

УДК 004.77:004.42:796.062

РОЗРОБКА SAAS-ДОДАТКУ ДЛЯ УПРАВЛІННЯ ФІТНЕС-КЛУБОМ

Міхальов В. В.

**Науковий керівник: доктор техн. наук, професор Жібка В.В., Державний
університет інформаційно-комунікаційних технологій**

*Центральноукраїнський державний університет імені Володимира Винниченка,
м. Кропивницький, Україна*

У статті розглядаються науково-практичні аспекти проектування та розробки SaaS-платформи для автоматизації діяльності фітнес-центрів. Досліджено переваги використання хмарної архітектури порівняно з традиційним програмним забезпеченням. Описано технологічний стек розробки, що включає Next.js, TypeScript та PostgreSQL. Особлива увага приділяється створенню багаторівневої системи доступу для адміністраторів, тренерів та клієнтів. Результатом роботи є розроблене програмне забезпечення, яке інтегрує фінансовий облік, управління персоналом та сервіси для користувачів у єдиному веб-інтерфейсі.

Ключові слова: SaaS-додаток, хмарні обчислення, автоматизація фітнес-центру, Next.js, веб-технології, цифрова трансформація.

Development of a SaaS application for fitness club management

V. Mikhalov

**Scientific supervisor: Doctor of Technical Sciences, Professor Zhebka V.V., State University
of Information and Communication Technologies**

Volodymyr Vynnychenko Central Ukrainian State University, Kropyvnytskyi, Ukraine

This paper is devoted to the design and implementation of a SaaS (Software as a Service) platform for automating business processes in the fitness industry. The article analyzes the current state of digital transformation in sports services, justifies the expediency of using cloud-based solutions to optimize club management, and explores the architectural capabilities of the Next.js framework and PostgreSQL database for creating scalable web applications. A detailed structure of the system with a multi-level access model for administrators, coaches, and clients is presented. A functional system for subscription management, online payments integration via Stripe, and interactive training schedules has been developed. Particular attention is paid to data security, role-based access control, and the development of a responsive user interface to improve the efficiency of fitness center operations.

Keywords: SaaS, fitness club, business process automation, Next.js, cloud technologies, web development, PostgreSQL, management systems.

Постановка проблеми. Стрімкий розвиток цифрових технологій докорінно змінив підходи до ведення бізнесу в усіх сферах послуг, і фітнес-індустрія не є винятком. Сучасні спортивні заклади стикаються з проблемою ефективного управління великими обсягами даних: клієнтськими базами, розкладами тренувань, фінансовими транзакціями та складськими запасами. Традиційні методи ведення обліку або застарілі локальні програмні продукти вже не задовольняють вимогам мобільності та масштабованості. Виникає потреба у впровадженні хмарних рішень, які дозволяють керувати бізнес-процесами в режимі реального часу з будь-якої точки світу.

Актуальність розробки SaaS-додатку (Software as a Service) для фітнес-клубів зумовлена необхідністю зниження витрат на ІТ-інфраструктуру та підвищення якості обслуговування клієнтів. Модель SaaS забезпечує автоматичне оновлення системи, високий рівень безпеки даних та доступність сервісу за підпискою, що є вигідним як для власників бізнесу, так і для розробників. У контексті глобальної цифровізації створення вітчизняного конкурентоспроможного програмного забезпечення (ПЗ) для спортивної галузі є стратегічно важливим завданням.

Аналіз останніх досліджень і публікацій. Питанням архітектури SaaS-систем та автоматизації бізнес-процесів присвячено багато робіт як вітчизняних, так і зарубіжних авторів. Проте, незважаючи на значну кількість існуючих CMS-систем (наприклад, Fitbase або Glofox), залишається відкритим питання створення універсальних платформ, які б ефективно поєднували функціонал для адміністрації та персональні сервіси для клієнтів (онлайн-запис, оплата, зворотний зв'язок) у межах єдиного архітектурного рішення на базі сучасних Full-stack фреймворків.

Мета статті – розробка програмного забезпечення, яке інтегрує фінансовий облік, управління персоналом та сервіси для користувачів у єдиному веб-інтерфейсі.

Виклад основного матеріалу (результатів) дослідження. Аналіз існуючих засобів автоматизації показує, що на теперішній час ринок

програмного забезпечення для фітнес-індустрії представлений широким спектром рішень. Серед найбільш відомих систем можна виділити такі: Mindbody, Glofox та Fitbase. Проте їх детальний аналіз дозволив виявити ряд недоліків, що обмежують їхнє застосування в умовах сучасного українського ринку. Виділимо ці недоліки: висока вартість впровадження; низька гнучкість; обмежена інтеграція.

Висока вартість впровадження полягає в тому, що більшість систем орієнтовані на великі мережеві фітнес-центри і мають складну модель ціноутворення. Низька гнучкість пов'язана з відсутністю можливості швидкої адаптації під потреби конкретного закладу або незалежного тренера. Обмежена інтеграція характеризується складністю поєднання функціоналу для клієнта (запис, оплата) та внутрішнього обліку адміністрації в одному легкому веб-додатку.

Таким чином, було прийнято рішення про розробку власної SaaS-платформи, яка базується на архітектурному принципі поділу обов'язків (Separation of Concerns) та використанні хмарних технологій.

Обґрунтуємо вибір технологічного стеку. Для реалізації проекту обрано сучасний та масштабований стек технологій, який дозволяє забезпечити високу швидкість розробки та стабільність системи.

Next.js (Framework). Використання Next.js як основного фреймворку обумовлено його здатністю обробляти як клієнтську, так і серверну частини додатку. Підтримка технології Server-Side Rendering (SSR) та Static Site Generation (SSG) дозволяє значно пришвидшити завантаження сторінок та покращити індексацію системи пошуковими роботами.

PostgreSQL (Database). Для зберігання структурованих даних обрано реляційну СУБД PostgreSQL. Вона забезпечує високу надійність транзакцій, підтримку складних запитів та можливість масштабування бази даних при збільшенні кількості користувачів.

Prisma (ORM). Використання ORM-системи Prisma дозволило типізувати запити до бази даних, що мінімізує кількість помилок на етапі розробки та спрощує підтримку коду.

TypeScript. Вибір мови програмування TypeScript забезпечує строгу типізацію, що критично важливо для великих проєктів, де інтегровано фінансові операції та складну логіку прав доступу.

Stripe API. Для реалізації платіжного функціоналу обрано Stripe як один із найбільш захищених та гнучких інструментів для обробки транзакцій у моделі SaaS.

Опишемо архітектурну модель додатка. Розроблений SaaS-додаток базується на багаторівневій архітектурі, що дозволяє ізолювати логіку обробки даних від інтерфейсу користувача. Враховуючи специфіку моделі «програмне забезпечення як сервіс», було реалізовано механізм мультиорендності (multi-tenancy) на рівні логіки додатка, де дані різних фітнес-клубів зберігаються в єдиній базі, але розмежовуються за допомогою унікальних ідентифікаторів організацій.

Проєктування бази даних охарактеризуємо через структуру бази даних, що розроблена з урахуванням високої зв'язності об'єктів. Основними сутностями системи є: User, Gym, Subscription, Workout/Class, Transaction.

User зберігає дані про всіх зареєстрованих осіб та їхні ролі (ADMIN, COACH, CLIENT). *Gym* містить інформацію про конкретний фітнес-центр (назва, адреса, налаштування). *Subscription* описує типи абонементів, їхню вартість та тривалість. *Workout/Class* описує графік занять, що пов'язує тренера, клієнта та конкретний часовий слот. *Transaction* фіксує всі фінансові операції через інтеграцію зі Stripe.

Охарактеризуємо функціональні модулі системи. Відповідно до визначених ролей користувачів, у системі реалізовано три основні функціональні блоки.

1. Модуль адміністратора клубу (Admin Dashboard). Цей блок є центром управління бізнес-процесами. Він дозволяє здійснювати наступні дії:

проводити моніторинг фінансових показників (дохід за період, кількість проданих абонементів); керувати списком співробітників (призначення тренерів, налаштування їхніх профілів); редагувати сітку розкладу та типів послуг.

2. Модуль тренера (Coach Interface). Орієнтований на операційну роботу з клієнтами. Основні функції включають: перегляд персонального графіку тренувань; відмітку відвідуваності клієнтів; доступ до карток клієнтів для відстеження їхнього прогресу.
3. Клієнтський інтерфейс (Client Portal). Реалізований як адаптивна веб-сторінка, що дозволяє користувачу здійснювати наступне: самостійно обирати та оплачувати абонементи онлайн; записуватися на групові або індивідуальні заняття; переглядати історію своїх транзакцій та активних послуг.

Інтерфейс та User Experience (UX). При розробці фронтенд-частини на Next.js особлива увага приділялася швидкості відгуку інтерфейсу. Використання бібліотеки Tailwind CSS дозволило створити адаптивний дизайн, який однаково коректно відображається як на робочих станціях адміністраторів, так і на смартфонах клієнтів.

Забезпечення безпеки та цілісності даних. Оскільки SaaS-додаток передбачає зберігання персональних даних клієнтів та проведення фінансових операцій, особливу увагу було приділено безпеці. У системі реалізовано автентифікацію та авторизація, шифрування, а також безпека платежів. Автентифікація та авторизація полягає у використанні сучасних протоколів (через інтеграцію з Clerk або NextAuth), що забезпечує захищений вхід та рольову модель доступу, розкиває. Шифрування забезпечується тим, що всі дані передаються через протокол HTTPS, а конфіденційна інформація в базі даних PostgreSQL захищена на рівні інфраструктури хмарного провайдера. Завдяки використанню Stripe API реалізовано безпеку платежів, а саме, додаток не зберігає дані банківських карток на своїх серверах, що відповідає стандарту PCI DSS.

Впровадження розробленого SaaS-додатку в діяльність фітнес-центру дозволяє досягти наступних результатів.

1. Зменшення операційних витрат (автоматизація запису та оплат знижує навантаження на адміністраторів на 30-40%).
2. Підвищення лояльності клієнтів (можливість керувати послугами через смартфон підвищує рівень утримання клієнтів (Retention Rate)).
3. Мінімізація помилок (автоматичний облік відвідувань та термінів дії абонементів виключає людський фактор у фінансовій звітності).

Висновки та перспективи подальших пошуків у напрямі дослідження.

У ході дослідження було розроблено та описано SaaS-додаток для управління фітнес-клубом, який є комплексним інструментом цифровізації спортивної індустрії. Виконана робота дозволяє зробити наступні висновки.

1. Доведено концептуальну перевагу впровадження хмарної моделі SaaS (Software as a Service) для автоматизації процесів у сфері фітнес-індустрії. Встановлено, що перехід на хмарні рішення не лише оптимізує операційні витрати, а й суттєво підвищує масштабованість бізнесу та рівень доступності сервісів для кінцевих користувачів у режимі 24/7.
2. Науково обґрунтовано вибір технологічного стеку на основі Next.js, TypeScript та PostgreSQL. Поєднання цих інструментів забезпечує оптимальний баланс між продуктивністю інтерфейсу, типізацією даних та надійністю збереження інформації, що є критичним для безпечного функціонування сучасних вебдодатків.
3. Трирівнева архітектура доступу забезпечує ефективну взаємодію між власниками бізнесу, тренерським складом та кінцевими споживачами послуг. Даний архітектурний підхід дозволяє мінімізувати конфлікти доступу та підвищити ефективність управління цифровою екосистемою спортивного закладу.

Перспективи подальших досліджень полягають в інтеграції в систему модулів штучного інтелекту для аналізу тренувальних планів та розробки

нативного мобільного додатка для iOS та Android для ще зручнішої взаємодії з клієнтами.

Список використаної літератури:

1. Антощук С. Г., Ніколаєв О. В., Кузнецова С. А. Технології хмарних обчислень : навч. посіб. Одеса : ОНПУ, 2020. 184 с.
2. Бичков О. С., Березовська Ю. В. Проектування архітектури сучасних хмарних застосунків. *Вісник Київського національного університету імені Тараса Шевченка. Серія: Фізико-математичні науки*. 2021. № 2. С. 45–51.
3. Плєскач В. Л., Затонацька Т. Г. Електронна комерція : підручник. Київ : Знання, 2016. 535 с.
4. Next.js Documentation. Vercel. URL: <https://nextjs.org/docs> (дата звернення: 12.11.2025).
5. PostgreSQL 16.0 Documentation. *The PostgreSQL Global Development Group*. URL: <https://www.postgresql.org/docs/16/index.html> (дата звернення: 10.11.2025).
6. Stripe API Reference. *Stripe*. URL: <https://stripe.com/docs/api> (дата звернення: 15.11.2025).
7. Mell P., Grance T. The NIST Definition of Cloud Computing. *National Institute of Standards and Technology*. 2011. URL: <https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf> (дата звернення: 18.10.2025).
8. Ляшенко В.П., Назарова О.С. Огляд сучасних фреймворків для розробки високонавантажених веб-систем. *Системи обробки інформації*. 2022. Вип.1 (168). С. 112–119.
9. Бублик М. В., Коритко Т. Ю. Цифрова трансформація сфери послуг в умовах розбудови інформаційного суспільства. *Економіка та суспільство*. 2023. № 48. URL: <https://economyandsociety.in.ua/index.php/journal/article/view/2214> (дата звернення: 20.10.2025).

Відомості про автора:

Міхальов Владислав Валерійович – студент II курсу магістратури факультету інформаційних технологій, математики та природничих наук Центральноукраїнського державного університету імені Володимира Винниченка, тел. +380500264609, e-mail: vmihalov@ukr.net.