

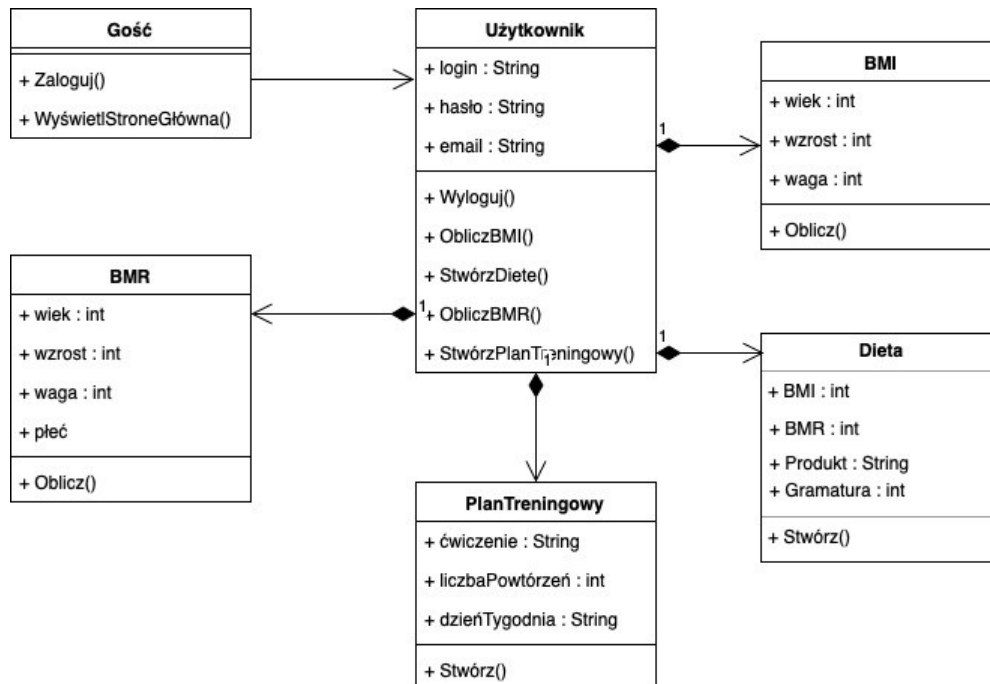
Opis projektu

Aplikacja webowa wykonana za pomocą Python Flask oraz React Redux, mająca na celu pomoc w planowaniu codziennego żywienia. Umożliwia śledzenie przez użytkownika wartości kalorycznej codziennych posiłków, historię tego jak wyglądały one w przeszłości, oraz algorytm proponujący ćwiczenia zależnie od nadwyżki kalorii w spożywanych posiłkach. Użytkownik może wprowadzić informacje o swoim ciele(wzrost, waga, wiek, wielkość mięśni, waga, którą chce osiągnąć), na podstawie których algorytm będzie mógł wyliczyć dietę i czas potrzebny do osiągnięcia założonych celów. Posiada ona bazę produktów i ćwiczeń aktualizowaną na bieżąco za pomocą skryptu zapisującego do bazy dane z pliku csv.

Wymagania funkcjonalne

- zaktualizowana baza produktów spożywczych
- zaktualizowana baza ćwiczeń treningowych
- możliwość dodawania własnych przepisów
- dodawanie spożytych każdego dnia produktów
- śledzenie historii, obliczanie spożycia kalorii
- lista posiłków, możliwość dodawania własnych
- algorytm podpowiadający plan ćwiczeń w zależności od spożytych kalorii
- kalkulator kalorii
- obliczanie zapotrzebowania (masa, wiek, wysiłek fizyczny)
- tworzenie przykładowych planów treningowych pod oczekiwania
- kalendarz postępu (spadek wagi)
- obrazowanie danych na wykresach(waga, mięśnie, osiągnięcia)
- wykorzystanie architektury REST

Diagram klas



Przypadki użycia

1. Logowanie

Warunki wstępne: inicjalizacja przez uruchomienie serwisu (otwarcie strony WWW)

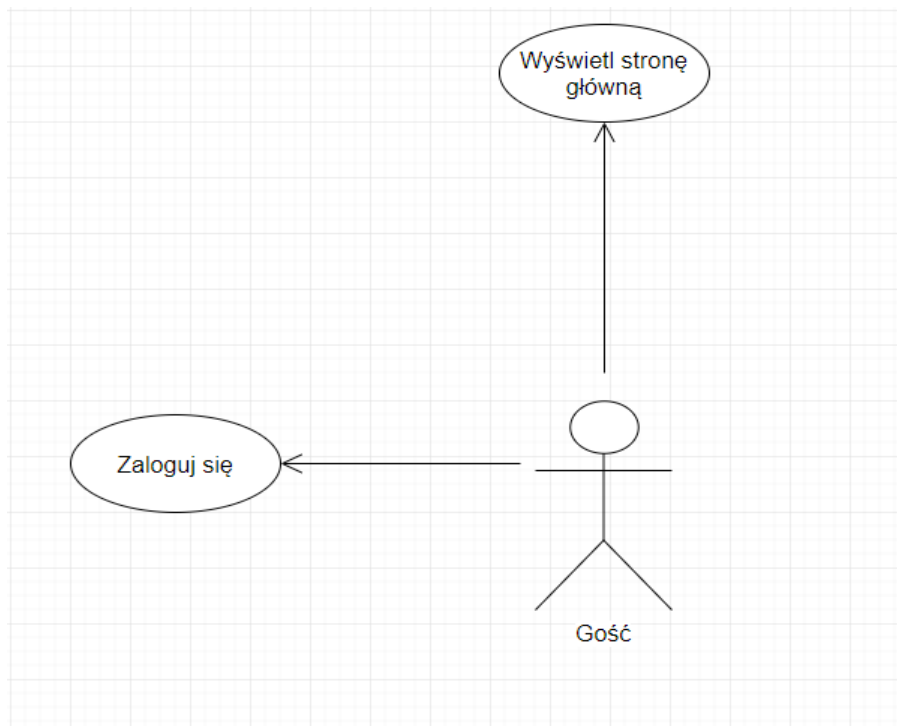
Warunki końcowe:

1. zalogowanie się
2. brak możliwości zalogowania - pozostanie na stronie głównej

Przebieg:

1. Wybranie przycisku logowania.
2. Wybranie metody logowania.
3. Zalogowanie się.
4. W przypadku poprawnego logowania przeniesienie na stronę profilową zalogowanego użytkownika.

W przeciwnym przypadku wyświetlenie informacji o braku możliwości zalogowania i pozostawienie użytkownika na stronie głównej.



2. Wyświetlenie profilu

Warunki wstępne: inicjalizacja przez uruchomienie serwisu (otwarcie strony WWW)

Warunki końcowe:

1. Wyświetlenie profilu.

Przebieg:

1. Należy zalogować się na swoje konto - wywołać przypadek użycia Logowanie.
2. Po udanym zalogowaniu nacisnąć przycisk służący do wyświetlenia profilu.

3. Wylogowanie

Warunki wstępne: inicjalizacja przez uruchomienie serwisu (otwarcie strony WWW)

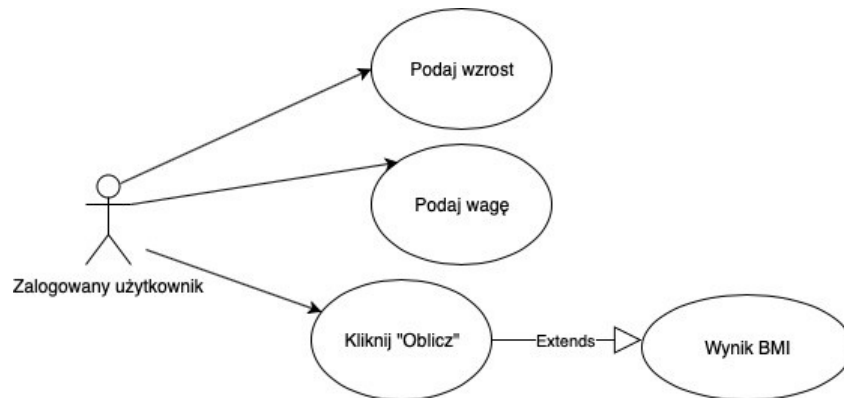
Warunki końcowe:

1. Poprawne wylogowanie się

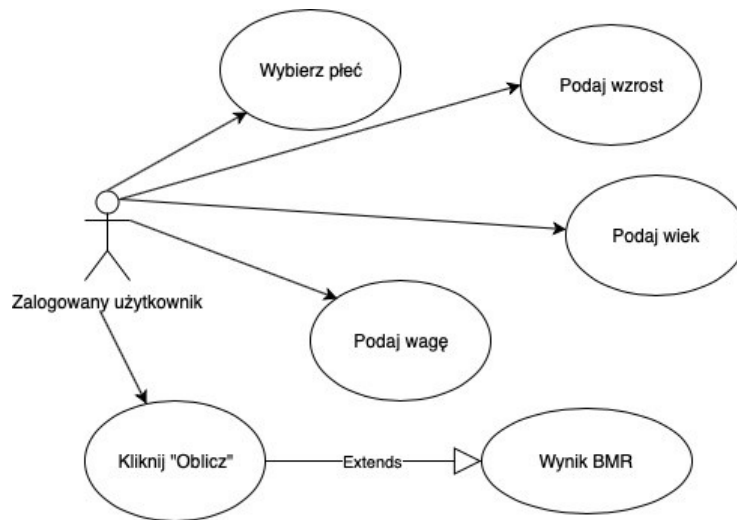
Przebieg:

1. Należy zalogować się na swoje konto - wywołać przypadek użycia Logowanie.
2. Po udanym zalogowaniu przejść do profilu - przypadek użycia Wyświetlanie profilu.
3. Należy nacisnąć przycisk wyloguj.

4. Obliczanie BMI



5. Obliczanie BMR



6. Podział zadań członków zespołu:

Artur Gołębiewski (backend):

- pierwsza iteracja obsługi użytkowników
- API obsługujące poszczególne produkty spożywcze
- modele poszczególnych produktów spożywczych
- pomoc przy dokumentacji

Mateusz Labuda (backend):

- dopracowanie obsługi użytkowników (role admin, user)
- API obsługujące plany dietetyczne oraz plany treningowe
- scraper produktów spożywczych + insert to db
- modele planów treningowych oraz dietetycznych
- refaktoryzacja struktury plików
- testy środowiska

Semen Bilikiewicz(frontend):

- utworzenie komponentów react
- kalkulator BMR i BMI
- wykonanie dokumentacji
- dodanie popupow informacyjnych

Wojciech Januszek (Frontend):

- obsługa rejestracji użytkowników
- logowanie użytkowników
- obsługa komunikacji z backendem
- architektura aplikacji React

Michał Gorzkowski

- podstawowa architektura projektu
- projekt strony głównej
- style aplikacji(niedokończone)
- tworzenie planu dietetycznego
- walidacja