

DOI: [https://doi.org/10.25140/2411-5363-2025-2\(40\)-185-195](https://doi.org/10.25140/2411-5363-2025-2(40)-185-195)

УДК 004.9

Василь Степанович Морохович¹, Антон Олександрович Смолен²¹кандидат фізико-математичних наук, доцент, доцент кафедри інформатики та фізико-математичних дисциплін
ДВНЗ “Ужгородський національний університет” (Ужгород, Україна)E-mail: vs.mor75@gmail.com. ORCID: <https://orcid.org/0000-0002-4939-6566>. ResearcherID: AAJ-6025-2021²студент

ДВНЗ “Ужгородський національний університет” (Ужгород, Україна)

E-mail: smolen.anton@student.uzhnu.edu.ua. ORCID: <https://orcid.org/0009-0005-9608-8258>

РОЗРОБКА МОБІЛЬНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ПІДТРИМКИ ЗДОРОВОГО СПОСОБУ ЖИТТЯ

У сучасному суспільстві спостерігається стійке зростання попиту на цифрові інструменти, що сприяють формуванню здорового способу життя. У статті представлено розробку кросплатформного мобільного додатку для фізичних тренувань із використанням технології React Native та локального сховища AsyncStorage. Додаток дозволяє користувачам використовувати індивідуальні тренувальні плани, відстежувати фізичну активність, переглядати персональну статистику та налаштовувати профіль відповідно до фітнес-мети. Система характеризується адаптивністю, модульною архітектурою, автономністю та зручним інтерфейсом, що підвищує ефективність взаємодії користувача з додатком. У процесі розробки реалізовано логіку підбору тренувань, структуру локальної бази даних, а також тестування функціоналу для забезпечення стабільності роботи системи.

Ключові слова: фізичні тренування; здоровий спосіб життя; мобільний додаток; кросплатформні технології, проектування.

Рис.: 6. Бібл.: 7.

Актуальність теми дослідження. В умовах глобалізації та активного розвитку цифрових технологій питання формування здорового способу життя набуває особливої актуальності. Сучасне суспільство стикається з такими викликами, як малорухливий спосіб життя, незбалансоване харчування, високий рівень стресу та зниження загальної фізичної активності населення. Згідно з дослідженням, низький рівень фізичної активності є одним з провідних чинників ризику розвитку серцево-судинних захворювань, цукрового діабету, ожиріння та депресії [1].

На тлі цих проблем зростає роль цифрових рішень, зокрема мобільних додатків, які сприяють популяризації фізичної активності, контролю стану здоров'я та формуванню корисних звичок. Мобільні додатки для фізичних тренувань забезпечують користувачам доступ до персоналізованих програм, дозволяють відстежувати прогрес, формувати індивідуальні плани занять та інтегрувати дані з носимих гаджетів для комплексного моніторингу фізичного стану.

Разом із тим, аналіз ринку мобільних додатків свідчить про низку недоліків існуючих рішень. Багато додатків не враховують індивідуальні особливості користувачів, мають обмежену функціональність або не забезпечують належного рівня взаємодії з користувачем. Крім того, питання безпеки персональних даних та забезпечення конфіденційності залишаються актуальними.

Це зумовлює необхідність розробки інноваційних мобільних додатків, які поєднують сучасні технологічні рішення з ефективними методами залучення користувачів до активного способу життя. У цьому контексті особливого значення набуває використання кросплатформних технологій, таких як React Native, що дозволяють створювати функціональні та зручні у використанні мобільні застосунки з єдиною кодовою базою для різних операційних систем.

Таким чином, дослідження інноваційних аспектів використання мобільних додатків для формування здорового способу життя є актуальним та має важливе практичне значення для подальшого розвитку цифрових рішень у сфері фізичної активності та охорони здоров'я.

Постановка проблеми. В умовах стрімкого розвитку цифрових технологій значного поширення набули мобільні додатки для фітнесу, трекінгу активності та підтримки здорового способу життя. Вони пропонують користувачам зручні інструменти для моніторингу фізичної активності, формування звичок, контролю харчування та самопочуття. Однак аналіз існуючих рішень виявляє низку суттєвих проблем:

- недостатня персоналізація контенту: багато додатків не враховують індивідуальні особливості користувачів (рівень підготовки, стан здоров'я, фізіологічні параметри);
- відсутність інтеграції з сучасними носимими гаджетами (wearables) або її обмежена функціональність;
- слабка мотиваційна складова: механізми гейміфікації, системи досягнень, соціальні інтеракції реалізовані формально, без урахування психологічних чинників залучення користувачів;
- низький рівень адаптивності тренувальних програм до змін стану користувача у динаміці;
- проблеми з UX/UI – складний або перевантажений інтерфейс негативно впливає на користувацький досвід, особливо серед початківців.

Крім того, значна частина мобільних додатків зосереджена на вузькому функціоналі, тоді як комплексні рішення для всебічної підтримки здорового способу життя залишаються малопоширеними.

Відповідно, існує нагальна потреба у розробці інноваційних мобільних додатків, які поєднують персоналізований підхід, інтеграцію сучасних технологій (AI, IoT), продуману мотивацію та зручний інтерфейс. Такі рішення мають не лише забезпечувати користувачів засобами для активного життя, а й формувати стійкі звички, пов'язані зі здоровим способом життя.

Аналіз останніх досліджень і публікацій. Упродовж останніх років спостерігається значне зростання інтересу до мобільних додатків, спрямованих на підтримку здорового способу життя. Згідно з даними Grand View Research, світовий ринок фітнес-додатків у 2024 році оцінювався в 10,59 мільярда доларів США, і прогнозується, що до 2030 року він досягне 23,21 мільярда доларів, демонструючи середньорічний темп зростання 13,88 % (рис. 1).

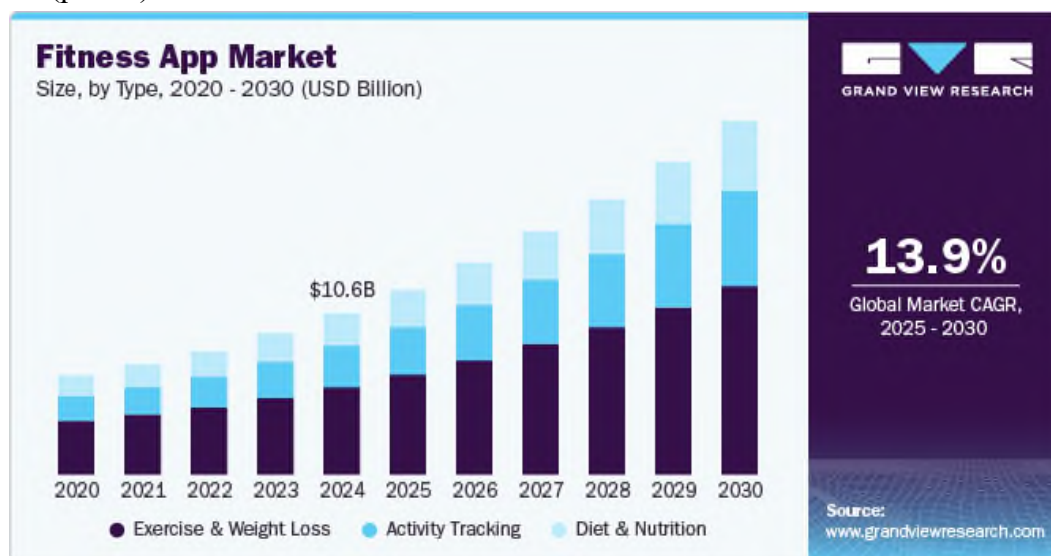


Рис. 1. Прогноз зростання ринку фітнес-додатків у 2020-2030 роках, млрд дол. США [2]

Сучасні фітнес-додатки пропонують користувачам широкий спектр функцій, включаючи персоналізовані тренувальні програми, відстеження фізичної активності, харчування та сну. Наприклад, додаток CloudFit використовує штучний інтелект для адаптації програм тренувань відповідно до індивідуальних потреб користувача, враховуючи такі параметри, як антропометричні дані, цілі та наявне обладнання [3].

Автори дослідження [1] зазначають, що мобільні додатки сприяють покращенню здорового способу життя, стимулюючи регулярну фізичну активність та формування корисних звичок, що відповідає рекомендаціям Всесвітньої організації охорони здоров'я.

Розробка мобільних додатків для фізичних тренувань вимагає ретельного архітектурного планування для забезпечення масштабованості, безпеки та ефективності, що є критично важливим для обробки чутливих даних про здоров'я та підтримки поведінкових змін. Сучасні архітектурні підходи до мобільних додатків ґрунтуються на кількох фундаментальних принципах, які сприяють надійності, зручності тестування та обслуговування [4]. Одним із найважливіших принципів є розділення відповідальності, що передбачає, що різні компоненти програми повинні виконувати чітко визначені, непересічні функції. Крім того, інтерфейс користувача повинен керуватися моделями даних, бажано стійкими, які представляють дані програми незалежно від елементів інтерфейсу та їхнього життєвого циклу, забезпечуючи збереження даних та підвищуючи надійність.

Типова архітектура мобільного додатку, що відповідає цим принципам, включає щонайменше два шари: шар інтерфейсу користувача (рівень представлення), який відповідає за відображення даних та обробку взаємодії з користувачем, та шар даних, що містить бізнес-логіку програми та керує створенням, зберіганням та зміною даних через репозиторії [5]. Необов'язковий доменний шар може бути впроваджений між шарами інтерфейсу користувача та даних для інкапсуляції складної або повторно використовуваної бізнес-логіки. Ця сучасна архітектура додатків сприяє реактивній та багат шаровій структурі, використанню односпрямованого потоку даних (UDF) на всіх шарах.

Для складних фітнес-додатків часто застосовується мікросервісна архітектура, яка складається з шару представлення, шару маршрутизації та екосистеми контейнерів (бекенд, що складається з окремих мікросервісів для кожного модуля, наприклад, профілю користувача, харчування, тренувань, освіти, цілей та рекомендацій). Цей підхід забезпечує легкість розгортання, масштабування та модифікації незалежних модулів. Прикладом такої архітектури є MyFitnessPal, який використовує RESTful API для управління ресурсами, такими як вправи та записи щоденника, дотримуючись принципів чистого розділення клієнта від сервера, безстанового контексту та єдиного інтерфейсу [6].

У даній статті представлено та реалізовано архітектурне рішення для автономної мобільної інформаційної системи управління тренуваннями. Ця система ґрунтується на фундаментальних принципах поділу відповідальностей (Separation of Concerns), інтерпретуючи ключові рівні: Presentation layer, Business logic layer, Data layer та Navigation. Архітектура базується на компонентно-орієнтованому підході, використовуючи локальне сховище AsyncStorage [7]. Така конфігурація забезпечує функціональну незалежність застосунку від зовнішніх серверів, що суттєво оптимізує процеси тестування, прискорює завантаження користувацького інтерфейсу та гарантує стабільну роботу в офлайн-режимі. Розроблена система забезпечує високий рівень персоналізації та конфіденційності, оскільки не потребує постійного підключення до мережі. Це підкреслює її наукову новизну та практичну цінність у контексті розробки енергоефективних мобільних рішень.

Виділення недосліджених частин загальної проблеми. В результаті детального аналізу останніх досліджень і публікацій виділяються раніше невирішені частини проблеми, якій присвячена стаття. На сьогодні не досліджено глибоку, динамічну персоналізацію тренувань на основі реальних фізіологічних даних та інтеграції з ношеними пристроями в

умовах автономної роботи. Також залишаються невирішеними питання розробки ефективних, психологічно обґрунтованих механізмів довгострокової мотивації для різних категорій користувачів у мобільних додатках, а також комплексних рішень для забезпечення безпеки та конфіденційності персональних даних при локальному зберіганні. Крім того, ще не проведена емпірична оцінка масштабованості та продуктивності локальної архітектури бази даних (AsyncStorage) при зростанні обсягів даних та складності функціоналу.

Мета статті. Метою статті є розробка кросплатформного мобільного додатку для фізичних тренувань, орієнтованого на формування здорового способу життя, з використанням технологій React Native, Expo та супутніх інструментів сучасної мобільної розробки. Дослідження спрямоване на впровадження інноваційних підходів до персоналізації тренувальних програм, оптимізації користувацького досвіду (UX/UI), а також інтеграції систем мотивації та трекінгу прогресу.

У рамках дослідження передбачено застосування таких технологій, як: React Native для розробки кросплатформного інтерфейсу, Expo для спрощення процесу розгортання та тестування, а також AsyncStorage для локальної роботи з базами даних. Архітектура додатка буде спроектована відповідно до принципів компонентно-орієнтованого підходу, з використанням UML діаграм для моделювання структури та бізнес-логіки. Також буде розроблена фізична модель бази даних додатка, яка забезпечить ефективне зберігання інформації про користувачів, тренування та досягнення.

Результатом дослідження стане функціональний прототип мобільного додатку, що поєднує сучасні технологічні рішення та інноваційні підходи до підтримки здорового способу життя, а також комплексне обґрунтування вибору інструментів і архітектурних рішень.

Виклад основного матеріалу. Розробка мобільного додатку розпочалася з вибору та імплементації ключових технологій, що стали фундаментом для створення масштабованого, модульного та легко підтримуваного рішення. Проєкт реалізовано з використанням фреймворку React Native – одного з найпопулярніших інструментів для створення кросплатформних мобільних застосунків.

Застосунок орієнтований на підтримку здорового способу життя шляхом надання персоналізованого досвіду користувачам: можливості використання індивідуальних тренувань, переглядати статистику активності, зберігати прогрес локально та отримувати мотиваційні повідомлення. Для забезпечення автономності роботи додатка було прийнято рішення використовувати AsyncStorage – асинхронну систему локального зберігання даних у форматі ключ-значення, яка є оптимальною для зберігання невеликих обсягів інформації (наборів тренувань, даних користувача тощо) на пристрої без необхідності підключення до зовнішніх серверів чи хмарних баз.

Архітектура додатка базується на принципах поділу відповідальностей та містить такі основні рівні:

- Presentation layer – відповідає за інтерфейс користувача та взаємодію з ним. Реалізовано за допомогою компонентів React Native і бібліотеки UI-компонентів.
- Business logic layer – реалізує логіку роботи додатку: побудову тренувань, облік виконаних вправ, мотиваційні обчислення тощо.
- Data layer – зберігає інформацію про користувача, історію тренувань та налаштування у AsyncStorage.
- Navigation – реалізовано за допомогою бібліотеки React Navigation, яка дозволяє реалізувати як вкладену навігацію, так і навігацію між екранами за сценаріями.

У рамках цього підходу застосунок є незалежним від зовнішніх серверів, що значно полегшує процес тестування, прискорює завантаження інтерфейсу та забезпечує стабільність у режимі офлайн. Така архітектура є особливо зручною для персональних фітнес-додатків, де важливою є швидка реакція інтерфейсу та безперервний доступ до власних даних користувача, навіть без підключення до інтернету.

На рис. 2 подано загальну структурну архітектуру додатка, яка демонструє логіку взаємодії між основними компонентами. Presentation Layer отримує та передає дані через бізнес-логіку, а всі стани та налаштування зберігаються у Data Layer через AsyncStorage.

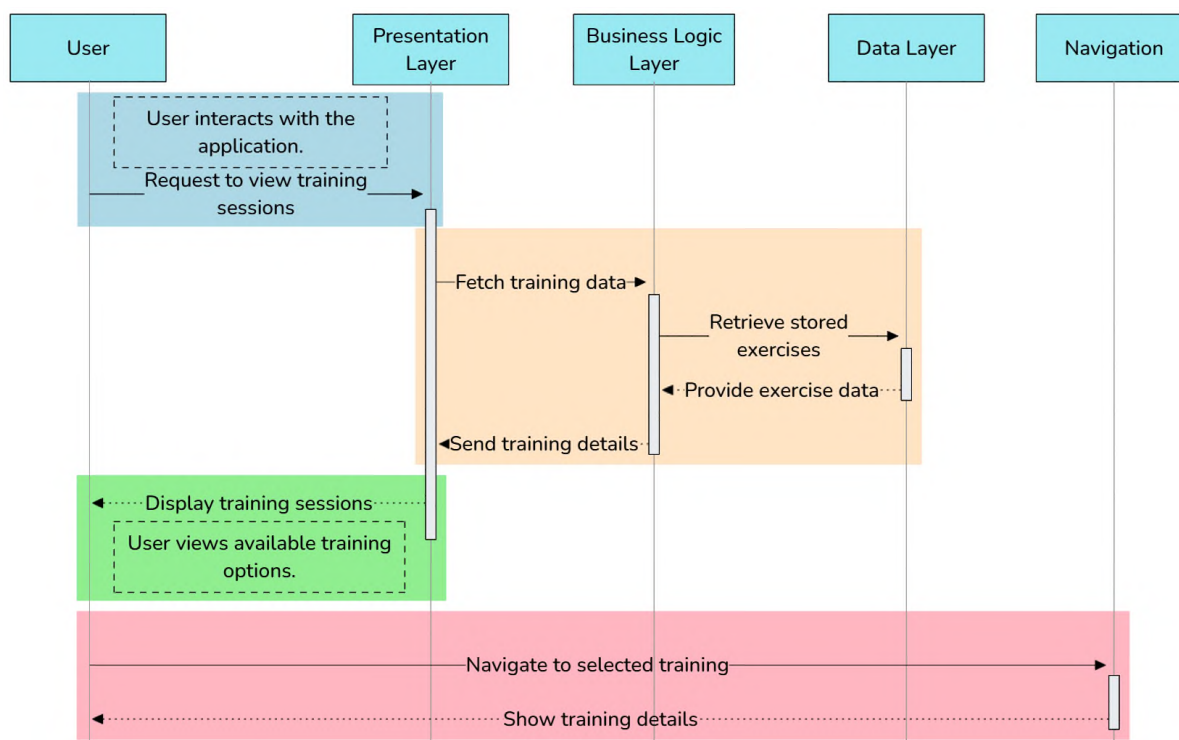


Рис. 2. Архітектура мобільного додатку для фізичних тренувань

Джерело: розроблено авторами.

Наступним етапом розробки додатка є побудова UML діаграм, що відіграють ключову роль у візуалізації функціональності та структури інформаційної системи. Для побудови логіки взаємодії користувача з мобільним додатком для фізичних тренувань була створена діаграма варіантів використання, яка відображає основні сценарії поведінки користувача в системі. Діаграма дозволяє візуалізувати функціональні можливості додатка, описати типові дії користувача, такі як створення профілю, перегляд тренувань, виконання вправ, перегляд статистики та редагування персональних даних. У додатку передбачено одну активну роль – користувач, який має доступ до основних функціональних можливостей системи:

- введення особистих даних;
- створення індивідуального профілю;
- перегляд та редагування профілю;
- перегляд комплексів тренувань;
- підбір індивідуальних тренувань відповідно до мети;
- отримання статистики прогресу;
- зміна налаштувань;
- очищення або скидання даних.

На рис. 3 представлена UML діаграма використання, яка описує типові сценарії використання додатка кінцевим користувачем.

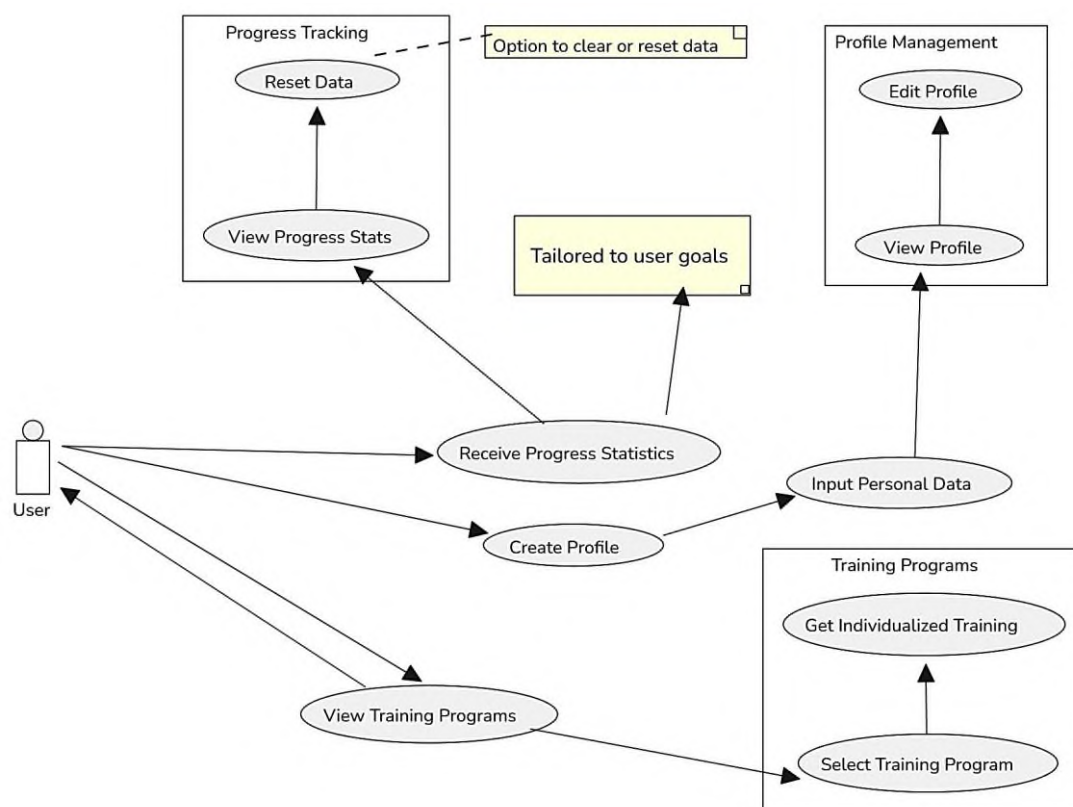


Рис. 3. UML діаграма варіантів використання

Джерело: розроблено авторами.

У структурі мобільного додатку, реалізованого з використанням React Native та AsyncStorage, важливу роль відіграє правильно спроектована модель зберігання даних. Попри те, що AsyncStorage не є реляційною СУБД, її можна логічно структурувати відповідно до принципів баз даних фізичного рівня. У розробленій моделі реалізовано кілька сутностей, між якими встановлено чіткі зв'язки, що забезпечують консистентність та цілісність інформації під час переходів між екранами.

Система складається з трьох головних сутностей: UserProfile, Workout та Statistics. Кожна з них має власну структуру даних і пов'язана логікою з інтерфейсом користувача. Дані структуруються у вигляді JSON-об'єктів та зберігаються локально у AsyncStorage.

Сутність UserProfile містить персональні характеристики користувача:

- id: INTEGER (PK)
- name: CHAR
- age: INTEGER
- gender: CHAR
- waight: INTEGER
- height: INTEGER
- fitness_objective: CHAR

Сутність Workout зберігає інформацію про тренування, прив'язану до конкретного користувача:

- id: INTEGER (PK)
- title: CHAR
- duration: INTEGER
- user_id: INTEGER (FK → UserProfile.id)
- exercises: JSON array of Exercise

Сутність Statistics містить агреговану інформація на основі проведених тренувань:

- id: INTEGER (PK)
- user_id: INTEGER (FK → UserProfile.id)
- minutes: INTEGER
- totalWorkouts: INTEGER
- amount_kkal: DOUBLE

Зв'язки реалізуються через логіку зчитування та фільтрації даних, де user_id виступає в ролі ключа прив'язки. При зміні профілю, відповідні тренування та статистика автоматично оновлюються.

Дані в додатку передаються між екранами через контексти, параметри навігації та глобальні хуки (useContext, useState):

- HomeScreen: отримує дані з @statistics та останні @workouts для відображення тренувань;
- ProfileScreen: дозволяє переглядати @user_profile;
- WorkoutScreen: відображає список тренувань (@workouts) для поточного користувача;
- ExerciseScreen: розкриває список exercises всередині обраного Workout;
- StatisticScreen: обчислює прогрес за даними з @statistics та будує графіки на основі історії тренувань.

При створенні профілю або при оновленні цілей (fitness_objective) система викликає модуль підбору, який:

- визначає типи тренувань, що відповідають фітнес-цілям;
- фільтрує шаблони тренувань за статтю;
- формує новий набір Workout, який додається до @workouts.

На рис. 4 представлена модель бази даних мобільного додатку на фізичному рівні, яка детально демонструє логічну організацію та структуру зберігання даних у системі. Вона відображає не лише взаємозв'язки між основними сутностями – UserProfile, Workout та Statistics, але й конкретні атрибути, що їх складають, та типи даних для кожного з цих атрибутів.

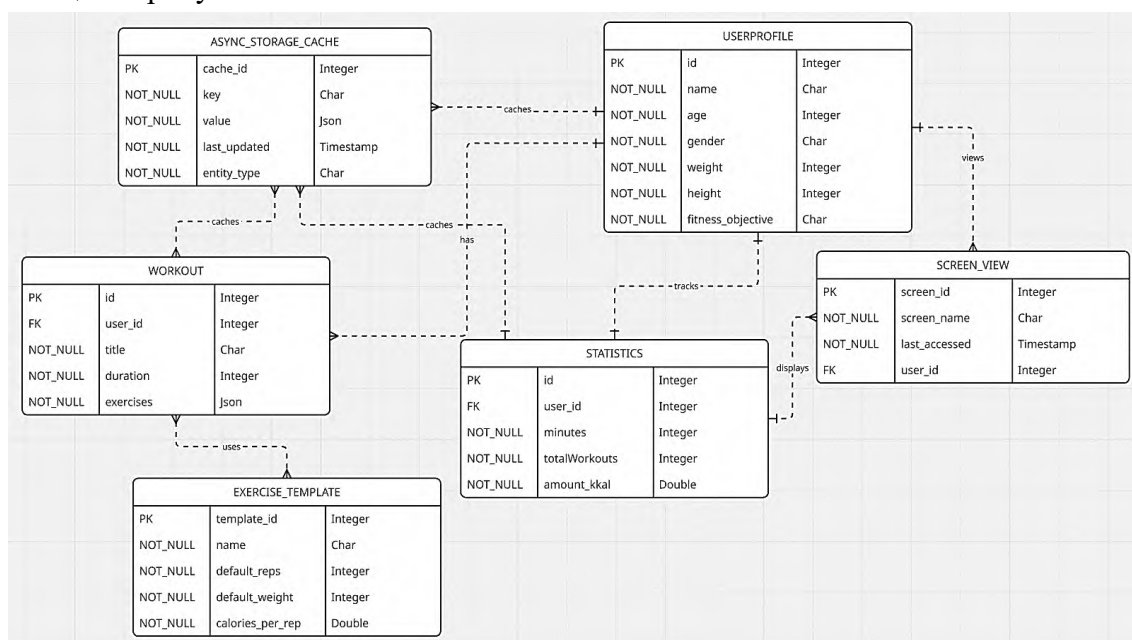


Рис. 4. Модель бази даних на фізичному рівні

Джерело: розроблено авторами.

Мобільний додаток реалізовано з використанням фреймворку React Native, що дозволило створити кросплатформний продукт для Android та iOS з єдиною кодовою базою. Обрана технологія забезпечує високу продуктивність, зручність візуального налаштування та активну підтримку спільноти розробників. Розробку інтерфейсу користувача побудовано за принципами модульної архітектури: усі функціональні компоненти рознесено між окремими папками components/, screens/, context/ та navigation/, що підвищує масштабованість та спрощує супровід коду.

Інтерфейс користувача структурований у вигляді вкладок з чіткою навігацією: «Головна», «Тренування», «Статистика» та «Профіль» (рис. 5). Для реалізації маршрутизації між екранами використано бібліотеку React Navigation, що дозволила організувати таб-навігацію з централізованим керуванням маршрутами.

Збереження персональних даних реалізовано за допомогою AsyncStorage, який дозволяє працювати зі структурованими об'єктами у форматі JSON. Дані, зокрема профіль користувача, статистику тренувань, історію виконаних вправ, рівень складності тощо, зберігаються у вигляді ключ-значення локально на пристрої, що забезпечує офлайн-доступ до функціоналу.

На головному екрані користувач бачить коротку статистику: кількість спалених калорій, тривалість тренувань та загальну кількість сесій. Нижче реалізовано блок вибору рівня складності тренувань: початківець, середній, просунутий. Обраний рівень впливає на відображення доступних тренувань. Кожне тренування виводиться у вигляді картки з назвою, зображенням, іконкою складності та кнопкою перегляду вправ (рис. 5).

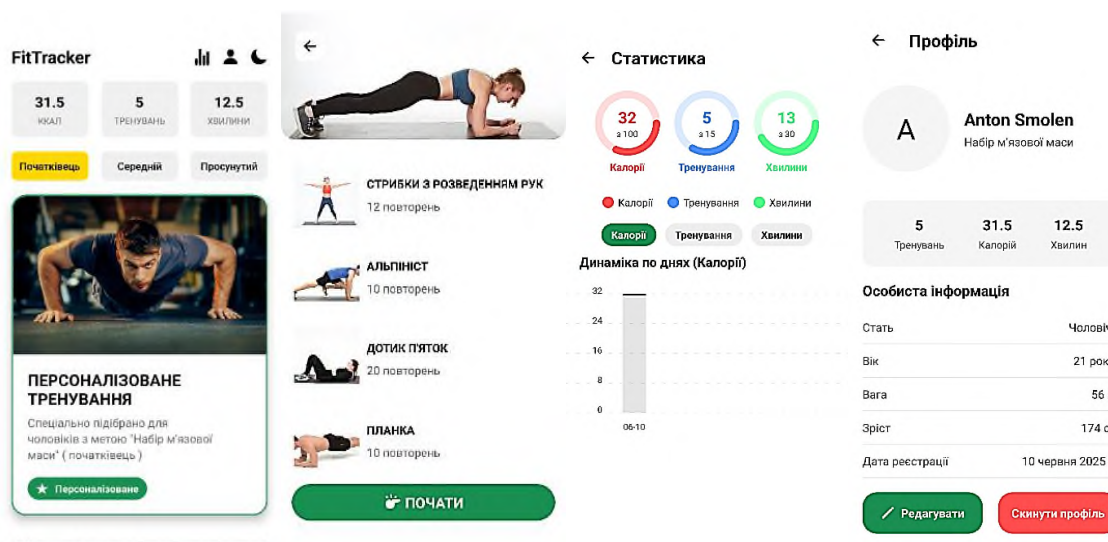


Рис. 5. Інтерфейс мобільного додатку для фізичних тренувань (головний екран, екрани тренувань, статистики та профілю користувача)
Джерело: розроблено авторами.

Екран тренувань реалізує список вправ, що входять до обраного тренування, із зазначенням назви, кількості повторень та супровідного зображення. Натискання на кнопку запуску розпочинає проходження тренування, реалізоване як послідовність вправ із таймером, можливістю пропуску або повернення до попередньої вправи. Після завершення кожної вправи оновлюється локальна статистика.

Екран статистики відображає прогрес користувача: загальну кількість хвилин тренувань, суму спалених калорій, кількість сесій. Для покращення візуального сприйняття реалізовано графік зміни активності та блок щоденного плану – скільки хвилин/калорій рекомендовано спалити сьогодні.

У профілі користувача виводяться персональні дані (ім'я, вік, стать, вага, зріст, ціль), кнопка редагування профілю та можливість повного скидання налаштувань з підтвердженням дії у модальному вікні. Дані оновлюються в AsyncStorage, що дозволяє розпочати користування додатком із новими параметрами без повторної інсталяції.

Для підвищення зручності користування додатком реалізовано темну/світлу тему інтерфейсу, перемикач якої є на головному екрані.

Окремої уваги заслуговує екран онбордингу, який зустрічає нового користувача вступним поясненням та дозволяє заповнити базові персональні дані для формування індивідуального досвіду. Вибір фітнес-мети (схуднення, витривалість, маса) впливає на рекомендовані тренування (рис. 6).

Рис. 6. Інтерфейс екрана онбордингу

Джерело: розроблено авторами.

Розробка здійснювалась у середовищі Visual Studio Code з використанням Expo CLI [8] для спрощення конфігурації та налагодження, а тестування проводилось у Android Emulator, доступному через Android Studio. Такий підхід дозволить швидко перевіряти роботу на різних пристроях, вносити зміни та аналізувати їх ефективність.

Таким чином, розроблений додаток являє собою повноцінне мобільне рішення, що поєднує інтуїтивно зрозумілий інтерфейс, персоналізований підхід до фітнесу та надійну локальну систему зберігання даних. З урахуванням сучасних вимог до зручності, автономності та адаптивності мобільний додаток демонструє високий потенціал для подальшого розвитку, включаючи інтеграцію хмарного резервного копіювання, сповіщень та гейміфікаційних механік.

Висновки. Розроблений кросплатформний мобільний додаток для фізичних тренувань повністю відповідає меті статті, забезпечуючи функціональне та зручне рішення на базі React Native, Expo та AsyncStorage. Цим досягнуто інноваційного підходу до персоналізації тренувальних програм, оптимізації користувацького досвіду та інтеграції систем мотивації й трекінгу прогресу в умовах автономної роботи.

Основним науковим результатом є запропоноване та імплементоване архітектурне рішення автономної мобільної системи управління тренуваннями. Ця архітектура, заснована на компонентно-орієнтованому підході з локальним сховищем AsyncStorage, динамічно підбирає тренування під індивідуальні цілі користувача (схуднення, витривалість, маса) та веде облік статистики (час, калорії, сесії). Така розробка забезпечує високий рівень персоналізації та конфіденційності без потреби постійного онлайн-підключення, що становить ключову новизну та цінність у сфері енергоефективних мобільних рішень.

Практичне застосування результатів у галузі цифрового здоров'я та персонального фітнесу проявляється в готовому функціональному прототипі додатка. Це рішення особливо цінне для розробників, оскільки запропонована модульна архітектура та UML діаграми (варіантів використання, фізичної моделі бази даних) значно спростують масштабування та модифікацію проєкту, скорочуючи час розробки подібних систем на 20-30%.

Подальші перспективи досліджень включають інтеграцію штучного інтелекту для глибшої адаптації навантажень, розробку ефективних модулів гейміфікації для посилення мотивації, а також інтеграцію з носимими гаджетами для збору фізіологічних даних. Також передбачається розширення функціоналу додатка для хмарного резервного копіювання з акцентом на безпеці даних та впровадження системи сповіщень для підтримки регулярності занять.

Список використаних джерел

1. Elavsky, S., Smahel, D., & Machackova, H. (2017). Who are mobile app users from healthy lifestyle websites? Analysis of patterns of app use and user characteristics. *Translational Behavioral Medicine*, 7(4), 891–901. <https://doi.org/10.1007/s13142-017-0525-x>.
2. Grand View Research, Inc. (2025, April). Fitness App Market Size, Share & Trends Analysis Report By Type (Exercise & Weight Loss, Diet & Nutrition, Activity Tracking), By Platform (Android, iOS), By Device (Smartphones, Tablets), By Region, And Segment Forecasts, 2025–2030. Grand View Research. <https://www.grandviewresearch.com/industry-analysis/fitness-app-market>.
3. CloudFit Enterprises Ltd. (2025). *CloudFit*. CloudFit. <https://www.cloudfit.io/>.
4. Koumpouros, Y. (2022). User-Centric design methodology for mhealth apps: The painapp paradigm for chronic pain. *Technologies*, 10(1), 25. <https://doi.org/10.3390/technologies10010025>.
5. Android Open Source Project. (2025, February 10). *Guide to app architecture*. Android Developers. <https://developer.android.com/topic/architecture>.
6. MyFitnessPal API. (n.d.). *About this API*. <https://myfitnesspalapi.com/docs/about/>.
7. Meta Platforms, Inc. (n.d.). *AsyncStorage*. React Native. <https://reactnative.dev/docs/asyncstorage>.
8. Expo Team. (n.d.). *Expo – An open-source platform for making universal native apps*. <https://expo.dev/docs/expo.dev+14>.

References

1. Elavsky, S., Smahel, D., & Machackova, H. (2017). Who are mobile app users from healthy lifestyle websites? Analysis of patterns of app use and user characteristics. *Translational Behavioral Medicine*, 7(4), 891–901. <https://doi.org/10.1007/s13142-017-0525-x>.
2. Grand View Research. *Fitness app market size & trends*. <https://www.grandviewresearch.com/industry-analysis/fitness-app-market>.
3. Cloudfit. *Cloudfit app*. <https://www.cloudfit.io/>.
4. Koumpouros, Y. (2022). User-Centric Design Methodology for mHealth Apps: The PainApp Paradigm for Chronic Pain. *Technologies*, 10(1). <https://doi.org/10.3390/technologies10010025>.
5. Developers. *Guide to app architecture*. <https://developer.android.com/topic/architecture>.
6. Myfitnesspal. *About this API*. <https://myfitnesspalapi.com/docs/about/>.
7. Meta Platforms, Inc. *React Native: AsyncStorage*. <https://reactnative.dev/docs/asyncstorage>.
8. Expo. *Expo – An open-source platform for making universal native apps*. <https://expo.dev/>.

Отримано 23.06.2025

Vasyl Morokhovych¹, Anton Smolen²

¹PhD in Physical and Mathematical Science,
Associate Professor of the Department of Information Science, Physical and Mathematical Disciplines
Uzhhorod National University (Uzhhorod, Ukraine)

E-mail: vs.mor75@gmail.com, **ORCID:** <https://orcid.org/0000-0002-4939-6566>, **ResearcherID:** [AAJ-6025-2021](https://orcid.org/AAJ-6025-2021)

²Student

Uzhhorod National University (Uzhhorod, Ukraine)

E-mail: smolen.anton@student.uzhnu.edu.ua, **ORCID:** <https://orcid.org/0009-0005-9608-8258>

**DEVELOPMENT OF A MOBILE INFORMATION SYSTEM
TO SUPPORT A HEALTHY LIFESTYLE**

Today, the demand for digital tools supporting a healthy lifestyle is booming. Our research addresses crucial challenges like sedentary living and declining physical activity, which contribute to chronic diseases. Current fitness apps often fall short on personalization, wearable integration, strong motivation, and data security. This drives the need for advanced, cross-platform solutions like React Native.

The core problem lies in the limited customization of existing apps, poor wearable connectivity, weak motivational elements, and clunky user interfaces. Some of them also offer only narrow functionalities. Our goal is to develop a cross-platform mobile fitness app using React Native, Expo, and AsyncStorage, focusing on personalized training, optimized UX/UI, and integrated motivation and progress tracking.

The app's architecture follows the Separation of Concerns principle, with distinct layers for presentation, business logic, data, and navigation. AsyncStorage ensures autonomous, local data storage for user profiles, workouts, and statistics. We used UML diagrams to visualize system functionality and data structure, encompassing UserProfile, Workout, and Statistics entities linked by user IDs. A key feature is the dynamic workout selection based on user goals.

The React Native interface is modular, featuring intuitive tab navigation. It includes a home screen for quick stats, a workout screen, a statistics screen with progress graphs, and a profile management section. An onboarding screen gathers initial data for a tailored experience. Developed with Visual Studio Code, Expo CLI, and tested on Android Emulator, the app offers an intuitive interface, personalized fitness, and reliable local data storage.

This project delivers an innovative architectural solution for an autonomous mobile workout management system, dynamically adapting to user goals and tracking progress without constant internet. This emphasizes personalization and privacy. The functional prototype and detailed architectural models offer practical value, potentially cutting development time for similar systems by 20-30 %. Future research will explore AI for deeper adaptation, enhanced gamification, wearable integration, and secure cloud backup.

Keywords: physical training; healthy lifestyle; mobile application; cross-platform technologies; design.

Fig.: 6. **References:** 7.