

Павло Ярема, Ірина Борецька*

¹ НЛТУ України, Львів, Україна, 23yarema.p@nltu.lviv.ua

² НЛТУ України, Львів, Україна. ORCID: 0000-0002-6767-104X,
boretska@nltu.edu.ua

Розроблення рекомендаційної системи для персоналізованих тренувань на базі Spring Framework з інтеграцією API штучного інтелекту

Анотація. Розроблено рекомендаційну систему для персоналізованих графіків тренувань, що використовує фреймворк Spring та інтеграцію API для автоматичного генерування графіків. Основна мета – аналіз даних користувачів та надання індивідуальних планів тренувань на основі фітнес-цілей, параметрів здоров'я та вподобань користувачів. Система застосовує сучасні вебтехнології та алгоритми машинного навчання для підвищення зручності користувача та ефективності тренувань. Такий підхід забезпечує персоналізовані рекомендації, що сприяють зростанню залученості користувачів та покращенню результатів тренувань.

Ключові слова – рекомендаційна система, індивідуальні тренування, фреймворк Spring, машинне навчання, персоналізовані тренування, WebSocket, STOMP.

Сучасний розвиток вебтехнологій та машинного навчання відкриває широкі можливості для створення інтелектуальних, адаптивних систем, які можуть ефективно задовольняти потреби користувачів у сфері фітнесу та здорового способу життя [1].

Дослідження спрямоване на використання фреймворку Spring для розробки рекомендаційної системи, яка автоматично генерує індивідуальні графіки тренувань, враховуючи персональні дані користувачів. Система містить механізми оброблення даних, валідації введення користувачів та динамічної генерації планів тренувань [2].

Основні завдання дослідження зосереджені на:

- розробленні вебсайту з рекомендаційною системою, що підтримує автоматизоване створення графіків тренувань на основі характеристики;

- створенні структурованої моделі даних, яка забезпечує підтримку динамічних графіків тренувань та зручну інтеграцію нових параметрів для персоналізації;
- використанні фреймворку Spring для організації взаємодії між компонентами системи. Застосування WebSocket і STOMP для реалізації чату, який підтримує комунікацію в реальному часі, а також використання RestTemplate для інтеграції з різними API [1].

Програмна реалізація. Для підтримки реального часу у чаті використовується WebSocket. Приклад коду конфігурації WebSocket на основі Spring [3]:

```
@Data
@Configuration
@ConfigurationProperties(prefix = "spring.rabbitmq")
@EnableWebSocketMessageBroker
public class WebSocketConfig implements WebSocketMessageBrokerConfigurer {
    private String host = "localhost";
    private Integer port = 61613;
    private String username = "guest";
    private String password = "guest";

    @Override
    public void registerStompEndpoints(StompEndpointRegistry registry) {
        registry.addEndpoint("/ws").withSockJS();
    }

    @Override
    public void configureMessageBroker(MessageBrokerRegistry registry) {
        registry.enableStompBrokerRelay("/topic")
            .setRelayHost(host).setRelayPort(port).setClientLogin(username).setClientPasscode(password)
            .setSystemLogin(username).setSystemPasscode(password);
    }
}
```

Система зберігає дані користувачів, такі як вік, рівень фізичної підготовки та показники здоров'я, у реляційній базі даних, що дає змогу створювати адаптивні графіки тренувань. Структура основних таблиць бази даних: users – таблиця для зберігання даних користувачів (ID, ім'я, вік, рівень підготовки, стан здоров'я тощо); workout_plans – таблиця для запису тренувальних планів, зв'язаних з користувачами (ID плану, дата початку, інтенсивність, типи тренувань); workout_days – таблиця для збереження днів тренувань у рамках плану; message_general_chat – таблиця, що зберігає історію чату користувачів [4].

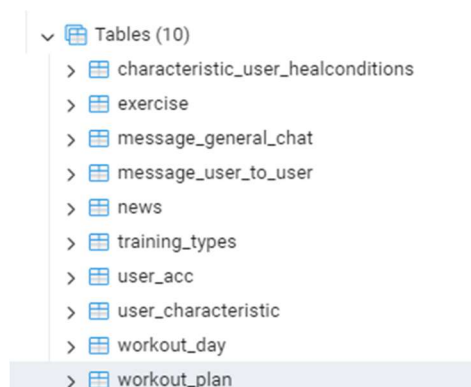


Рисунок 1. Таблиці бази даних

Приклад інтерфейсу. Графік тренувань генерується на основі зібраних даних користувача та зберігається у структурованій формі в базі даних. Ці дані можна відобразити на вебсайті, де користувач бачить, наприклад, розклад тренувань з описом.

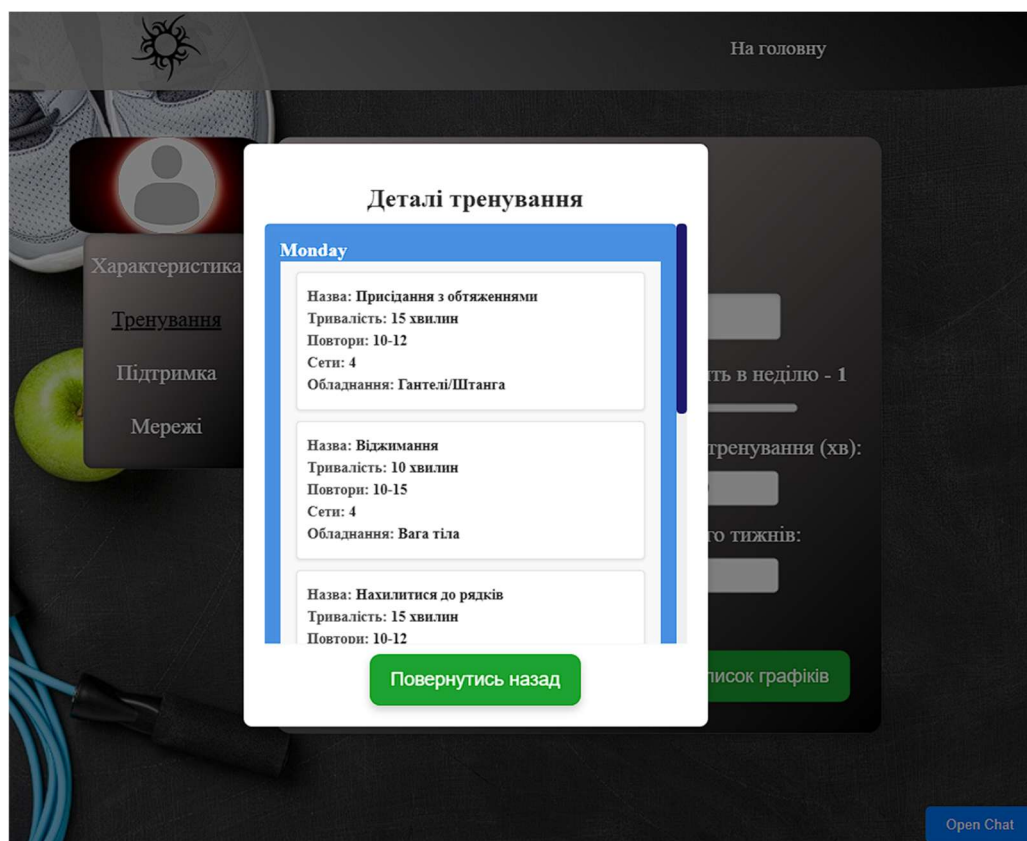


Рисунок 2. Інтерфейс отримання графіку тренувань

Висновок. Розроблення рекомендаційної системи на основі фреймворку Spring демонструє високу ефективність у пропонуванні індивідуальних тренувальних програм, адаптованих до особистих параметрів користувачів. Це дослідження показує, що такі інструменти можуть стати потужною підтримкою для

сучасних користувачів, які прагнуть персоналізованих рішень у фітнес-середовищі. Подальші дослідження можуть зосередитися на глибшій інтеграції аналітики на основі штучного інтелекту та розробці механізмів зворотного зв'язку в реальному часі.

Список використаних літературних джерел

- [1] Walls, C. (2020). Spring in Action, 6th Edition. Manning Publications.
- [2] Aurelien Geron. (2019). "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow". O'Reilly Media, Inc.
- [3] Klymenko, O. Java WebSocket в Spring: Вебсокети для інтерактивних вебдодатків [Електронний ресурс]. Режим доступу URL: <https://drukarnia.com.ua/articles/java-websocket-spring-websocket-A0uya>
- [4] Christian Bauer, Gavin King, Gary Gregory (2015). Java Persistence with Hibernate Second Edition. Manning Publications; Second edition.

Development of a recommendation system for personalized training based on the Spring Framework with AI API integration

Pavlo Yarema, Iryna Boretska

A recommendation system has been developed for personalized training schedules, using the Spring framework and API integration for automatic schedule generation. The main objective is to analyze user data and provide individualized training plans based on fitness goals, health parameters, and user preferences. The system leverages modern web technologies and machine learning algorithms to enhance user convenience and training effectiveness. This approach offers personalized recommendations, which contribute to increased user engagement and improved training outcomes.