**Siłka++ koncepcja wykonania systemu**

Marcin Sarnecki, Marcin Wróbel

**2. Scenariusze użycia**

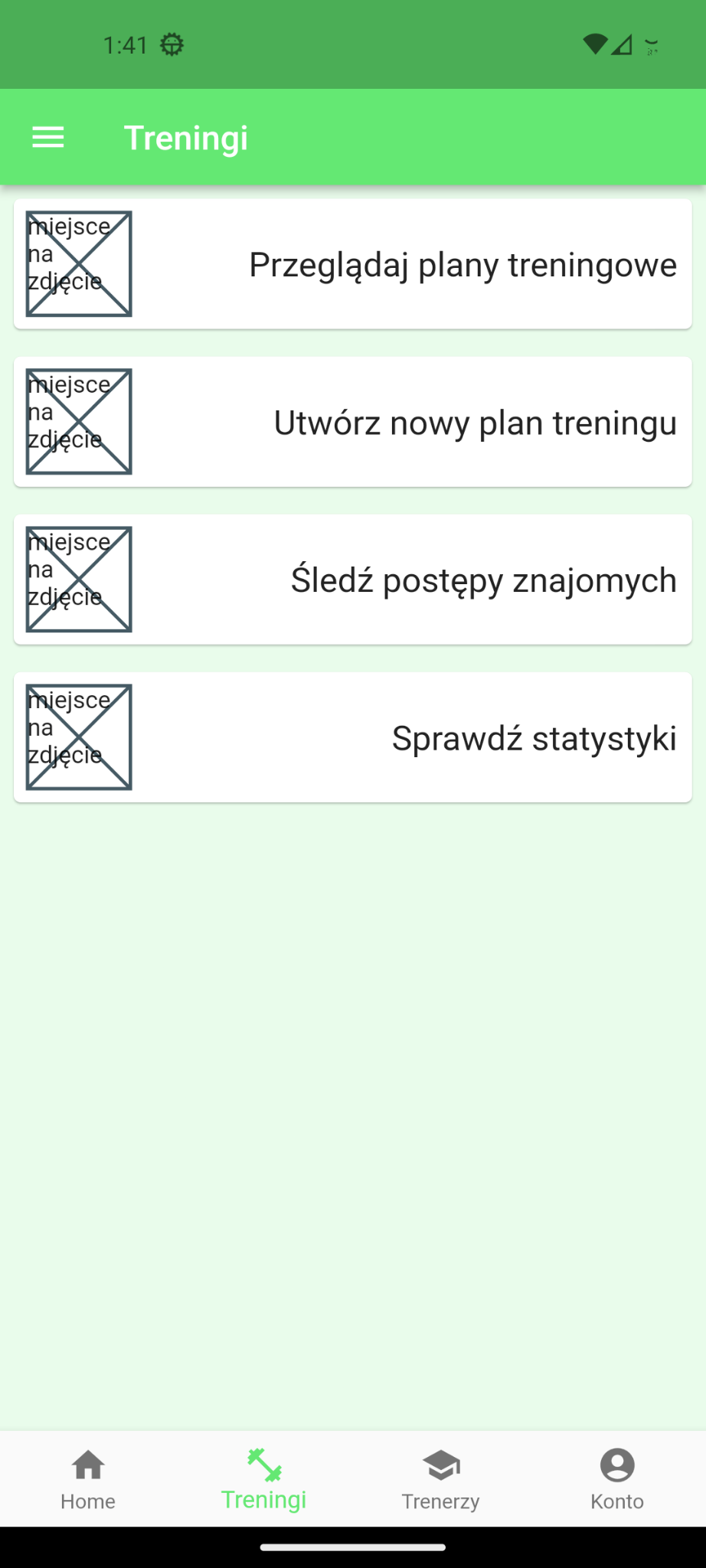
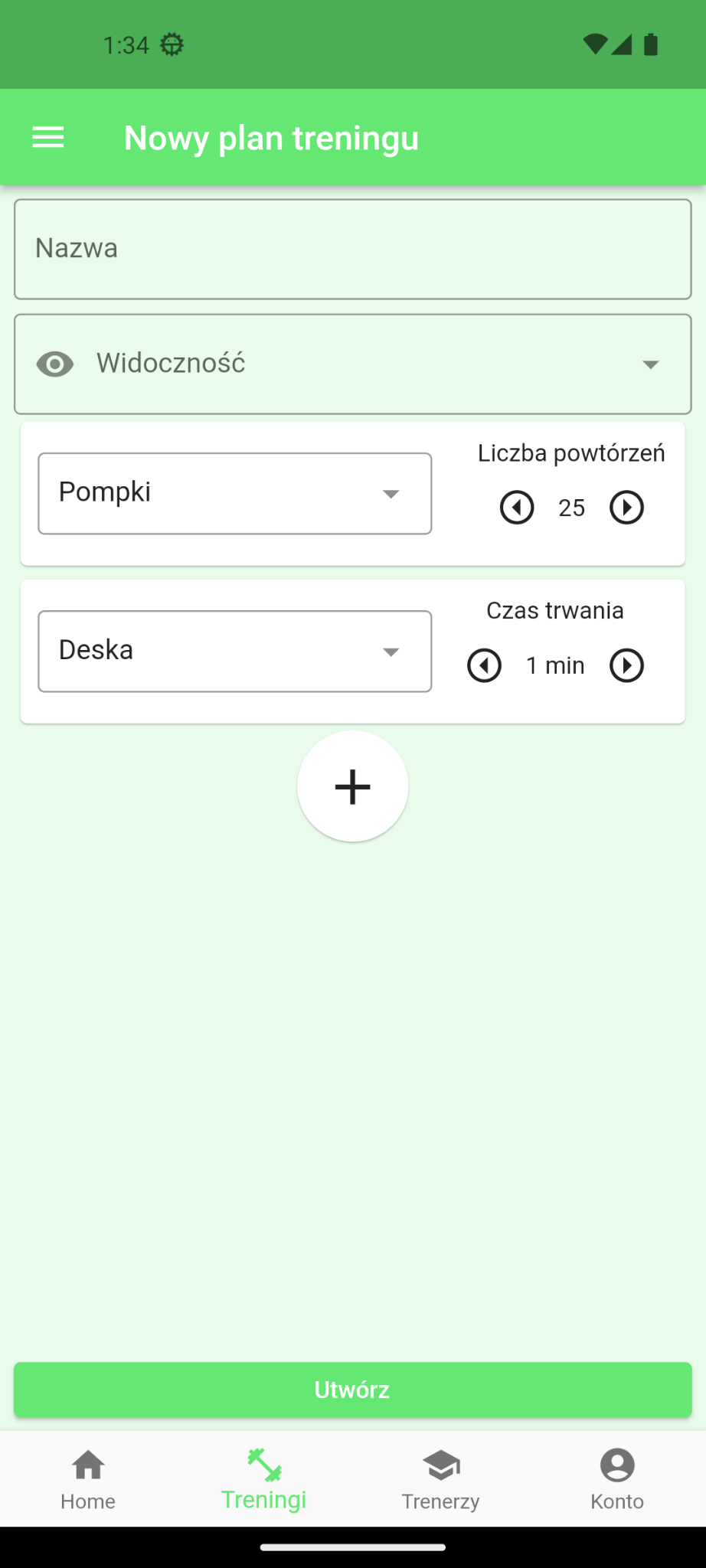
| Tytuł | Stworzenie planu treningowego |
| --- | --- |
| Opis | Użytkownik chce stworzyć nowy plan treningowy |
| Inicjator | Użytkownik |
| Warunki wstępne | Użytkownik jest zalogowany w aplikacji |
| Cel | Użytkownik dodał plan treningowy |
| Główny scenariusz | 1. Użytkownik wchodzi w zakładkę treningi. 2. System wyświetla dostępne opcje: przeglądaj zapisane plany treningowe; utwórz nowy plan treningowy; śledź postępy znajomych. 3. Użytkownik klika przycisk utworzenia nowego planu. 4. System wyświetla początkowo pustą listę ćwiczeń w nowym planie treningowym, oraz pole wyszukiwania ćwiczeń, pole do wpisania nazwy. 5. Użytkownik wypełnia pole nazwy planu. 6. Użytkownik wyszukuje ćwiczenia i dodaje je do planu. 7. Użytkownik dla każdego ćwiczenia może wybrać odpowiadające mu parametry, takie jak: liczbę powtórzeń; czas wykonywania; ciężar. 8. Użytkownik zatwierdza utworzenie całego planu poprzez kliknięcie przycisku “utwórz”. |
| Scenariusze alternatywne | 5a. Podana przez użytkownika nazwa planu. treningowego jest taka sama, jak nazwa innego planu tego użytkownika.  5a1. System wyświetla komunikat, że nazwa planu jest taka sama jak nazwa innego planu.  5a2. System blokuje możliwość naciśnięcia przycisku utwórz.  6a. Użytkownik nie dodał żadnego ćwiczenia do planu.  6a1. System wyświetla komunikat, że plan musi zawierać co najmniej jedno ćwiczenie.  6a2. System blokuje możliwość naciśnięcia przycisku utwórz. |

| Tytuł | Publiczne ogłoszenie się trenera |
| --- | --- |
| Opis | Trener chce przedstawić swoją ofertę, aby dotrzeć do użytkowników aplikacji |
| Inicjator | Użytkownik-trener |
| Warunki wstępne | Użytkownik jest zalogowany w aplikacji jako trener |
| Cel | Użytkownik dodaje swoje ogłoszenie |
| Główny scenariusz | 1. Użytkownik wchodzi w zakładkę trenerzy. 2. System wyświetla listę ogłoszeń, pole wyszukiwania i przycisk “plus” oznaczający dodaj ogłoszenie; 3. Użytkownik klika przycisk dodania ogłoszenia 4. System wyświetla formularz dodawania ogłoszenia zawierający pole nazwy, listę kategorii, lokalizacji, danych kontaktowych, dokładnego opisu, dotychczasowego doświadczenia w zawodzie, cenę konsultacji. 5. Użytkownik wypełnia wyżej wymienione pola.   6. Użytkownik klika przycisk utwórz ogłoszenie.  7. System dodaje ogłoszenie do bazy danych |
| Scenariusze alternatywne | 5. Użytkownik nie wypełnił jakiegoś pola  5a1. System wyświetla komunikat, że dane pole nie zostało wypełnione  5a2. System blokuje możliwość naciśnięcia przycisku utwórz. |

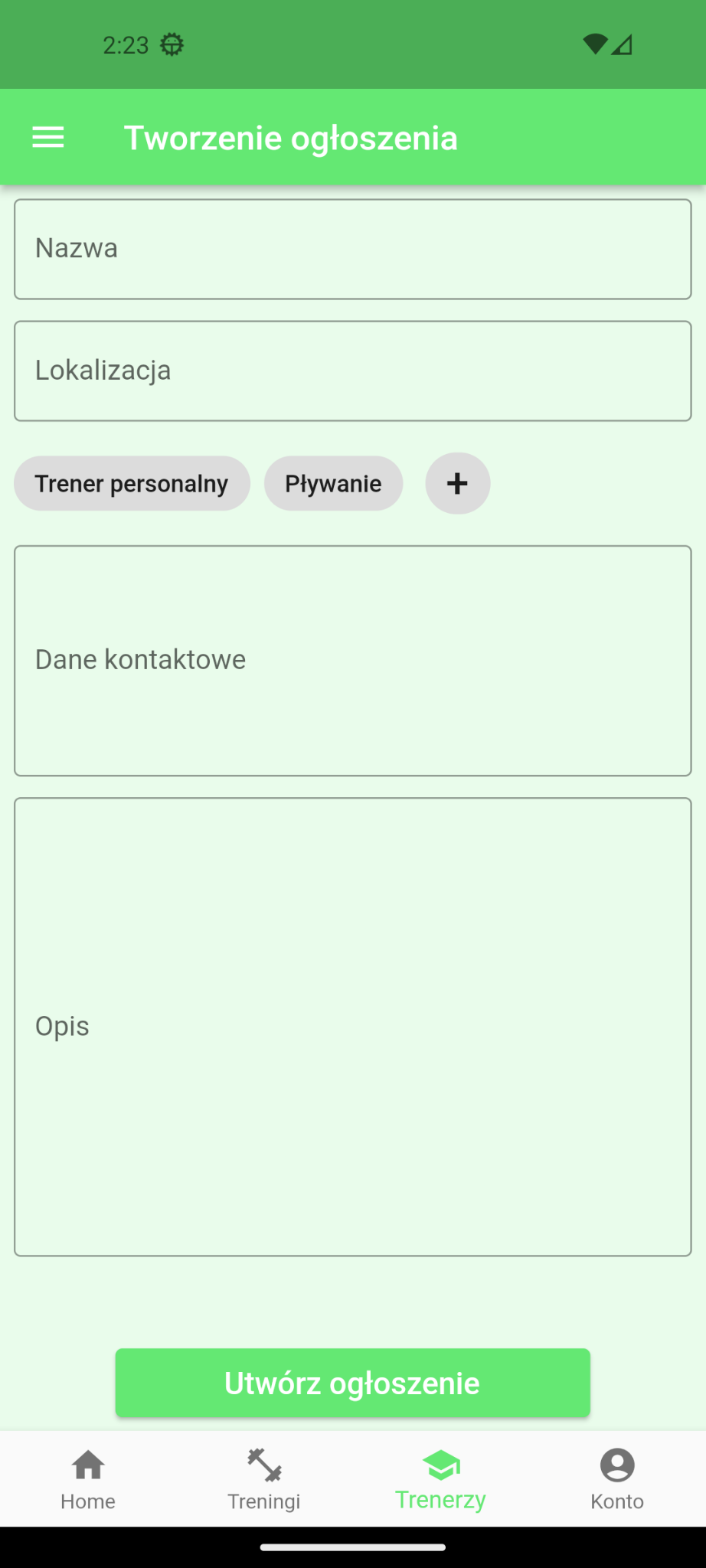
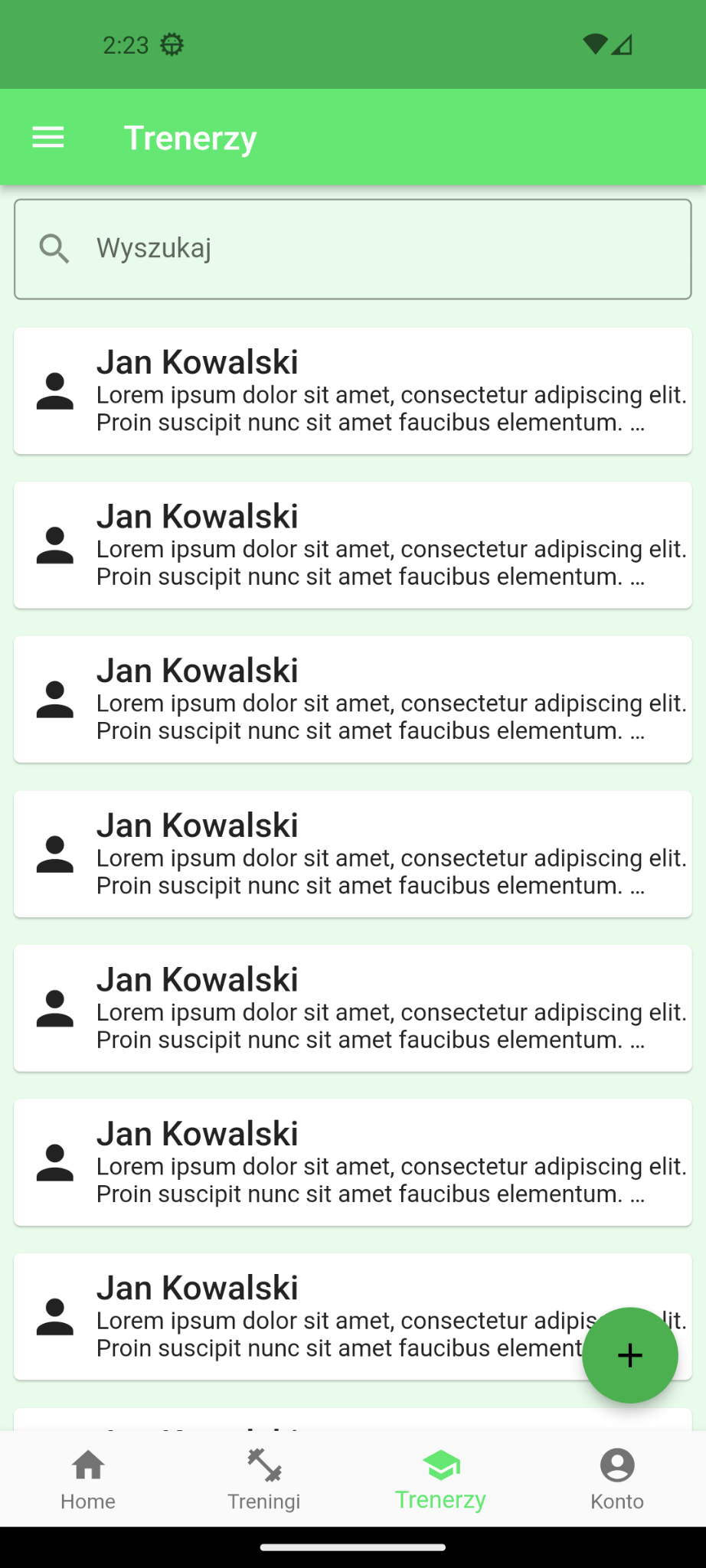
| Tytuł | Wykonanie treningu |
| --- | --- |
| Opis | Użytkownik w trakcie treningu zapisuje wykonywane ćwiczenia, a na końcu zapisuje cały trening |
| Inicjator | Użytkownik |
| Warunki wstępne | Użytkownik jest zalogowany w aplikacji |
| Cel | Wykonanie treningu i zapisanie go |
| Główny scenariusz | 1. Użytkownik wchodzi w zakładkę treningi. 2. System wyświetla dostępne opcje: przeglądaj zapisane plany treningowe; utwórz nowy plan treningowy; śledź postępy znajomych. 3. Użytkownik wybiera przeglądaj zapisane plany treningowe. 4. Użytkownik znajduje wybrany plan treningowy, klika przycisk “trenuj”. 5. Aplikacja wyświetla ćwiczenia po kolei. 6. Po zakończeniu każdego ćwiczenia użytkownik klika przycisk “następne ćwiczenie”. 7. Po wykonaniu ostatniego ćwiczenia aplikacja pokazuje podsumowanie treningu. 8. Aplikacja proponuje użytkownikowi podanie własnej wagi i/lub innych parametrów ciała w celu śledzenia statystyk. |
| Scenariusze alternatywne | 6a. Użytkownik decyduje się zatrzymać jakieś ćwiczenie  6a1. Użytkownik klika przycisk “pauza”  6a2. Aplikacja zapamiętuje, że ćwiczenie zostało zatrzymane i odpowiednio uwzględni to w wyświetlonym na końcu podsumowaniu |

**3. Projekty ekranów**

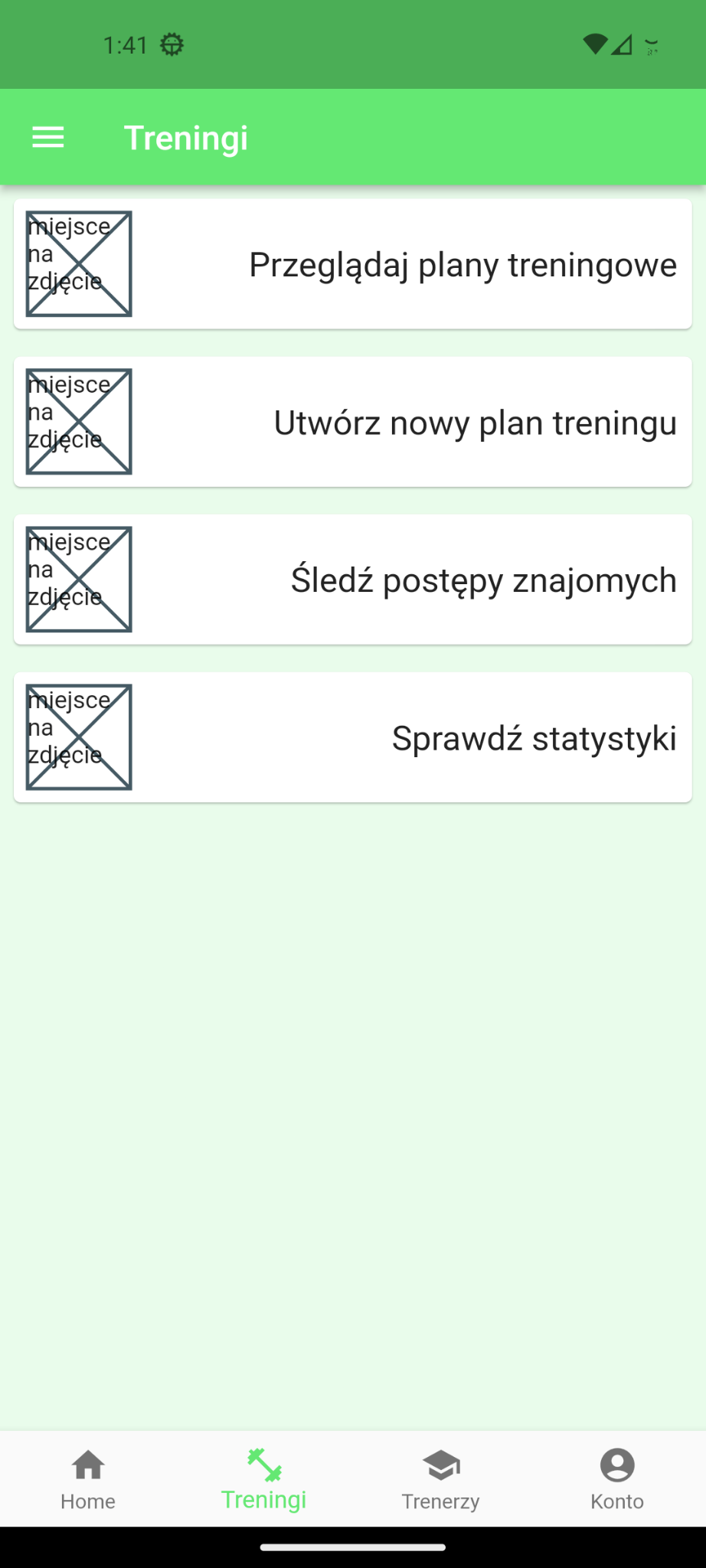
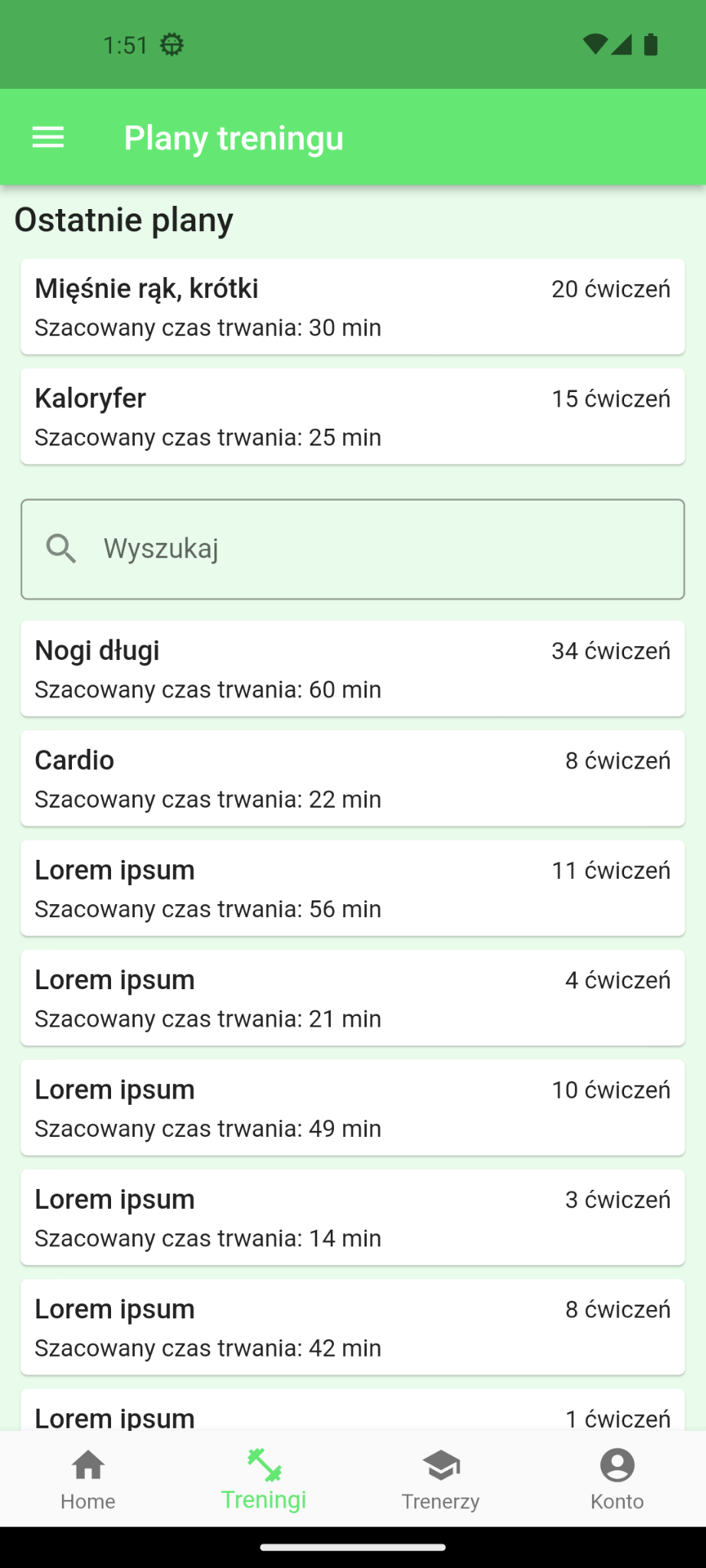
**Tworzenie planu treningowego:**

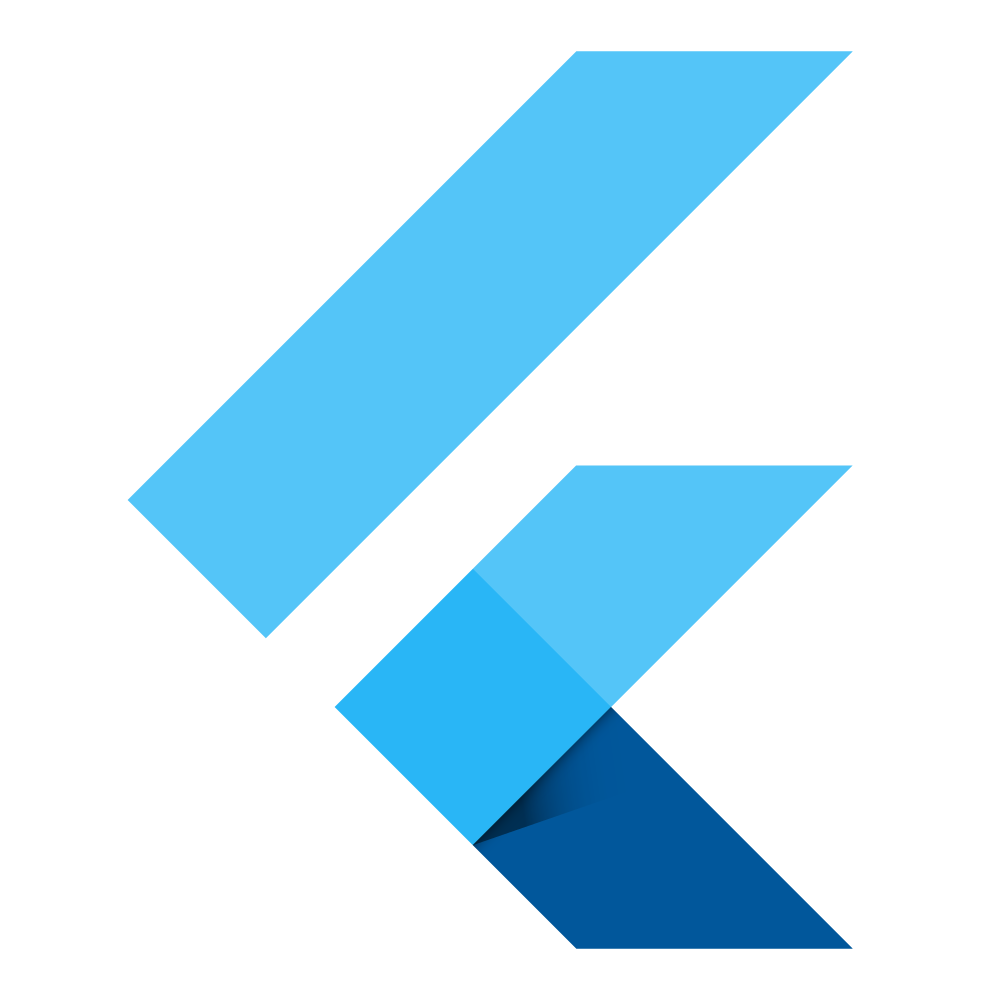
** **

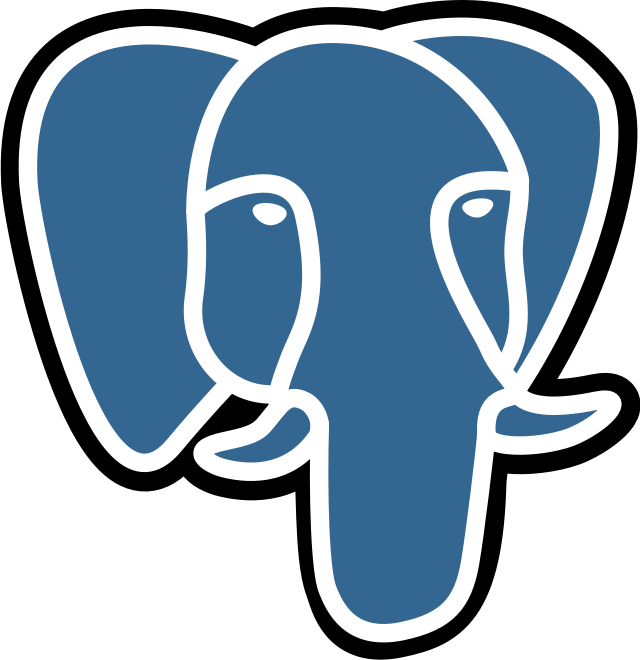
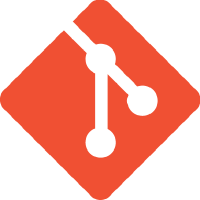
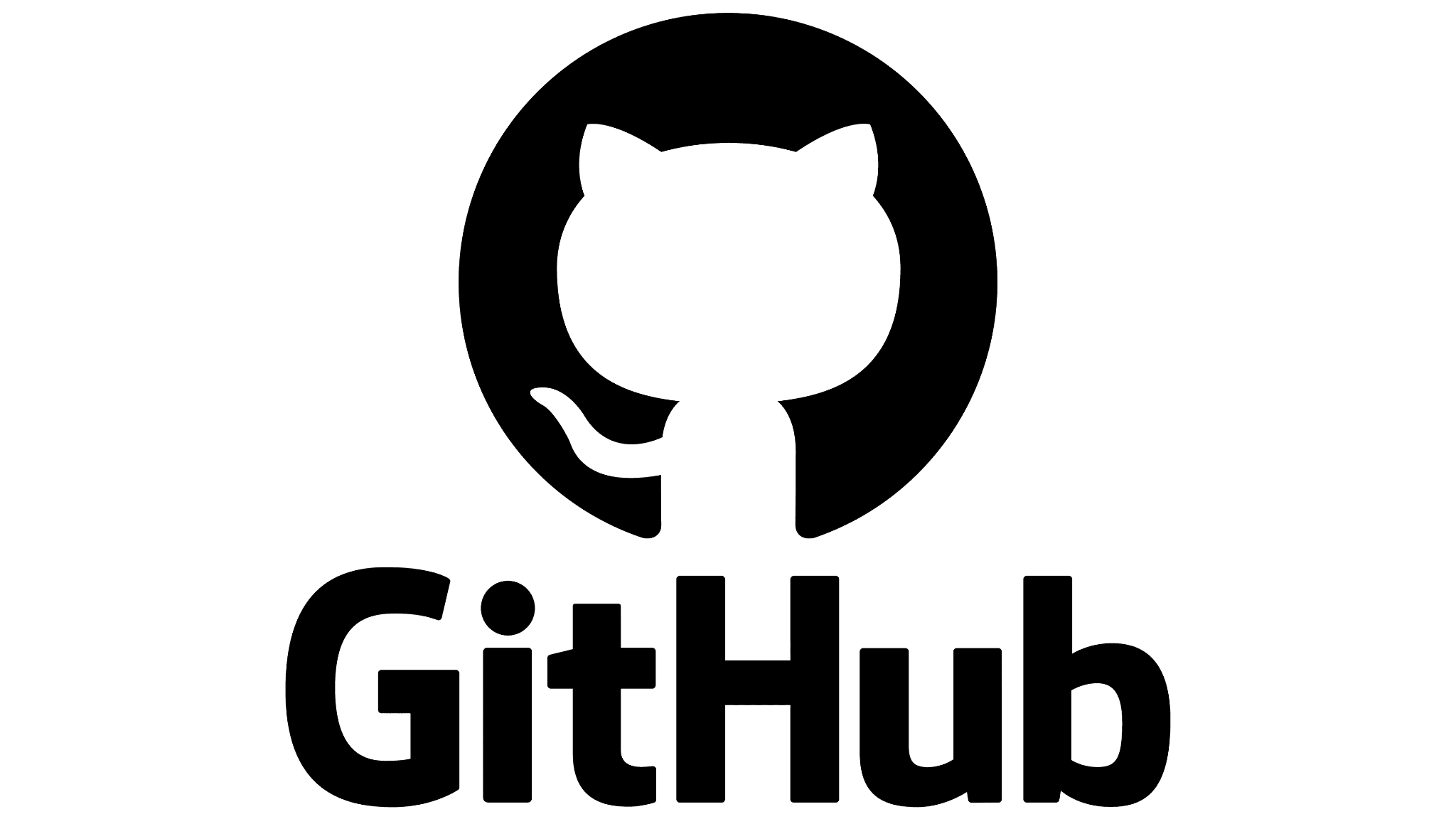
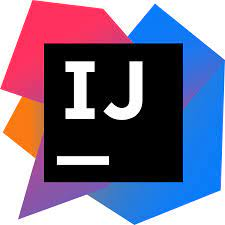
**Dodanie ogłoszenia trenera:**

****

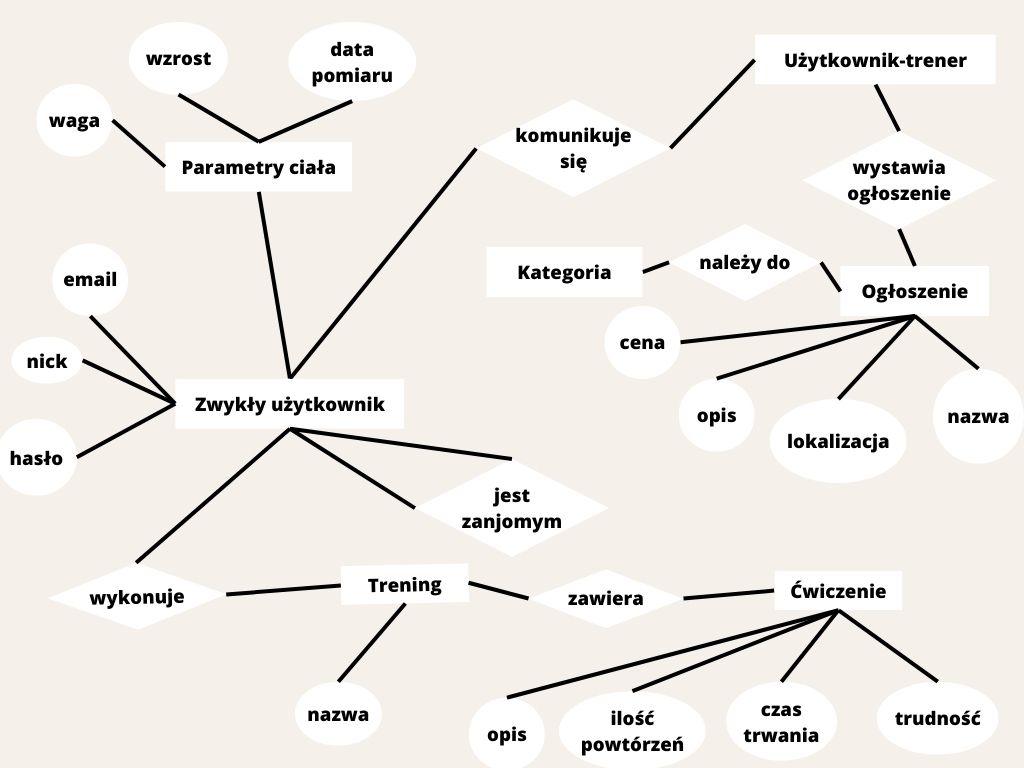
**Wykonanie treningu:**

**  **

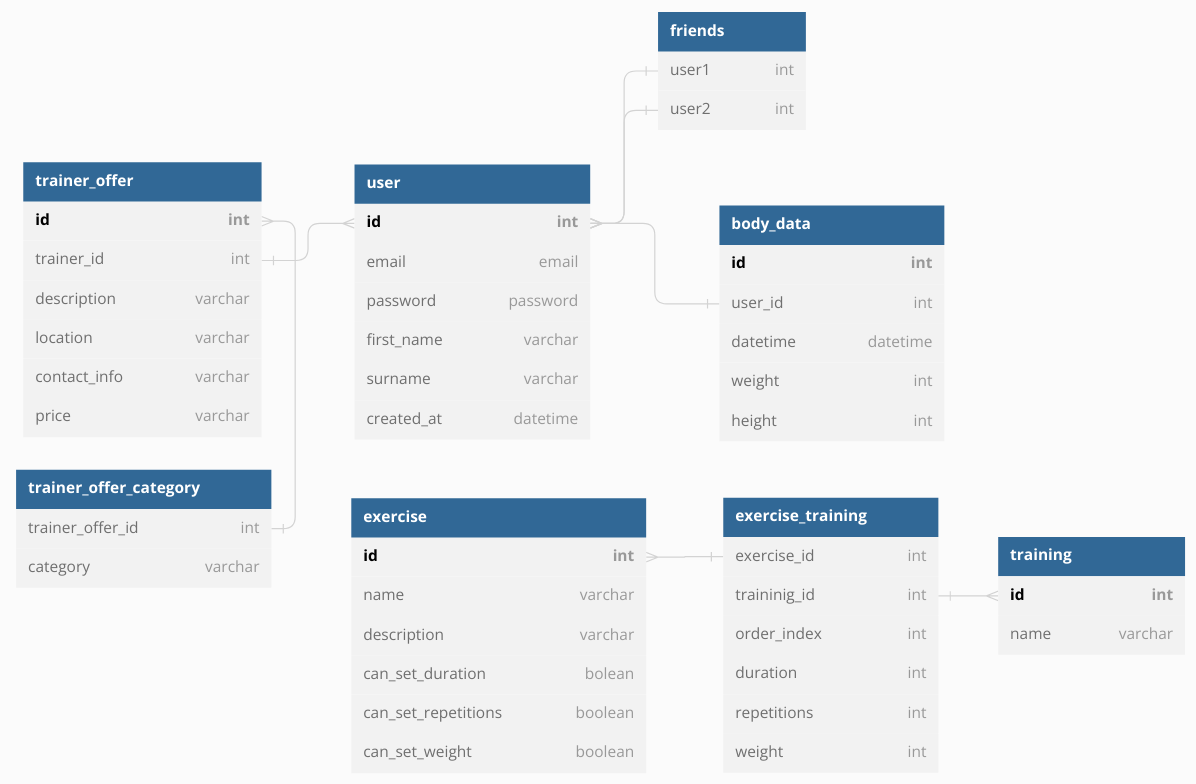
**4. Architektura systemu**

* Aplikacja mobilna zostanie stworzona w frameworku Flutter
* Backend zostanie napisany w Spring Bootcie
* Bazą danych będzie PostgreSQL 
* Git - system kontroli wersji 
* Repozytorium Gita będzie przechowywane na GitHubie
* Codemagic.io do Continuous Integration i Continuous Delivery (CI/CD)
* Do tworzenia oprogramowania będziemy używać IntelliJ IDEA
* Serwer będzie używał systemu operacyjnego Ubuntu Sever w najnowszej wersji LTS

**5A. Model konceptualny**



**5B. Schemat bazy danych**



**6. Główne zasady kodowania**

Zarówno kod, jak i komentarze będą w języku angielskim. Rozwój backend’u podlega zasadom *Google Java code Style Guide*. Kod frontendu będzie zgodny z *dart format*. Repozytorium będzie hostowane na Githubie. W każdym tygodniu będą odbywać się spotkania online programistów, aby wspólnie omówić bieżące sprawy związane z kodem.

**7. Identyfikacja ryzyka i opracowanie zasad zarządzania ryzykiem**

Istnieje ryzyko niedostarczenia aplikacji w wyznaczonym terminie. W razie niezrealizowania terminów dla odpowiednich funkcjonalności systemu, zostaną przeprowadzone dodatkowe spotkania z programistami, aby wspólnie rozwiązać zaistniałe problemy.

Istnieje ryzyko, że aplikacja po wydaniu będzie zawierała bugi, których testerzy nie zdołali wyłapać. W takim przypadku użytkownicy będą mogli skontaktować się z twórcami, aby opisać, co nie działa i co należy natychmiast naprawić.

Istnieje ryzyko, że serwery aplikacji będą przeciążone w najbardziej obleganych momentach dnia. Możliwym rozwiązaniem jest wykorzystanie load balancingu, na podstawie gotowych rozwiązań.