

Dokumentacja projektu wykonywanego w ramach zajęć BAZY DANTCH I

Michał Ćwierz - 2021

I. Projekt koncepcji, założenia.

1. Tematem który obrałem w realizacji projektu to aplikacja do tworzenia planów dietetycznych. Jego celem jest pomoc w osiągnięciu pożądanej masy. Drugim ważnym celem jest ułatwienie procesu przygotowania posiłków.
2. Baza powinna przechować dane użytkownika, jego plany co do zmiany masy oraz pozwalać na przypisanie ułożonej diety. Baza powinna pozwalać na dodawanie przepisów przez użytkowników, z których można układać dzienne plany żywieniowe. Możliwe powinno być również dodawanie i grupowanie poszczególnych składników i sprawdzania ich cen u dostawców.
3. Obliczanie BMI, generowanie listy zakupów, sumowanie kalorii i sprawdzanie czy mieszczą się we właściwym zakresie.

II. Projekt diagramów (konceptualny)

4. Użytkownik steruje przepływem danych poprzez dodawanie przepisów. Przepisy są układane w dzienne plany, a dzienne plany mogą zostać wykorzystane w układaniu diety. Dieta może zostać przypisana do użytkownika. Przepisy składają się z poszczególnych kroków. Każdy przepis posiada kategorie. Przepis składa się ze składników. Składniki można znaleźć u dostawców. Składniki również posiadają swoją kategorię.

5. Poszczególne encje:

użytkownik: id diety, imię, nazwisko, email, hasło, wzrost, waga

dieta: zmiana wagi, czas na zmianę wagi

dieta – dzienny plan: numer dnia

dzienny plan

dzienny plan – przepis: pora dnia

przepis: id użytkownika, nazwa, poziom trudności, czas wykonania, wegańskie, kalorie, węglowodany, białko, tłuszcz

krok przepisu: czas wykonania, instrukcja

ocena: id użytkownika, id przepisu, gwiazdka, komentarz

przepis –kategoria

kategoria przepisu: nazwa

przepis – składnik: porcja

składnik: nazwa

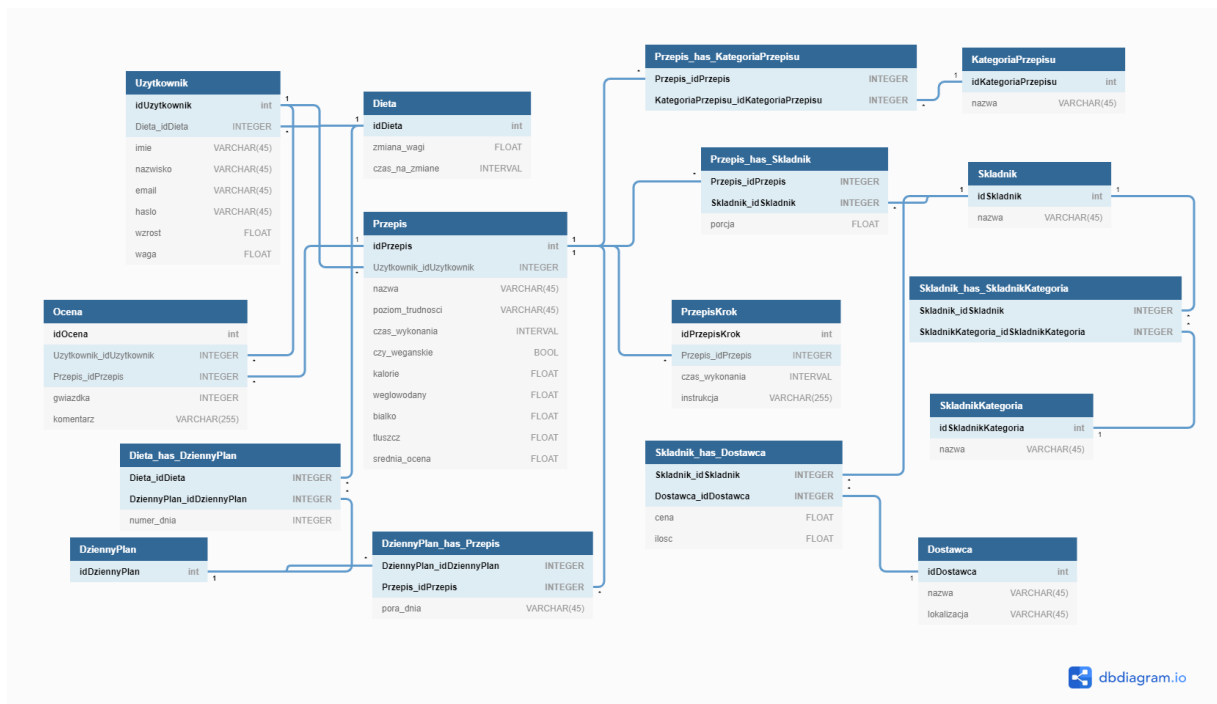
składnik – kategoria

kategoria składnika: nazwa

składnik –dostawca: cena, ilość

dostawca: nazwa, lokalizacja

6. Schemat ERD.



7. Projektowanie tabel, kluczy, indeksów.

Użytkownik steruje przepływem danych poprzez dodawanie przepisów. Przepisy są układane w dzienne plany, a dzienne plany mogą zostać wykorzystane w układaniu diety. Dieta może zostać przypisana do użytkownika. Przepisy składają się z poszczególnych kroków. Każdy przepis posiada kategorie. Przepis składa się z składników. Składniki można znaleźć u dostawców. Składniki również posiadają swoją kategorię. W pliku Fit_planer.sql jest tworzenie tabel wraz ze szczegółami.

8. Słowniki danych.

poziom trudności: „łatwy”, „średni”, „trudny”

Ograniczenia zawarte są w pliku Fit_planer.sql

9. Analiza zależności funkcyjnych i normalizacja tabel.

W finalnej wersji bazy pole BMI z tabeli użytkownik zostało usunięte i dodane do widoku użytkownik_stat. Również to samo stało się z polem średnia ocena w tabeli przepis przenosząc do przepis_stat. Pola sum w dziennym planie zostały usunięte i widoczne są teraz z widoku dziennyplan_stat.

10. Denormalizacja struktury tabel.

11. Zaprojektowanie operacji na danych.

Kwerendy zostały przedstawione w formie widoków zawartych w pliku widoki.txt

IV. Projekt funkcjonalny.

12. Wprowadzanie danych.

Każda tabela posiada funkcjonalność wyświetlania, dodawania, usuwania i modyfikowania rekordów.

13. Wizualizacja danych.

Forma generowania raportów została zawarta w głównym panelu aplikacji klienta podobnie jak inne tabele.

14. Zdefiniowanie panelu sterowania aplikacji.

Panel sterowania został utworzony za pomocą języka Python i jego biblioteki tkinter. Komunikacja z bazą danych odbywa się z pomocą biblioteki psycpg2.

15. Makropolecenia.

Dodaj rekord, usuń rekord, modyfikuj rekord, pokaż rekord.

V. Dokumentacja.

16. Wprowadzanie danych.

Dane wprowadzane są ręcznie w formie formularzy w GUI. Załączyłem również plik Insert.txt w którym przygotowałem inserty.

17. Dokumentacja użytkownika.

Do korzystania z aplikacji potrzebny jest Python3 oraz pobranie biblioteki pycpg2.

Do połączenia z bazą danych na pascalu potrzebny jest tunel. Ja robiłem to komendą:

```
ssh -L 5432:pascal.fis.agh.edu.pl:5432 8cwierz@pascal.fis.agh.edu.pl
```

Program uruchamiamy z poziomu folderu DatabaseClient metodą „python main.py”.

Aplikacja jest dość intuicyjna w obsłudze.

18. Opracowanie dokumentacji technicznej.

Connector – jest to singleton który odpowiada za połączenie z bazą danych, zawiera metody takie jak execute_insert, execute_delete... etc. które odpowiadają za wysyłanie kwerend.

MainComponent – składa się z klas która posiadają całą funkcjonalność potrzebną poszczególnym tabelom. Jest obiektem Frame biblioteki tkinter i pozwala na komunikację użytkownika z Connectorem.

AssociativeComponent – jest nieco zmodyfikowaną wersją MainComponent dla tablic asocjacyjnych z uwagi na brak domyślnego pola id.

RaportComponent – jest to zmodyfikowana wersja MainComponent pozwalająca tylko na wyświetlanie zawartości tabeli.

19. Wykaz literatury.

<https://docs.python.org/3/library/tkinter.html>

<https://www.pycpg.org/docs/>