне використання коду, а це сприяє ефективному використанню ресурсів і спрощує розроблення системи завдяки незалежності роботи над компонентами. З іншого боку, управління великою кількістю компонентів може стати викликом, особливо в складних системах. Забезпечення взаємодії між компонентами та розв'язання питань сумісності може вимагати додаткових зусиль і ресурсів. Зразком практичного застосування багатокомпонентної архітектури є система Fitbit, де трекер активності, мобільний застосунок і веб – сайт виконують різні функції, а їхні взаємодії утворюють інтегровану інформаційну систему для відстеження здоров'я та фізичної активності. 2019 року Fitbit була п'ятою найбільшою компанією з виробництва технологій для носіння приладів (бездротових пристроїв, моніторів фізичної підготовки та трекерів активності, таких як розумні годинники, крокоміри та монітори пульсу і якості сну) за обсягом постачання. У 2020 року продано понад 120 млн пристроїв Fitbit для 30 млн користувачів зі 100 країн.

Розподілені системи використовують компоненти, що функціонують в інфокомунікаційних мережах і характеризуються децентралізацією, розподіленими взаємодіями між обчислювальними вузлами. Комунікації між обчислювальними вузлами можуть стати джерелом затримок і витрат, а вирішення проблем консистентності стає складнішим завданням. Крім того, забезпечення безпеки в розподілених системах є важливою задачею через велику кількість обчислювальних вузлів і точок доступу. Прикладом є Nutrino – веб-платформа, що використовує розподілені системи для збору та оброблення даних про харчування та надає персоналізовані рекомендації. Мікросервісна архітектура – це новітній підхід до розроблення програмного забезпечення, за якого застосунок розподілено на невеликі автономні сервіси, кожен з яких виконує обмежену кількість функцій. Усі складові системи вза-

ємодіють між собою через API, що дає їм змогу функціонувати незалежно один від одного. Основною ідеєю є розподіл функцій на окремі компоненти, які можна розвивати, масштабувати та випускати окремо. Серед недоліків такої архітектури передусім виділяють децентралізовану структуру, яка може спричиняти труднощі в управлінні комунікаціями та мережею, версіями та оновленнями. Також у разі вибору «мікросервісів» необхідні складніші засоби моніторингу для відстеження стану та продуктивності, якщо порівнювати з монолітною архітектурою. Наприклад, HealthyTrack – це мобільний застосунок для ведення здорового способу життя, який використовує мікросервісну архітектуру для оптимізації різних аспектів підрахунку калорій і здорового харчування.

Монолітна архітектура – традиційний підхід до побудови програмних систем, за якого весь функціонал зосереджено у кодовій базі та виконується в єдиному процесі [7].

Запуск і оновлення системи в монолітній архітектурі вимагає мінімальних зусиль, оскільки всі компоненти знаходяться разом, що робить процес розгортання відносно простим і зрозумілим для розробників. Завдяки централізованій структурі процесу моніторингу та налагодження також стають простішими. Проте з розвитком програмної системи виникають проблеми гнучкості, масштабованості, налагодження. Приміром, система Cronometer – прикладний застосунок для ведення щоденного журналу харчування, де весь функціонал (введення даних, аналіз, статистика) вбудовано в єдиний монолітний застосунок. Цей прикладний застосунок для відстежування макронутрієнтів створено 2005 року директором А. Девідсоном.

На додаток до відстеження їжі, програма уможливує контроль фізичних вправ, що є частиною функціонала безкоштовної версії. Якщо перейти до платної версії Cronometer Gold, з'явиться можливість відстежувати біометричні дані або різні вимірювання тіла та кардіо, а також результати лабораторних досліджень.

У табл.1 наведено порівняльну характеристику монолітної та мікросервісної архітектур.

Таблиця 1

Порівняльна характеристика монолітної та мікросервісної архітектур

Параметр	Монолітна архітектура	Мікросервісна архітектура
Гнучкість	Низька	Висока
Масштабованість	Обмежена	Висока
Керування залежностями	Просте	Складне
Ізоляція та оновлення	Складне	Просте

Джерело : власна розробка

У таблиці 2 наведено порівняльну характеристику вище наведених програмних систем.

Таблиця 2

Порівняльний аналіз систем підрахунку калорій

Функціонал	<i>Cronometer</i>	<i>MyFitnessPal</i>	<i>Fitbit</i>
Підрахунок калорій	Так	Так	Так
Ведення журналу харчування	Так	Так	Так
Створення персоналізованого меню	Ні	Ні	Ні
Персоналізовані рекомендації щодо харчування	Так	Ні	Так
Відстеження фізіологічних показників	Ні	Так	Так
Інтеграція з іншими пристроями	Так	Так	Так
Відстеження настрою	Ні	Ні	Так
Наявність статистики у вигляді графіків	Ні	Так	Так

Джерело : власна розробка

У результаті порівняльного аналізу було досліджено основний функціонал для програмних систем підрахунку калорій і

виявлено, що, хоча усі програмні системи надають багато функцій, їм бракує зрозумілого та зручного дружнього інтерфейсу і таких функцій, як створення тижневого меню, складання списку продуктів з підбраного плану харчування та можливостей в режимі реального часу здійснювати зміни клієнтом до прикладного застосунку.

Зважаючи на вище вказані переваги та недоліки програмних систем із різними архітектурними рішеннями було обрано монолітну архітектуру як задовільний вибір для успішної реалізації програмної системи підрахунку калорій і проведення експериментальної апробації та статистичного аналізу показників клієнтів щодо зменшення ваги.

Програмна система підрахунку калорій має забезпечувати користувачам зручний і ефективний інструмент для контролю раціону та підтримки здорового способу життя. Вона дозволяє реєструватися з введенням персональних даних, визначає норми калорій і допомагає в плануванні харчування. Система також дає можливість слідкувати за харчовими звичками та фізичною активністю, сприяючи досягненню конкретних цілей щодо фізичної форми та здоров'я.

2. Результати

Програмна система з монолітної архітектурою є корисною для тих користувачів, у яких є медична потреба в щоденному відстеженні прийому їжі з урахуванням кількості калорій і макронутрієнтів або бажання дізнатися більше про страви, які вони вживають щодня. Для зручного використання програми розроблено простий, зрозумілий, дружній інтерфейс, що дає можливість користувачу швидко отримати всю необхідну інформацію. Розроблена інформаційна система забезпечує такі можливості: реєстрацію з внесенням таких персональних даних, які допоможуть у складанні унікального меню. Зокрема, це показники росту, ваги, віку, схильності до алергії тощо; забезпечення безпеки збережених даних користувача; вхід до інформаційної системи за допомогою імені користувача та паролю; зміна персональних даних; автоматичне визначення норм КБЖВ користувача;

пошук і додавання страв; формування списку улюблених або рекомендованих страв; відслідковування харчових звичок і фізичної активності за допомогою журналу; отримання таблиці з меню; збереження меню у форматі html, pdf; отримання списку продуктів тощо.

Програмна система має такі функції як-от: реєстрації; авторизації; зміну налаштувань; збереження даних; виведення даних у вигляді діаграм; пошуку та додавання/видалення інформації; автоматичної генерації таблиці на основі фізіологічних показників людини; збереження таблиці та списку; розрахунок кілокалорій, які потрібно споживати людині; автоматичної генерації та виведення списку продуктів, виходячи з їжі, сформованої у попередньо складеному плані харчування. Вимоги до безпеки програмної системи є такі: паролі користувачів мають бути шифрованими за допомогою алгоритму *pwdencrypt* перед збереженням у базі даних, щоб запобігти можливості несанкціонованого доступу до персональних даних; використання *stored procedures* для виконання операцій з даними для обмеження прямого доступу до таблиць і забезпечення доступу для контролю і моніторингу лише через визначені процедури. Вимоги до продуктивності програмної системи підрахунку калорій є такі: забезпечення ефективного використання ресурсів для швидкого обчислення калорій та інших даних, що дозволить користувачам швидко отримувати результати без зайвої затримки; використання запитів до бази даних і кешування результатів для зменшення навантаження на сервер і прискорення роботи програми. Вимоги до зручності програмної системи підрахунку калорій є наступними: забезпечення інтуїтивно зрозумілого та дружнього користувачького інтерфейсу, який сприяє зручності взаємодії з програмою; розроблення інтерфейсу з урахуванням принципів дизайну, таких як простота, послідовність і доступність для забезпечення набутого досвіду користувачів; чіткий розподіл логіки бізнес – процесів, подання та керування даними, щоб забезпечити простоту розроблення та забезпечення функціонування прикладної інформаційної системи.

Було змодельовано та спроектовано функціональні елементи програмної системи підрахунку калорій на основі відповідних UML діаграм. Для моделювання функціональності програмної системи з погляду її взаємодії з користувачем було розроблено діаграму варіантів використання. Діаграма використання цього прикладного застосунку складається з одного актора – користувача та восьми головних прецедентів, що містять залучені прецеденти зв'язком «include» (рис.1).

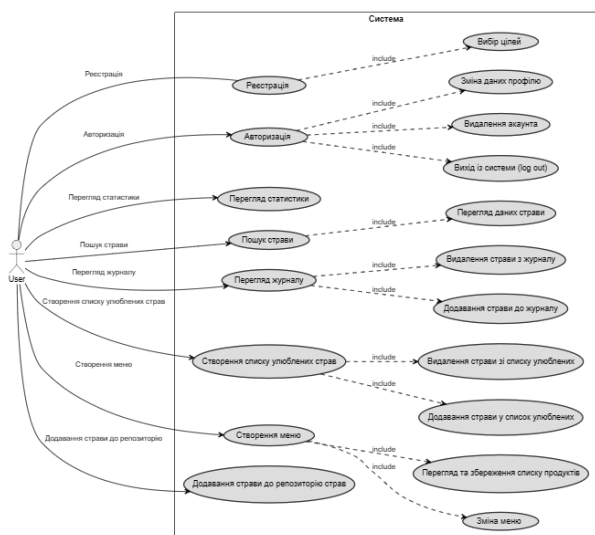


Рис. 1. Діаграма варіантів використання відповідно до функціональних вимог програмної системи підрахунку калорій

Як зазначено вище, монолітна архітектура програмної системи – це традиційна модель розроблення програмного забезпечення, де все програмне забезпечення розробляється як єдиний фрагмент коду. Всі компоненти архітектури взаємозалежні та взаємопов'язані, і кожен компонент необхідний для запуску програми. Перевагами цього підходу є простота розроблення та впровадження, легка масштабованість на початкових етапах, а недоліками – труднощі масштабування, якщо програмна система стає великою, вищі ризики, пов'язані з підтримкою та модифікацією. Насамкінець, це може призвести до збільшення часу розгортання, труднощів у впровадженні нових функцій та ускладнення тестування через велику взаємозалежність компонентів.

На рис.2 зображено архітектуру програмної системи підрахунку калорій.

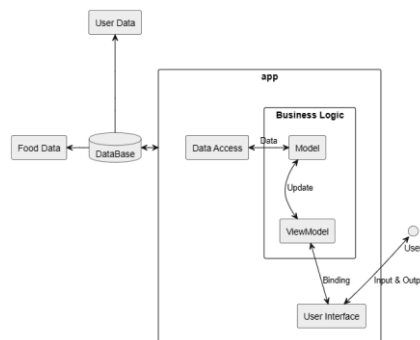


Рис. 2. Архітектура програмної системи підрахунку калорій

Для зменшення складності програмного коду використовувався архітектурний шаблон MVVM (Model – View – ViewModel). У процесі створення програмної системи підрахунку калорій було застосовано: середовище розроблення Microsoft Visual Studio 2022; фреймворк .Net Framework версії 7.0.0; методику ADO.NET; технологію Windows Presentation Foundation (WPF); мову програмування C#; мову розмітки XAML; базу даних Microsoft SQL Server 2019, Microsoft SQL Server Management Studio 19.

Для реалізації інформаційної системи підрахунку калорій було створено прикладний застосунок для ОС Windows. Організацію архітектури системи показано на рис. 3.

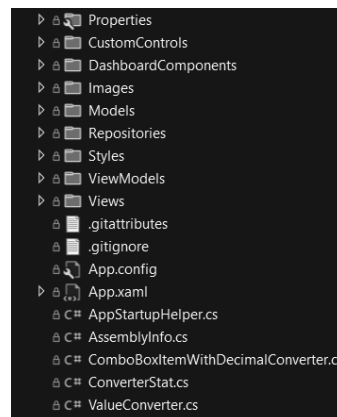


Рис. 3. Організація архітектури системи

Через те, що шаблон MVVM передбачає структуру View – ViewModel – Model, програмну систему було розподілено на відповідні категорії, що можна представити у вигляді каталогів проекту. Всі сторінки та вікна мають xaml – файли у теці Views, а класи, що відповідають за логіку у теці ViewModels. Для представлення об'єктів даних є моделі користувача та страви у каталозі Models, а також інтерфейси для роботи

з базою даних у теці Repositories. Стили візуальних елементів програми, такі як кольори та розмір, знаходяться у теці Styles. Прикладний застосунок містить два вікна – авторизації, де необхідно увійти до інформаційної системи чи зареєструватись, і головне вікно, де відображають статистику, страви та налаштування профілю клієнта. За перемикання сторінок головного вікна відповідає модель подання, а за відображення – вид – обгортка. Застосунок складається з таких елементів, як вікно авторизації: вхід; реєстрація; головне вікно: сторінка статистики, сторінка пошуку страв, сторінка додавання/видалення страв, сторінка улюблених (рекомендованих) страв, сторінка меню, сторінка налаштувань. Сторінка статистики є обгорткою для дочірніх елементів, що розміщуються у теці Dashboard Components.

База даних системи містить тринадцять таблиць, кожна з яких має первинний ключ. Таблиці поділяють на дві основні групи: ті, що належать до даних користувача, та ті, що зберігають інформацію про страви. До групи користувачів відносять три таблиці: «Users», «Activity», «UserAllergy». Таблиця «UserAllergy» містить два зовнішніх ключа та поєднує користувача з алергеном з таблиці «Allergens», що у свою чергу пов'язана з таблицею «ProductAllergens», яка містить інформацію про алергенні продукти. Відношення «UserAllergy» встановлює відповідність між ім'ям користувача та типом алергії, а «ProductAllergens» між назвою продукту зі списку продуктів і назвою складових їжі. Таким чином пов'язується інформація щодо алергії з метою для подальшого складання плану харчування на основі збережених даних. Група стосовно страв нараховує п'ять таблиць, а саме: «Products», «Meals», «MealNutrition», «MealProducts», «Product Allergens», зв'язаних між собою ID страви. Структуру бази даних представлено на рис. 4.

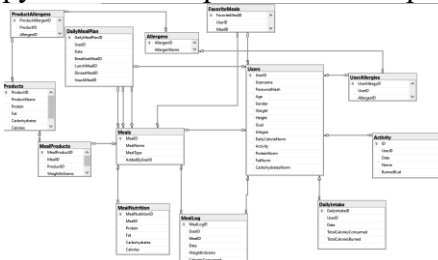


Рис. 4. Схема бази даних програмної системи підрахунку калорій

Для того, щоб працювати з програмною системою, користувачу треба увійти до неї, тому при завантаженні викликається вікно аутентифікації, що отримує значення логіну і пароля та передає їх до моделі подання, яка в свою чергу застосовує репозиторій користувачів, який підключено до бази даних, для перевірки існування користувача із заданими показниками. Якщо такого клієнта немає, то з'являється відповідне повідомлення, після чого можна повторити спробу входу. Якщо ж користувач з вказаними логіном і паролем існує, залежно від введених параметрів викликається інтерфейс, реалізований у головному вікні програми, і закривається форма аутентифікації. Сторінка з назвою «Registration» відкривається у вікні «Login». Сторінка реєстрації користувача містить 10 полів для заповнення даних користувача, кнопку підтвердження реєстрації та посилання на сторінку авторизації. Поля передбачають введення такої інформації: ім'я користувача; пароль; вік; зріст; вага; стать; алергія (інші захворювання); вегетаріанство; активність; ціль. Усі поля, окрім «вегетаріанства», є обов'язковими для заповнення.

Сторінка статистики – одна з основних, бо відображає кілька різних статистичних елементів для полегшення розуміння споживання калорій користувачем. Діаграма відображає відсоткове співвідношення спожитих калорій до денної норми. Інформація праворуч від діаграми вказує на спожиті білки, жири та вуглеводи, а також їх норму. Дані посередині сторінки показують кількість спожитих і спалених кілокалорій. А нижня стовпчикова діаграма дає зрозуміти тенденцію споживання калорій за останні 7 записів. Уся інформація оновлюється автоматично після відповідних внесених змін до бази даних. У програмі передбачено можливість користувачу зробити свій внесок у розширення бази даних страв. Для цього на відповідній сторінці «Contribute» є форма для додавання страви, що містить поле назви, спадаючий список типу страви, прапорець для визначення, чи є ця стравка вегетаріанською, «контейнер» для додавання продуктів. У програмній системі також є можливість зберігати страви до списку улюблених. На сторінці налашту-

вань користувач має змогу змінити окремо такі показники, як вік, зріст, вага, ціль, рівень активності, а також видалити свій акаунт. У разі видалення, усі пов'язані із користувачем записи зникнуть з бази даних. Для перевірки працездатності програмної системи було обрано два методи: автоматизоване модульне тестування та ручне тестування різноманітних сценаріїв за методом «чорної скриньки». Особливу увагу приділено тестуванню поведінки ViewModels, які відповідають за логіку застосунка і зв'язують користувацький інтерфейс з даними. Це тестування проводилось за допомогою NUnit Test та ReSharper. Науковий результатом дослідження є також оптимізація дієти, передусім розроблення алгоритму для покращення дієти на основі потреб у калоріях і властивостей харчування для забезпечення здорового способу життя. Подальші наукові розвідки цього дослідження спрямовані на підтримку мовних і географічних відмінностей із можливістю адаптації програмної системи до різних мов і кулінарних традицій різних регіонів світу.

Список літератури

1. Принципи раціонального здорового харчування. Accessed: 10.04.2024. <http://surl.li/smryp>.
2. R. Haque, R. Nhan, N. A. Shihavuddin, M. Mahbubul Syeed, M. Faisal Uddin, Lightweight and Parameter-Optimized Real-Time Food Calorie Estimation from Images Using CNN-Based Approach, in: Appl. Sci. 2022, 12, 9733. <https://doi.org/10.3390/app12199733> (дата звернення 08.04.2024).
3. R. Jayasingh .J., B. Nagajayanthi, M. Neebha, M. Sagayam, A.A. Elngar, New calculation of calorie content and determining nutritional level from day-to-day intake of food using Image Processing, in: Research Square. 2022. DOI: <https://doi.org/10.21203/rs.3.rs-1766525/v1>
4. Як розрахувати свій базовий метаболізм: формула Харріса – Бенедикта. Apollo Online. URL:

<https://apollo.online/blog-post/yak-rozrahuvaty-svij-bazovyj-metabolizm-formula-harrisa-benedykta/> (дата звернення: 23.01.2024).

5. H. Mitchell, Food Analysis, Springer Science & Business Media, 2012. 500 p.
6. The Best Fitness Trackers and Watches for Everyone. Accessed: 18.01.2024. <https://www.wired.com/gallery/best-fitness-tracker/>
7. QALight. Монолітна архітектура ПЗ. Accessed: 18.01.2024. <https://qalight.ua/baza-znaniy/shho-take-monolitna-arhitektura/>

Одержано: 11.04.2024

Внутрішня рецензія отримана: 20.04.2024

Зовнішня рецензія отримана: 22.04.2024

Про авторів :

¹Плескач Валентина ,
Доктор економічних наук,
кандидат технічних наук,
професор
<https://orcid.org/0000-0003-0552-0972>.

¹Вакуленко Євгенія
Студентка
<https://orcid.org/0009-0002-4256-4464>

²Сердюк Анатолій ,
Кандидат технічних наук,
доцент
<https://orcid.org/0000-0002-0787-3889>.

Місце роботи авторів:

¹ФІТ КНУ імені Т.Шевченка

²Faculty of Power and
Aeronautical Engineering,
University of Technology,
00-665 Warsaw, Nowowiejska 24,
Poland
Anatolii.Serdiuk@pw.edu.pl