

Changelog

Jest to pomocniczy rozdział opisujący zmiany w kolejnych wersjach pracy wysyłanej do promotora.

v2

- Uwzględniono sugestie z komentarzy:
 - poprawiłem błędy wyszczególnione błędy ortograficzne, interpunkcyjne, stylistyczne
 - W kwestii rozwiązań konkurencyjnych i pracy offline;napisałem maile do supportu wszystkich porównywanych rozwiązań z pytaniem czy jest możliwa praca offline. Wszyscy odpisali, że nie. (oczywiście poza aliantem, który działa tylko offline)
 - dodałem zgodę na przetwarzanie danych osobowych w wymaganiach funkcjonalnych i use casach
 - diagramy przypadków użycia powiększyłem o ok 30% a klas o ok 10%
 - dodałem reguły w latex'ie które powinny wyeliminować większość wdów i sierot
 - dodałem wersje przeglądarek/systemów w wymaganiach niefunkcjonalnych
- Zmiany:
 - krótkie streszczenie na stronie tytułowej
 - przegląd rozwiązań konkurencyjnych
 - uzupełniłem opis problemu we wstępie
 - założenia projektowe: uzupełniłem słownik
 - diagramy przypadków użycia przerzuciłem do sekcji z wymaganiami funkcjonalnymi w założeniach projektowych
 - poprawiłem scope wymagań funkcjonalnych
 - uprościłem diagramy przypadków użycia
 - uzupełniłem prototypy interfejsu
 - w dodatku (jdl) zamiast całego opisu domeny zostawiłem tylko definicję mikroserwiów. odniesienie do dodatku będzie w rozdziale implementacja
 - kategorie, ograniczenia, reguły i diagramy klas zostały rozdzielone na sekcje dla każdej poddziedziny w celu zwiększenia czytelności
 - wyrzuciłem ze scope'u mało istotne kategorie; głównie z poddziedzin administracyjnej i wizyt
 - zmiana oznaczeń kategorii/graniczeń/reguł z KAT/XXX na KAT/Y/XX, gdzie Y to numer poddziedziny
- Nowości:
 - stan wiedzy i technik: przegląd literatury dietetycznej, opis architektury mikroserwisów
 - projekt: opis podstawowej architektury systemu
 - projekt: projekt bazy danych - wprowadzenie teoretyczne
 - implementacja: instalacja oprogramowania
- Podsumowanie:
 - Chciałbym zamknąć już wszystko od początku pracy do końca opisu projektu (tj. do końca rozdziału 3.)
 - Właściwej treści w pracy jest więcej, a stron mniej niż w poprzednio przesłanej wersji

- Do zrobienia:
 - uzupełnić wykorzystywane środowiska i narzędzia programistyczne
 - uzupełnić opisy mockupów
 - opisać: architektura systemu, prezentacja aplikacji, dokumentacja kodu, testy, podsumowanie



Politechnika Wrocławska

Wydział Informatyki i Zarządzania
Kierunek studiów: Informatyka

Praca dyplomowa – inżynierska

PROJEKT I IMPLEMENTACJA SYSTEMU DO ZARZĄDZANIA DIETĄ W OPARCIU O ARCHITEKTURĘ MIKROSERWISÓW

Krzysztof Marczyński

słowa kluczowe:
układanie jadłospisów
architektura mikroserwisów
aplikacja webowa

krótkie streszczenie:

Praca przedstawia proces projektowania i implementacji systemu do zarządzania dietą z wykorzystaniem architektury mikroserwisów. Omówiona zostanie wiedza wynikająca z omówienia rozwiązań konkurencyjnych, przeglądu literatury domenowej, a także z przeglądu stylów architektury oprogramowania. Zasadniczym problemem rozwiązywanym przez tworzony produkt jest układanie jadłospisów przez dietetyka i przydzielanie ich pacjentowi.

Opiekun pracy dyplomowej	dr inż. Michał Szczepanik
	Tytuł/stopień naukowy/imię i nazwisko	ocena	podpis
Ostateczna ocena za pracę dyplomową			
Przewodniczący Komisji egzaminu dyplomowego
	Tytuł/stopień naukowy/imię i nazwisko	ocena	podpis

*Do celów archiwalnych pracę dyplomową zakwalifikowano do:**

- a) kategorii A (akta wieczyste)
- b) kategorii BE 50 (po 50 latach podlegające ekspertyzie)

* niepotrzebne skreślić

pieczętka wydziałowa

Wrocław, rok 2019

Streszczenie

Celem pracy było opracowanie systemu do zarządzania dietą w architekturze mikroserwisów. Aby osiągnąć ten cel przeprowadzono analizę istniejących rozwiązań konkurencyjnych, przedstawiono niezbędną wiedzę domenową oraz porównano popularne style architektury aplikacji. Na podstawie zgromadzonej wiedzy wyszczególniono niezbędne założenia projektowe, zaprojektowano interfejs oraz zdefiniowano kategorie danych wraz z regułami i ograniczeniami ich dotyczącymi. Następnie przedstawiono opis implementacji stworzonej na podstawie opracowanego projektu. W implementacji kluczową rolę odegrały języki Java i TypeScript, platforma deweloperska JHipster oraz stos technologii Netflix OSS dla architektury mikroserwisów. Stworzone rozwiązanie może zostać wykorzystane przez dietetyków w celu przeprowadzania kompleksowej obsługi wizyty pacjenta z położeniem szczególnego nacisku na układanie jadłospisów i udostępnianie go pacjentom.

Abstract

The aim of this work was to develop a diet management system based on microservice architecture. To achieve that goal, an analysis of existing competitive solutions was performed, the necessary domain knowledge was presented, and popular application architecture styles were compared. Based on the accumulated knowledge, the necessary design assumptions were specified, the interface was designed, and categories of data were defined along with the rules and restrictions concerning them. Then a description of the implementation based on the developed project was presented. The key role in the implementation was played by languages Java and TypeScript, the JHipster development platform and Netflix OSS technology stack for a microservices architecture. The created solution can be used by dietitians in order to conduct comprehensive service of the patient's visit with particular emphasis on designing the meal plans and sharing them with patients.

Spis treści

Wstęp	1
Opis problemu	1
Cel pracy	2
Zakres pracy	2
1. Stan wiedzy i techniki w zakresie tematyki pracy	3
1.1. Przegląd istniejących rozwiązań konkurencyjnych	3
1.2. Przegląd literatury dietetycznej	6
1.3. Architektura mikroserwisów	8
2. Założenia projektowe	10
2.1. Uwagi wstępne	10
2.2. Słownik pojęć domenowych	10
2.3. Sformułowanie problemu	11
2.4. Pozycjonowanie produktu	12
2.5. Dekompozycja problemu w oparciu o poddziedziny	12
2.6. Podsumowanie użytkowników systemu	12
2.7. Wymagania funkcjonalne	14
2.8. Wymagania niefunkcjonalne	24
3. Projekt	25
3.1. Prototyp interfejsu	25
3.1.1. Poddziedzina administracyjna	26
3.1.2. Poddziedzina produkty	30
3.1.3. Poddziedzina przepisy	31
3.1.4. Poddziedzina jadłospisy	33
3.1.5. Poddziedzina wizyty	34
3.2. Opis podstawowej architektury systemu	39
3.3. Projekt bazy danych	39
3.3.1. Ogólne cechy dziedziny	40
3.3.2. Poddziedzina administracyjna	41
3.3.3. Poddziedzina produkty	44
3.3.4. Poddziedzina przepisy	52
3.3.5. Poddziedzina jadłospisy	59
3.3.6. Poddziedzina wizyty	64
4. Implementacja	70
4.1. Wykorzystywane środowiska i narzędzia programistyczne	70
4.2. Architektura systemu	70
4.3. Instalacja oprogramowania	71

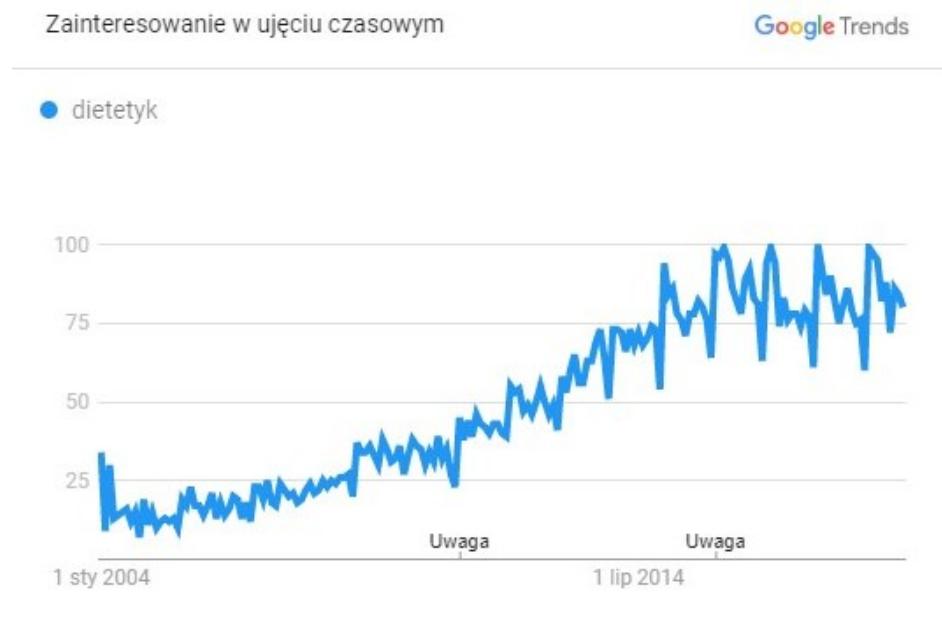
4.3.1. Wymagania wstępne	71
4.3.2. Instalacja	71
4.4. Prezentacja aplikacji	72
4.5. Dokumentacja kodu	72
4.6. Testy	72
Zakończenie	73
4.7. Podsumowanie pracy	73
4.8. Możliwości dalszego rozwoju	73
Bibliografia	75
Spis rysunków	77
Spis tabel	79
Spis kodów źródłowych	80
Dodatek A. JDL	81

Wstęp

Opis problemu

Zanim będzie można rozpocząć analizę problemu, warto zdefiniować co można rozumieć pod pojęciem diety. Według definicji z Encyklopedii Powszechnej PWN, dieta to "system odżywiania z ustaleniem jakości i ilości pokarmów, dostosowany do potrzeb organizmu"[15]. Opierając się na tej definicji, można opisać system do zarządzania dietą jako system, który będzie ułatwiał dietetykowi tworzenie jadłospisów dostosowanych do potrzeb żywieniowych organizmu pacjenta, zarządzanie stworzonymi jadłospisami oraz pozwalać udostępniać utworzony jadłospis pacjentowi w formie umożliwiającej w jak najprostszy sposób zastosowanie przez pacjenta przygotowanej dla niego diety.

Obserwując trendy występujące we współczesnym społeczeństwie można zauważać, że zdrowy styl życia stał się modny, a czasem nawet utożsamiany ze statusem społecznym. W związku z tą tendencją coraz więcej osób regularnie uprawia sport, rezygnuje z używek, a także dba o dietę. Analizując przedstawionek na rysunku 1 dane dotyczące wyszukiwania hasła "dietetyk" w latach 2004-2019 poprzez narzędzie Google Trends[10] można zauważać, że popularność wyszukiwania tego hasła w latach 2016-2019 jest ponad 5 krotnie większa niż w roku 2004.



Polska. Od 01.01.2004 do 12.09.2019. Wyszukiwarka Google.

Rys. 1: Zainteresowanie hasłem "dietetyk" w ujęciu czasowym [10]

Zwiększone zainteresowanie usługami dietetycznymi powoduje zwiększone zapotrzebowanie na wysokiej jakości, nowoczesne narzędzia wspomagające pracę dietetyka. W chwili pisania niniejszej pracy w Polsce popularność zyskało jedynie kilka programów oferujących kompleksowe funkcjonalności potrzebne w codziennej praktyce dietetyka, można więc sądzić, że rynek aplikacji tego typu nie został jeszcze nasycony.

Biorąc pod uwagę powyższe spostrzeżenia, warto w tym miejscu podkreślić, że Światowa Organizacja Zdrowia określiła otyłość (i bardziej ogólnie choroby dietozależne) jako jeden z głównych problemów zdrowia publicznego[32]. Fakt ten podkreśla jak duże brzemię odpowiedzialności spoczywa na ramionach dietetyków, a co za tym idzie jak istotne jest dostarczenie specjalistom dietetyki narzędzi ułatwiających niesienie pomocy pacjentom.

Cel pracy

Celem pracy jest projekt i budowa platformy do zarządzania dietą w oparciu o architekturę mikroserwisów. Tworzona platforma będzie obejmowała cały proces zarządzania dietą, czyli przede wszystkim: zebranie przez dietetyka wywiadu żywieniowego od pacjenta, stworzenie przez dietetyka jadłospisu, udostępnienie jadłospisu pacjentowi i ułatwienie pacjentowi stosowania diety poprzez generowanie listy zakupów.

Zakres pracy

Praca w swoim zakresie zawiera opracowanie projektu systemu, w ramach którego, między innymi, przygotowane zostaną diagramy UML takie jak diagram przypadków użycia, diagram klas i diagram rozmieszczenia. Przygotowana zostanie również implementacja w oparciu o języki Java i TypeScript oraz o stos technologii Netflix OSS dla architektury mikroserwisów. Na koniec zostanie przedstawiona dokumentacja kodu oraz pokrótkę omówiony zostanie sposób instalacji i korzystania z systemu.

1. Stan wiedzy i techniki w zakresie tematyki pracy

1.1. Przegląd istniejących rozwiązań konkurencyjnych

Na rynku Polskim funkcjonuje zaledwie kilka narzędzi wspomagających w kompleksowy sposób pracę dietetyka. W niniejszej sekcji zostaną omówione systemy cieszące się największą popularnością oraz przedstawione zostanie zbiórce porównanie ich najważniejszych cech. W przypadku większości z porównywanych programów warunki korzystania z usługi pozwalają na rejestrację w systemie jedynie wykwalifikowanym dietetykom. Z tego względu wszystkie analizowane dane zostały zebrane z publicznie dostępnych źródeł, takich jak strony odpowiednich programów, blogi internetowe, filmy promocyjne na platformie YouTube[11]. Dodatkowo, żeby rozwiązać wątpliwości czy dane rozwiązanie konkurencyjne umożliwia korzystanie z kluczowych funkcjonalności bez dostępu do internetu, przeprowadzono stosowną korespondencję z obsługą klienta poszczególnych programów.

- **TiqDiet**

TiqDiet[26] jest wygodnym w użyciu programem, który pozwala dietetykom w prosty sposób tworzyć jadłospisy i udostępniać je pacjentom. W celu uproszczenia pracy dietetyka, dostępnych jest wiele szablonów jadłospisów, które łatwo można dostosować do indywidualnych potrzeb pacjenta. Pacjent może odbierać ułożoną dietę poprzez responsywną stronę internetową, aplikację mobilną oraz za pomocą inteligentnego zegarka. Aplikacje mobilne pozwalają ponadto na automatyczne przypominanie pacjentowi m.in. o konieczności zażycia suplementu oraz o konieczności regularnego picia wody. Komunikacja pomiędzy pacjentem, a dietetykiem może odbywać się w czasie rzeczywistym za pomocą zintegrowanego chatu. Ponadto dietetyk ma możliwość obserwowania postępów pacjentów w stosowaniu diety, a w razie potrzeby może zalecać wizytę u lekarza czy też zażycie dodatkowych suplementów.

- **Kcalmar PRO**

Kcalmar[12] jest systemem, którego głównym założeniem jest maksymalne skrócenie czasu potrzebnego na stworzenie programu żywieniowego dopasowanego do potrzeb pacjenta. Zapewnia zaawansowany system podpowiedzi ułatwiający projektowanie zbilansowanej diety z wyraźnym oznaczeniem alergenów czy zduplikowanych potraw. Jadłospisy mogą być automatycznie skalowane z automatycznym przeliczeniem miar domowych. Co ciekawe system pozwala również na wyszukiwanie dietetyków w wybranych miastach i filtrowanie ich według typów diet i jednostek chorobowych, w których się specjalizują.

- **Dietetyk Pro**

Program Dietetyk Pro[2] na tle konkurencji wyróżnia się tym, że poza główną funkcjonalnością układania jadłospisu, abonenci mogą również korzystać ze szkoleń eksperckich i literatury dietetycznej dostępnej w ramach platformy. Dodatkowo dietetycy po wykupieniu subskrypcji mogą skorzystać ze zdalnej pomocy z obsługi programu.

Ciekawym udogodnieniem jest możliwość szerokiej konfiguracji ekranu startowego, np poprzez dodanie kalkulatora wartości odżywczych czy też wyświetlanie listy zaplanowanych wizyt. Spośród porównywanych programów Dietetyk Pro posiada największe bazy produktów i przepisów wyprzedzając konkurencję niemal dwukrotnie.

- **Aliant**

Do grupy klientów docelowych programu Aliant[1] należą zarówno dietetycy jak również trenerzy personalni. Tak jak inne porównywane aplikacje, główną funkcjonalnością programu Aliant jest układanie jadłospisów, jednakże w przeciwieństwie do konkurencji, aplikacja jest dostępna tylko jako aplikacja na platformę Windows. Nie ma możliwości dostępu do systemu przez stronę internetową, jednak pozwala na automatyczne dodawanie z internetu zewnętrznych baz produktów po zaakceptowaniu ich licencji wykorzystania. Brakuje również zintegrowanego narzędzia do komunikacji z pacjentami, udostępnianie ułożonego jadłospisu musi odbywać się w całości poza systemem.

- **Dietico**

Program Dietico[33] szczerzy się stale powiększaną bazą przepisów i produktów. Pozwala na układanie jadłospisu za pomocą wygodnego interfejsu, a przejrzysty system podpowiedzi pozwala szybko wykryć powtarzające się dania oraz produkty na które pacjent jest uczulony. Twórcy programu szczerzą się możliwością uwzględnienia w układanej diecie posiadanego przez pacjenta wyposażenia kuchennego i sezonowych produktów spożywczych

- **Vitme**

Program Vitme[28] umożliwia prowadzenie kart pacjentów, projektowanie jadłospisów oraz generowanie wydruków w PDF. Program w porównaniu z konkurencją jest oferowany w bardzo korzystnej cenie oraz posiada bogatą bazę produktów, jednak stosunkowo niewielką bazę przepisów. Do wad produktu można zaliczyć przestarzały i mało przejrzysty interfejs, który może zniechęcić niektórych potencjalnych klientów.

W tabeli 1.1 przedstawiono porównanie najważniejszych cech funkcjonalnych, a w tabeli 1.2 cech niefunkcjonalnych 6 istniejących na rynku rozwiązań konkurencyjnych[23]. Warto zwrócić uwagę, że funkcjonalności takie jak możliwość wykorzystania gotowych szablonów diet, wysyłanie diety do pacjenta, przeprowadzanie wywiadu żywieniowego czy automatyczne generowanie listy zakupów nie występują w niektórych spośród analizowanych systemów. Na tej podstawie można wysunąć hipotezę, że te funkcjonalności - mimo iż istotne - nie są kluczowe w systemie wspomagającym pracę dietetyka. Natomiast możliwość tworzenia jadłospisów z wykorzystaniem własnych produktów i przepisów, zapisywanie ich do plików oraz przypisywanie do stosownych kart pacjenta są oczekiwane w tego typu aplikacji.

Tabela 1.1: Rozwiązania konkurencyjne - cechy funkcjonalne (opr.wł)

	TiqDiet	Kcalmar Pro	Dietetyk Pro	Aliant	Dietico	Vitme
Tworzenie jadłospisów	TAK	TAK	TAK	TAK	TAK	TAK
Gotowe szablony diet	TAK	TAK	TAK	TAK	NIE	NIE
Zapis diety do pliku	TAK	TAK	TAK	TAK	TAK	TAK
Wysyłanie diet do pacjenta	TAK	TAK	TAK	TAK	NIE	TAK
Komunikacja z pacjentem	TAK	TAK	TAK	NIE	NIE	TAK
Karta pacjenta	TAK	TAK	TAK	TAK	TAK	TAK
Wywiad żywieniowy	TAK	TAK	TAK	NIE	TAK	TAK
Lista zakupów	TAK	TAK	TAK	TAK	TAK	NIE
Dodawanie własnych produktów	TAK	TAK	TAK	TAK	TAK	TAK
Dodawanie własnych przepisów	TAK	TAK	TAK	TAK	TAK	TAK

Tabela 1.2: Rozwiązań konkurencyjne - cechy niefunkcjonalne (opr.wł)

	TiqDiet	Kcalmar Pro	Dietetyk Pro	Aliant	Dietico	Vitme
Liczba produktów w bazie	1000	1400	6000	3500	900	5000
Liczba gotowych przepisów	200	800	2800	1700	1900	400
Praca offline	NIE	NIE	NIE	TAK	NIE	NIE
Praca online	TAK	TAK	TAK	NIE	TAK	TAK
Aplikacja mobilna dla dietetyka	TAK	TAK	TAK	NIE	NIE	NIE
Aplikacja mobilna dla pacjenta	TAK	TAK	NIE	NIE	NIE	TAK
Dostęp dla pacjenta przez przeglądarkę internetową	TAK	TAK	TAK	NIE	NIE	TAK
Darmowy okres testowy	14dni	14dni	7dni	bezterminowo	14dni	14dni
Cena w abonamencie rocznym	199	1188	246	699	546	219

1.2. Przegląd literatury dietetycznej

W rozdziale 1.1 dokonano przeglądu rozwiązań konkurencyjnych. Na podstawie dokonanej analizy możliwe będzie zdefiniowanie głównych wymagań funkcjonalnych projektowanego systemu, jednakże konieczne jest odwołanie się do literatury dziedzinowej, żeby potwierdzić zasadność przyjętych założeń istotnych z punktu widzenia dietetyki.

Pierwszym rozważanym pojęciem jest podstawowa przemiana materii (PPM). Jest to poziom zapotrzebowania energetycznego organizmu znajdującego się w stanie spoczynku (czyli minimalny poziom zapotrzebowania energetycznego) wyznaczany na podstawie wieku i masy ciała osoby. Aby obliczyć wartość energetyczną posiłku należy wyznaczyć ekwiwalent metaboliczny (MET) podstawowych wartości odżywczych, tj. białek, tłuszczy i węglowodanów[3].

Kolejnym uwzględnianym współczynnikiem jest współczynnik poziomu aktywności fizycznej (ang. Physical Activity Level - PAL). Organizacja Narodów Zjednoczonych do spraw Wyżywienia i Rolnictwa definiuje 5 poziomów aktywności fizycznej[5]:

- brak aktywności fizycznej (wartość współczynnika 1,2 - 1,39)
- niska aktywność fizyczna (wartość współczynnika 1,4 - 1,69)
- umiarkowana aktywność fizyczna (wartość współczynnika 1,7 - 1,99)
- wysoka aktywność fizyczna (wartość współczynnika 2 - 2,4)
- bardzo wysoka aktywność fizyczna (wartość współczynnika > 2,4)

Iloczyn PPM i PAL określa stopień całkowitej przemiany materii (CPM)[14]. Przy czym dla prawidłowo zaplanowanej diety, dzienna energia powinna być dostarczana:

- ok 10% z białek
- ok 60% z węglowodanów
- ok 30% z tłuszczy

Wymienione wyżej składniki tworzą najważniejszą grupę składników, tzw grupę składników energetycznych, ale należy zwrócić uwagę, że do prawidłowego funkcjonowania organizmu definiuje się ok 40 innych niezbędnych składników należących do grup składników budulcowych (głównie jod, wapń, lipidy, fosfor, żelazo i siarka) i regulujących (głównie witaminy, błonnik oraz mikro- i makroelementy), które również powinny być uwzględnione podczas układania zbilansowanej diety[3].

Podczas układania jadłospisu uwzględnia się podstawowe typy diet[8]:

- podstawowa
- kleikowa
- papkowata
- bogatopotasowa
- niskosodowa
- niskocholesterolowa
- bezglutenowa
- bogatoresztkowa
- łatwostrawna
- ubogotłuszczowa
- bogatobiałkowa
- ubogobiałkowa
- bogatoenergetyczna
- ubogoenergetyczna

Jak zauważono wyżej, podstawowym kryterium potrzebnym do skomponowania odpowiednio zbilansowanej diety jest odpowiednie dobranie wartości odżywczej produktów spożywczych. Wartości te mogą być uzyskane z tabel składu i wartości odżywczej żywności. Na rynku polskim tabele takie są odpłatnie udostępniane przez polski Instytut Żywności i Żywienia (IŻŻ)[16], jednakże licencja wspomnianego opracowania nie zezwala na wykorzystanie danych zawartych w zestawieniu bez wykupienia odpowiedniego abonamentu[13].

Podobne rozwiązanie w języku angielskim oferuje Departament Rolnictwa Stanów Zjednoczonych (ang. United States Department of Agriculture - USDA) udostępniając całkowicie za darmo do dowolnego użytku narodową bazę danych wartości odżywczych dla standardowych odwołań (ang. National Nutrient Database for Standard Reference)[30] potocznie nazywana "bazą USDA". Dane są udostępniane w formie pliku bazy Microsoft Access. Baza zawiera:

- ponad 7 tysięcy produktów spożywczych
- ponad 600 tysięcy wartości odżywczych
- ok 100 definicji wartości odżywczych
- ponad 14 tysięcy miar domowych
- ok 25 kategorii produktów spożywczych

Na koniec nalażałyby rozważyć wskaźniki pozwalające ocenić wpływ diety. Podstawowe wyznaczane wartości to wskaźnik masy ciała (ang. Body Mass Index - BMI), stosunek obwodu talii do obwodu bioder (ang. Waist to Hip Ratio - WHR) oraz ilość tkanki tłuszczowej w organizmie. W celu wyznaczenia ilości tkanki tłuszczowej w organizmie przeprowadza się pomiar fałdów skórno-tłuszczowych lub stosuje się bioelektryczną metodę impedancji (ang. bioelectrical impedance analysis - BIA). Metoda BIA polega na przepuszczeniu przez organizm ładunku elektrycznego o natężeniu poniżej 1mA. Metoda ta jest nieinwazyjna i bezpieczne jest jej stosowanie w każdym stanie zdrowia pacjenta. Pozwala nie tylko na wyznaczenie ilości tkanki tłuszczowej, ale także ilości wody w organizmie, wapnia w kościach, a także masy mięśniowej[3].

1.3. Architektura mikroserwisów

Tworzenie złożonych systemów informatycznych umożliwiających bezproblemowe korzystanie jednocześnie przez miliony użytkowników jest zadaniem niebanalnym. W klasycznym podejściu, implementowano systemy w architekturze monolitycznej. Aplikacja napisana w takiej architekturze jest samowystarczalna w kontekście jej zachowania. Może komunikować się z zewnętrznymi usługami lub źródłami danych w celu wykonania operacji, ale logika biznesowa potrzebna do wykonania każdej operacji jest w całości zawarta w obrębie aplikacji. W przypadku wystąpienia potrzeby skalowania horyzontalnego takiej aplikacji, konieczne jest powielanie całej aplikacji na każdym z serwerów[18].

Architektura mikroserwisów, zgodnie z tym co sugeruje nazwa, skupia się na budowaniu aplikacji będącej zbiorem niewielkich, luźno powiązanych serwisów komunikujących się ze sobą na przykład za pomocą protokołu HTTP czy AMQP. Serwisy implementowane i wdrażane są niezależnie od siebie[29]. Efektem tworzenia niezależnych serwisów jest skalowanie tylko serwisów, które tego wymagają, co pozwala na optymalne wykorzystanie zasobów[25].

Architektura monolityczna ma wiele zalet[24], spośród których do najważniejszych należą:

- Prostota implementacji
- Możliwość łatwego przeprowadzania radykalnych zmian w programie
- Prostota testowania
- Prostota wdrażania aplikacji na środowisko produkcyjne
- Prostota skalowania aplikacji

Martin Fowler podkreśla, że w przypadku wielu aplikacji architektura monolityczna jest jak najbardziej wystarczająca, a w przypadku gdy system jest wystarczająco złożony, żeby użycie mikroserwisów przyniosło realny zysk zwykle lepiej jest zacząć od monolitu, a następnie przeprowadzić migrację do architektury mikroserwisów poprzez wydzielanie modułów w obrębie monolitu i późniejsze przekształcanie ich w niezależne serwisy[6].

W przypadku aplikacji monolitycznej łatwo jest doprowadzić do sytuacji w której poszczególne moduły są ze sobą ściśle powiązane, co zwykle ma bardzo negatywny wpływ na wydajność aplikacji, utrudnia wprowadzanie zmian w kodzie i prowadzi do występowania trudnych do wykrycia błędów w implementacji. Sytuację w której w aplikacji powstaje dużo przypadkowych powiązań i zależności Vaughn Vernon określił mianem "Wielkiej Kuli Błota"[31].

Do głównych zalet zastosowania mikroserwisów należy stosowanie luźnego powiązania serwisów, co poniekąd wymusza, żeby zależności pomiędzy serwisami były bardziej przemyślane i lepiej zaprojektowane. Z pomocą we właściwym zaprojektowaniu serwisów i zależności między nimi przychodzi strategiczne wzorce DDD. Jednym z takich wzorców jest dekompozycja w oparciu o poddziedziny[4]. Dziedzina systemu jest dzielona na poddziedziny poprzez zdefiniowanie przestrzeni problemów biznesowych w obrębie względnie niezależnych obszarów specjalizacji. W przypadku architektury mikroserwisów można wyznaczyć poszczególne serwisy poprzez zdefiniowanie poddziedzin systemu i stworzenie serwisu dla każdej z nich[24].

2. Założenia projektowe

2.1. Uwagi wstępne

W niniejszym rozdziale opisano wizję systemu, który będzie wspomagał układanie diety.

Zalogowani dietetycy będą mogli zarządzać produktami, ich wartościami odżywczymi oraz miarami domowymi. Korzystając ze stworzonych produktów dietetycy będą mogli tworzyć przepisy, a następnie, w ramach jadłospisu, dodawać do planów posiłków przepisy i pojedyncze produkty.

Dietetycy będą mogli również zarządzać pacjentami i ich wizytami. W ramach wizyty dietetyk będzie mógł przeprowadzić wywiad żywieniowy, zebrać pomiary ciała pacjenta i przydzielić pacjentowi jadłospis.

2.2. Słownik pojęć domenowych

Na podstawie rozważań z rozdziału 1 sporządzono następującą listę definicji istotną z punktu widzenia projektu:

- Administrator
 - użytkownik posiadający uprawnienia do zarządzania uprawnieniami użytkowników
- BIA
 - metoda impedancji bioelektrycznej wykorzystywana do analizy składu ciała
- BMI
 - wskaźnik masy ciała
- CPM
 - całkowita przemiana materii
- Dieta
 - sposób odżywiania
- Dietetyk
 - specjalista w dziedzinie dietetyki
- Jadłospis
 - plan posiłków zdefiniowany na określoną liczbę dni z uwzględnieniem określonych wymagań
 - karta przedstawiająca przebieg współpracy dietetyka z pacjentem
- Karta pacjenta
 - ekwiwalent metaboliczny
- MET
 - definicja pospolitej miary, takiej jak np. łyżeczka w gramach
- Miara domowa
 - klient dietetyka
- PAL
 - współczynnik aktywności fizycznej
- Podstawowe wartości odżywcze
 - energia, białko, tłuszcze, węglowodany
- Pomiary ciała
 - pomiary ciała pacjenta przeprowadzane przez dietetyka
- Posiłek
 - posiłek jest przydzielany do jadłospisu; zawiera produkty i przepisy
- PPM
 - podstawowa przemiana materii

- Produkt
 - produkt spożywczy, dla którego specyfikowane są wartości odżywcze i miary domowe
- Przepis
 - opis składników i kroków przygotowania dania
 - semantyczny podział przepisu, np. sernik może mieć sekcje związane z przygotowaniem ciasta, nadzienia i polewy
- Sekcja przepisu
- USDA
- Wartość odżywczna
- Wizyta
- Wywiad żywieniowy
 - konkretna wizyta pacjenta
 - wywiad przeprowadzany z pacjentem uwzględniający jego nawyki żywieniowe, nietolerancje, choroby, przyjmowane leki, itp.

2.3. Sformułowanie problemu

W tabeli 2.1 przedstawiono sformułowanie rozważanego w pracy problemu wraz z jego wpływem i propozycją pomyślnego rozwiązania.

Tabela 2.1: Sformułowanie problemu (opr.wł)

Problem	Problem z ręcznym układaniem jadłospisu
Dotyczy	Dietetyków
Wpływ problemu	<ul style="list-style-type: none"> • Dietetyk poświęca dużo czasu na wyszukiwanie informacji o każdym produkcie, którego potrzebuje wykorzystać w układanym jadłospisie • Dietetyk poświęca dużo czasu na obliczanie wartości odżywcznych w każdym przepisie • Dietetyk poświęca dużo czasu na obliczanie wartości odżywcznych w każdym jadłospisie. • Dietetyk ma problem z przeliczeniem miar domowych produktów na gramy
Pomyślne rozwiązanie	<ul style="list-style-type: none"> • Będzie zwalniało dietetyka z konieczności obliczania wartości odżywcznych dla przepisów i jadłospisów • Będzie ułatwiało dietetykowi przekazywanie stworzonego jadłospisu pacjentów

2.4. Pozycjonowanie produktu

W tabeli 2.2 przedstawiono pozycjonowanie opracowywanego produktu względem rynku produktów wspomagających pracę dietetyka.

Tabela 2.2: Pozycjonowanie produktu (opr.wł)

Dla	Dietetyka
Który	Chce łatwiej zarządzać dietą
Nazwa produktu	Webowa aplikacja wspomagająca układanie jadłospisu
Który	Skraca czas potrzebny na ułożenie i zarządzanie jadłospisami
Inaczej niż	Kalkulator kalorii
Nasz produkt	Skupia się na tworzeniu i udostępnianiu jadłospisów

2.5. Dekompozycja problemu w oparciu o poddziedziny

Na podstawie wywiadu z dietetykiem, analizy rozwiązań konkurencyjnych oraz opierając się na opisanym w rozdziale 1.3 wzorcu dekompozycji problemu w oparciu o poddziedziny dla omawianej aplikacji wspomagania zarządzania dietą można wyszczególnić następujące poddziedziny:

- poddziedzina administracyjna - służąca jako brama aplikacji, pozwalająca na zarządzanie użytkownikami i administrowanie aplikacją
- poddziedzina produkty - skupiąca się na zarządzaniu produktami spożywczymi, ich wartościami odżywczymi i miarami domowymi
- poddziedzina przepisy - pozwalająca na tworzenie i zarządzanie przepisami, w tym przypisywanie do przepisów produktów
- poddziedzina jadłospisy - pozwalająca na tworzenie i zarządzanie jadłospisami, w tym przypisywanie do jadłospisów produktów i przepisów
- poddziedzina wizyty - skupiąca się na całkowitym zarządzaniu wizytami pacjenta w obrębie karty pacjenta, a w szczególności przypisywaniem do wizyty jadłospisów, przeprowadzaniem wywiadu żywieniowego czy też zbierania pomiarów ciała pacjenta

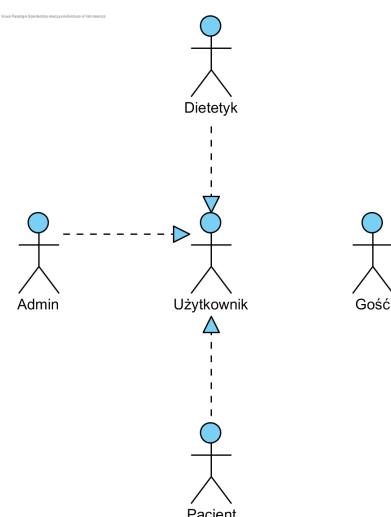
2.6. Podsumowanie użytkowników systemu

W tabeli 2.3 przedstawiono podsumowanie użytkowników projektowanego ich krótki opis oraz ich podstawowe odpowiedzialności związane z korzystaniem z systemu.

Tabela 2.3: Użytkownicy (opr.wł)

Nazwa	Opis	Odpowiedzialności
Gość	Niezalogowany użytkownik	<ul style="list-style-type: none"> • Zakłada konto użytkownika • Wyświetla stronę główną
Pacjent	Klient dietetyka	<ul style="list-style-type: none"> • Otrzymuje ułożony jadłospis
Dietetyk	Specjalista w dziedzinie dietetyki	<ul style="list-style-type: none"> • Używa założonego konta • Wprowadza, edytuje i usuwa produkty, przepisy i jadłospisy
Administrator	Osoba zarządzająca działaniem aplikacji	<ul style="list-style-type: none"> • Przydzielanie i odbieranie użytkownikom uprawnień • Zarządzanie definicjami wartości odżywczych, typami diet, typami posiłków, typami dań i wyposażeniem kuchennym • Zarządzanie treścią witryny, informacjami kontaktowymi i cennikiem

Na rysunku 2.1 przedstawiono relacje pomiędzy użytkownikami korzystając z notacji diagramu przypadków użycia języka UML.



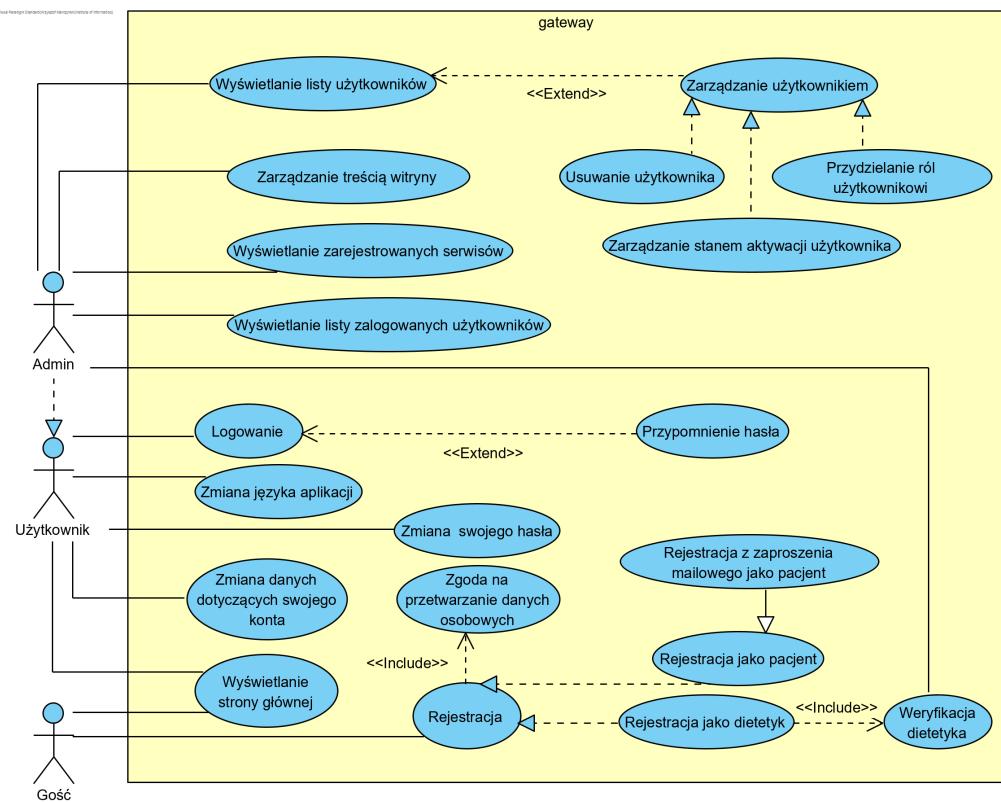
Rys. 2.1: Diagram przypadków użycia - użytkownicy (opr.wł)

2.7. Wymagania funkcjonalne

W tabelach 2.4 - 2.8 przedstawiono wymagania funkcjonalne dla poddziedzin systemu w postaci zestawienia potrzeb użytkowników systemu z cechami związanymi z realizacją danej potrzeby. Następnie wymagania dla konkretnych poddziedzin sformalizowano w postaci diagramów przypadków użycia języka UML odpowiednio na rysunkach 2.2 - 2.6.

Tabela 2.4: Wymagania funkcjonalne - poddziedzina administracyjna (opr.wł)

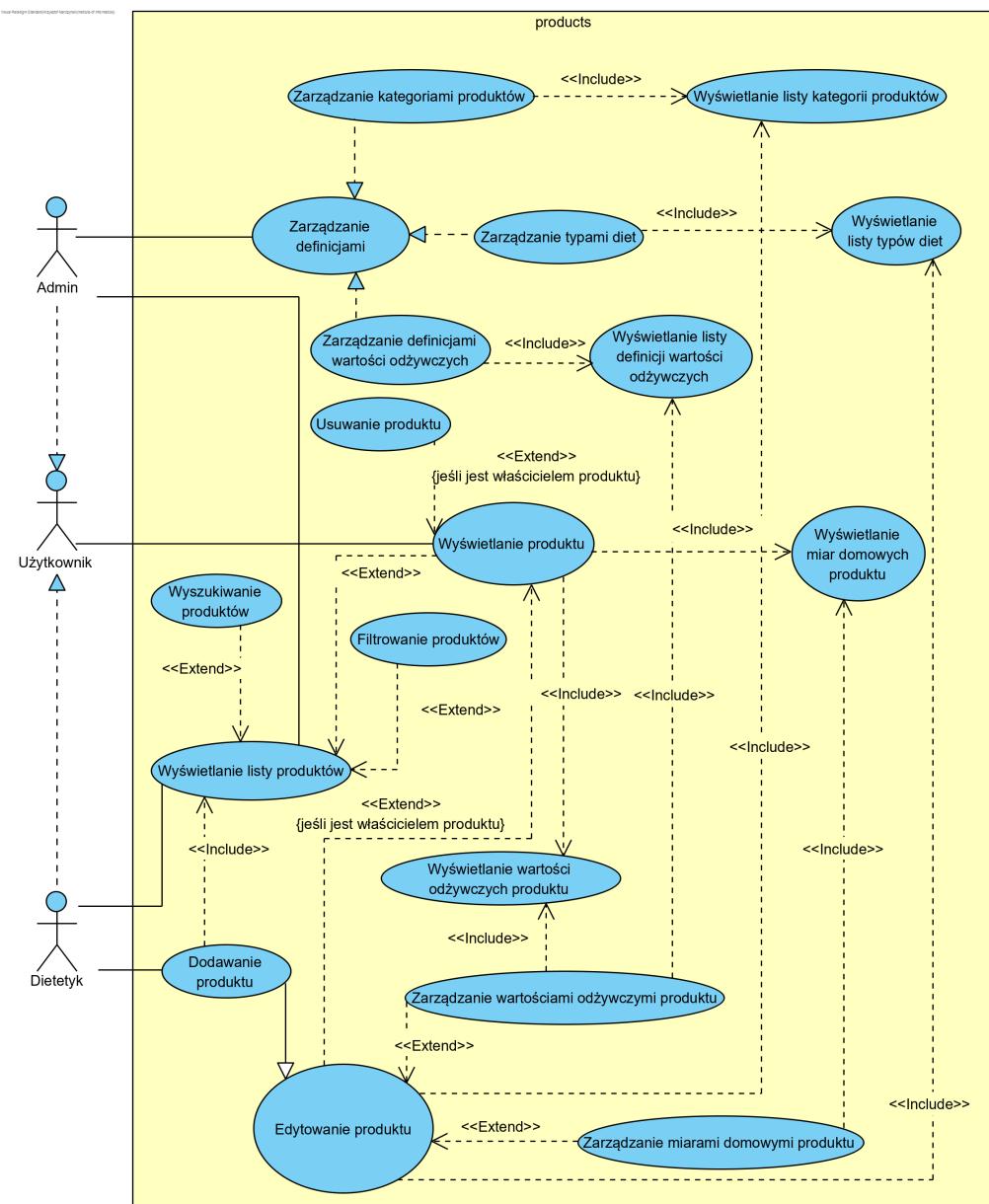
Potrzeby	Cechy
Administrator potrzebuje widzieć listę użytkowników	<ul style="list-style-type: none">Przydzielanie i odbieranie użytkownikom uprawnień
Użytkownik potrzebuje korzystać ze swojego konta	<ul style="list-style-type: none">Logowanie do systemuPrzypomnienie hasłaZarządzanie swoimi danymi osobowymi
Użytkownik chce przeglądać witrynę w swoim języku	<ul style="list-style-type: none">Obsługa witryny w wielu językach
Gość potrzebuje korzystać z systemu	<ul style="list-style-type: none">Zakładanie konta użytkownikaWyrażenie zgody na przetwarzanie swoich danych osobowych



Rys. 2.2: Diagram przypadków użycia - poddziedzina administracyjna (opr.wł)

Tabela 2.5: Wymagania funkcjonalne - poddziedzina produkty (opr.wł)

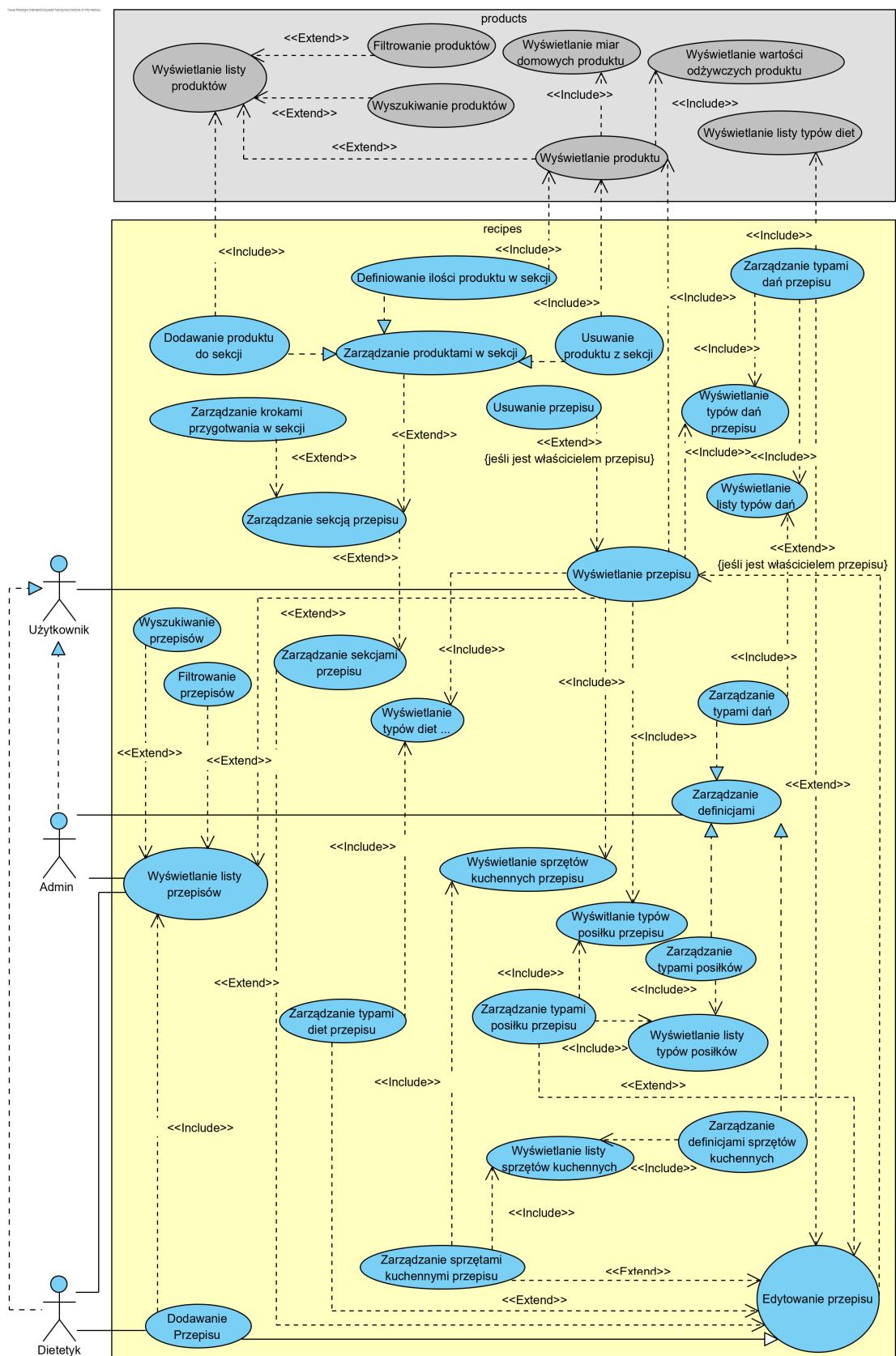
Potrzeby	Cechy
Administrator potrzebuje zarządzać definicjami potrzebnymi w produktach	<ul style="list-style-type: none"> ● Zarządzanie definicjami wartości odżywcznych ● Zarządzanie kategoriami produktów ● Zarządzanie rodzajami diet
Dietetyk potrzebuje widzieć listę produktów	<ul style="list-style-type: none"> ● Wyszukiwanie produktów ● Filtrowanie produktów ● Dodawanie nowych produktów
Dietetyk potrzebuje zarządzać szczegółami produktu	<ul style="list-style-type: none"> ● Edytowanie i usuwanie produktów ● Definiowanie wartości odżywcznych dla produktu ● Definiowanie miar domowych dla produktu ● Przypisywanie produktu do kategorii i podkategorii ● Definiowanie do jakich typów diet produkt nadaje się, a do jakich nie



Rys. 2.3: Diagram przypadków użycia - poddziedzina produkty (opr.wł)

Tabela 2.6: Wymagania funkcjonalne - poddziedzina przepisy (opr.wł)

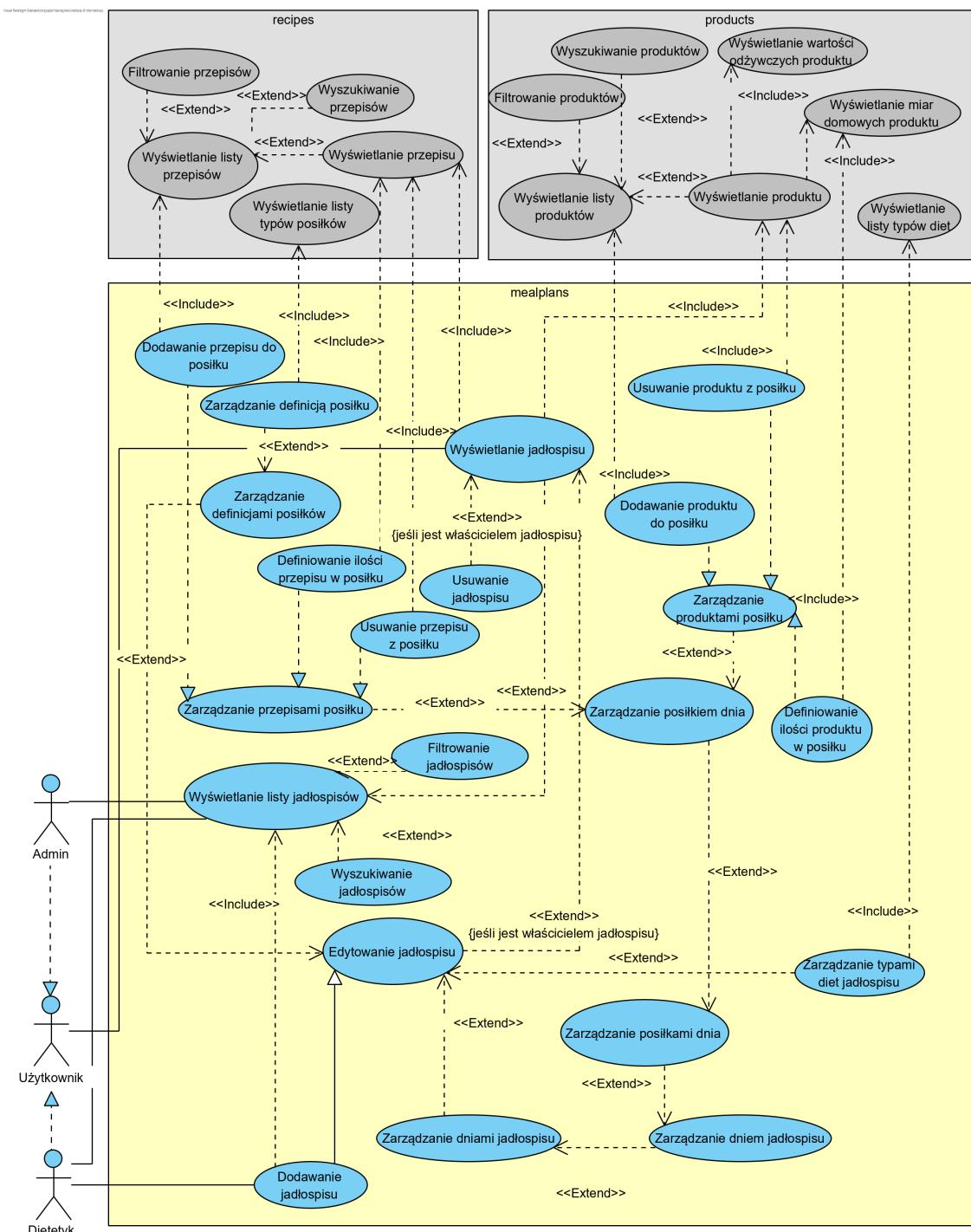
Potrzeby	Cechy
Administrator potrzebuje zarządzać definicjami niezbędnymi w przepisach	<ul style="list-style-type: none"> ● Zarządzanie typami posiłków ● Zarządzanie typami dań ● Zarządzanie definicjami wyposażenia kuchennego
Dietetyk potrzebuje widzieć listę przepisów	<ul style="list-style-type: none"> ● Wyszukiwanie przepisów ● Filtrowanie przepisów ● Dodawanie nowych przepisów
Dietetyk potrzebuje zarządzać szczegółami przepisu	<ul style="list-style-type: none"> ● Edytowanie i usuwanie przepisów ● Dodawanie wielu sekcji do przepisu ● Dodawanie do każdej sekcji listy składników ● Dodawanie do każdej sekcji sposobu przygotowania ● Dodawanie zdjęcia dania do przepisu ● Definiowanie czasu przygotowania posiłku



Rys. 2.4: Diagram przypadków użycia - podziedzina przepisy (opr.wł)

Tabela 2.7: Wymagania funkcjonalne - poddziedzina jadłospisy (opr.wł)

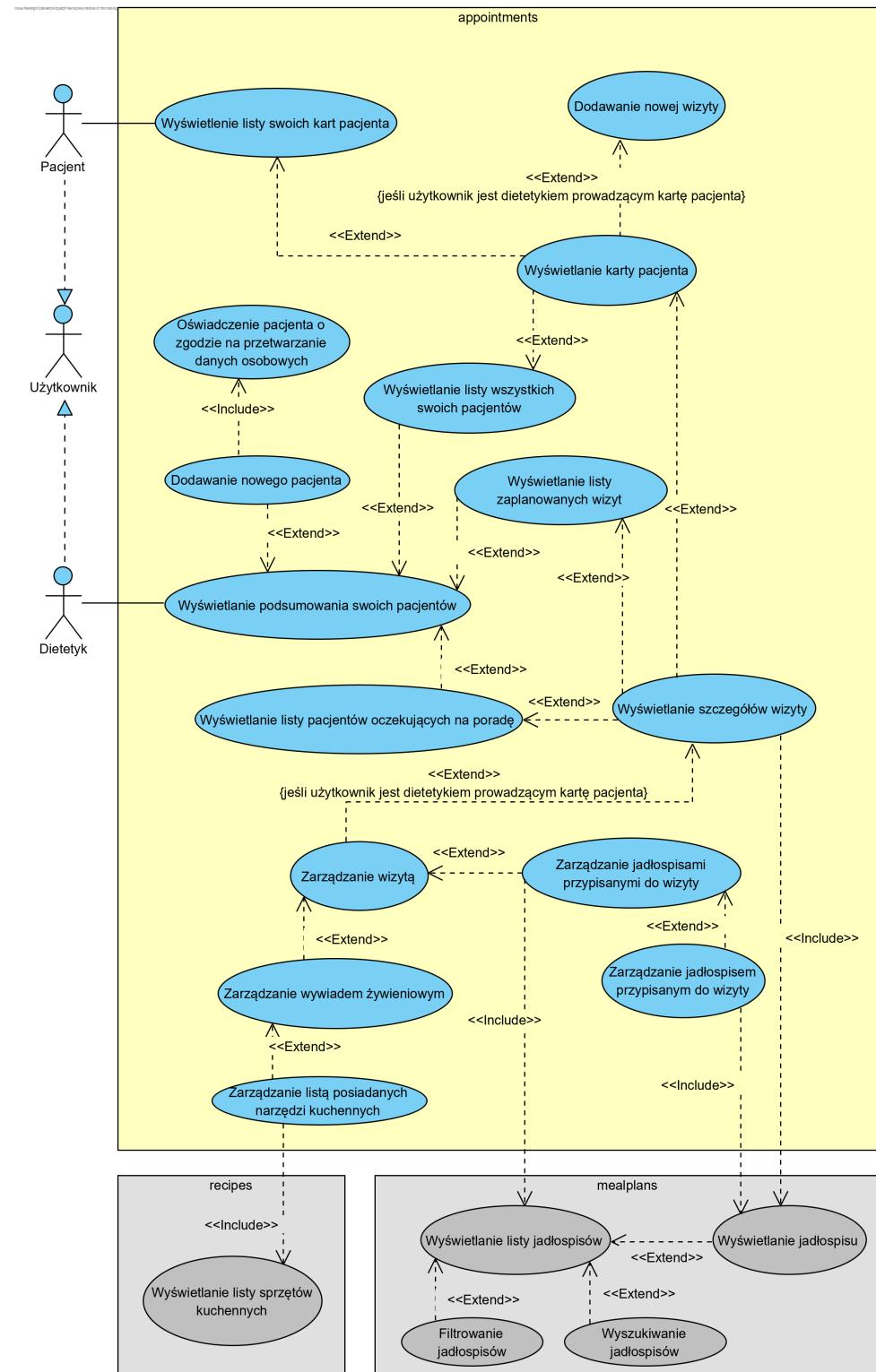
Potrzeby	Cechy
Dietetyk potrzebuje widzieć listę jadłospisów	<ul style="list-style-type: none"> ● Wyszukiwanie jadłospisów ● Filtrowanie jadłospisów ● Dodawanie nowych jadłospisów
Dietetyk potrzebuje zarządzać szczegółami jadłospisu	<ul style="list-style-type: none"> ● Dodawanie, edytowanie i usuwanie jadłospisów ● Definiowanie liczby dni na które będzie układany jadłospis ● Definiowanie liczby posiłków dziennie ● Definiowanie planowanego czasu każdego z posiłków ● Definiowanie procentowego udziału podstawowych wartości odżywczych w każdym posiłku ● Definiowanie posiłków w jadłospisie ● Dodawanie produktów i przepisów do posiłków



Rys. 2.5: Diagram przypadków użycia - poddziedzina jadłospisy (opr.wł)

Tabela 2.8: Wymagania funkcjonalne - poddziedzina wizyty (opr.wł)

Potrzeby	Cechy
Dietetyk potrzebuje wyświetlać listę swoich pacjentów	<ul style="list-style-type: none"> ● Wyszukiwanie pacjentów ● Wyświetlanie listy znalezionych pacjentów ● Wyświetlanie listy umówionych wizyt ● Wyświetlanie listy oczekujących porad ● Dodawanie nowych pacjentów
Dietetyk potrzebuje zarządzać kartą pacjenta	<ul style="list-style-type: none"> ● Wyświetlanie i edytowanie podstawowych informacji pacjenta ● Wyświetlanie listy wizyt pacjenta ● Wyświetlanie listy oczekujących porad pacjenta ● Dodawanie nowej wizyty pacjenta
Dietetyk potrzebuje wyświetlać szczegóły wizyty pacjenta	<ul style="list-style-type: none"> ● Wyświetlanie i edytowanie szczegółów wizyty pacjenta ● Zarządzanie pomiarami ciała pacjenta przypisanymi do wizyty ● Zarządzanie wywiadem żywieniowym przypisany do wizyty ● Zarządzanie jadłospisem przydzielonym do wizyty
Pacjent potrzebuje otrzymywać dietę	<ul style="list-style-type: none"> ● Udostępnianie pacjentowi jadłospisu
Pacjent chce mieć wgląd w swoją kartę	<ul style="list-style-type: none"> ● Logowanie do konta utworzonego w serwisie ● Dodawanie kart pacjenta do swojego konta po udostępnieniu ich przez dietetyka



Rys. 2.6: Diagram przypadków użycia - podzdiedzina wizyty (opr.wł)

2.8. Wymagania niefunkcjonalne

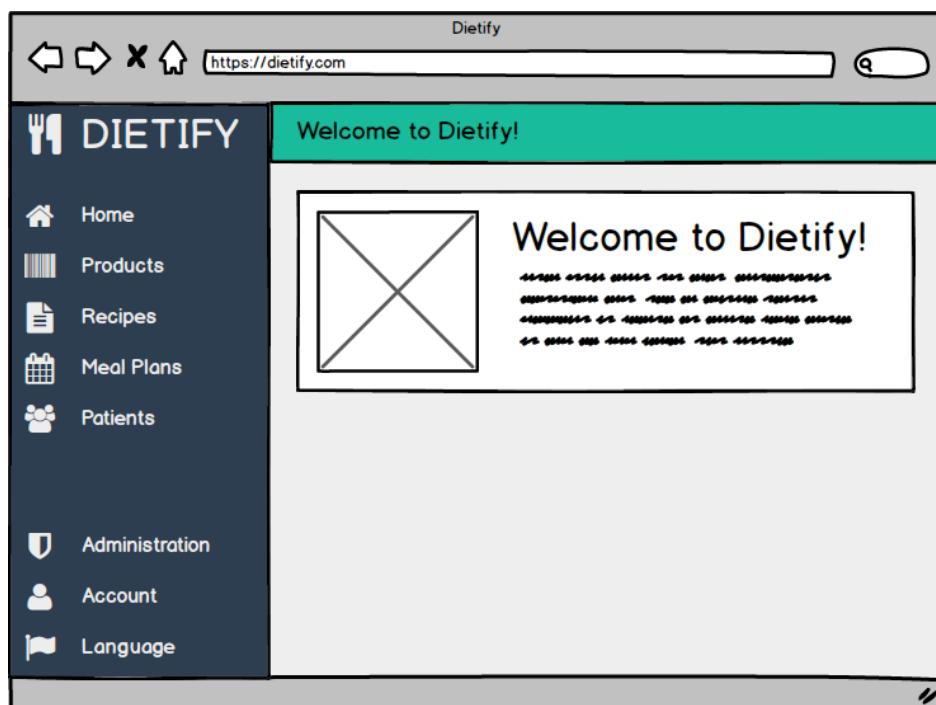
- System działa poprawnie w przeglądarkach Google Chrome 76, Mozilla Firefox 69, Safari 12, Opera 63, Microsoft Edge 17
- System działa na urządzenia mobilnych korzystających z systemu Android 9 i iOS 12
- System jest dostępny w polskiej i angielskiej wersji językowej
- System ma czytelny i minimalistyczny interfejs
- Aplikacja webowa jest w pełni responsywna i wygodna do używania na ekranach od 5 do 30 cali
- Aplikacja ma być oparta na architekturze mikroserwisów

3. Projekt

3.1. Prototyp interfejsu

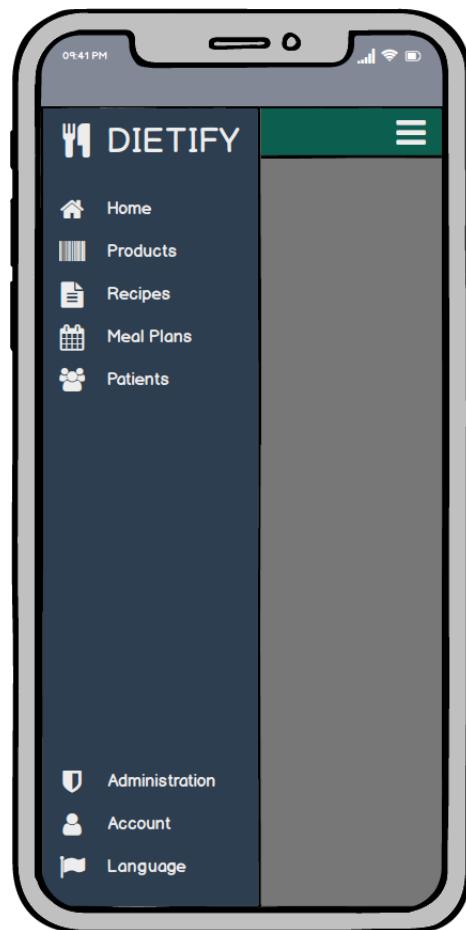
(opisać mockupy)

Opierając się zdobytej wiedzy dotyczącej potrzeb użytkowników aplikacji oraz na wymaganiach funkcjonalnych służących do zaspokojenia tych potrzeb opracowano prototypy kluczowych widoków interfejsu użytkownika. Zanim przedstawione zostaną prototypy dla poddziedzin warto wpierw omówić szablon widoku, który będzie stosowany w obrębie całej witryny. Na rysunku 3.1 przedstawiona została strona startowa witryny. U góry strony znajduje się belka zawierająca hasło przewodnie danego widoku. Po lewej stronie umieszczony został panel nawigacyjny, u góry którego widnieje logo i nazwa systemu. Na rzecz projektu roboczo przyjęto jako logo rysunek sztućców, a "Dietify" jako nazwę projektowanego systemu. Poniżej są odnośniki do strony startowej oraz do głównych widoków związanych z każdą z poddziedzin. Na dole panelu nawigacyjnego znajdują się łącza do zmiany języka, ustawień użytkownika oraz, jeśli zalogowany użytkownik ma uprawnienia administratora, do widoku administracyjnego. W głównej części widoku widoczna jest właściwa treść bieżącej strony. W przypadku omawianej strony startowej jest to wiadomość powitalna oraz obrazek związany z dietetyką.



Rys. 3.1: Prototyp interfejsu - strona startowa (opr.wł)

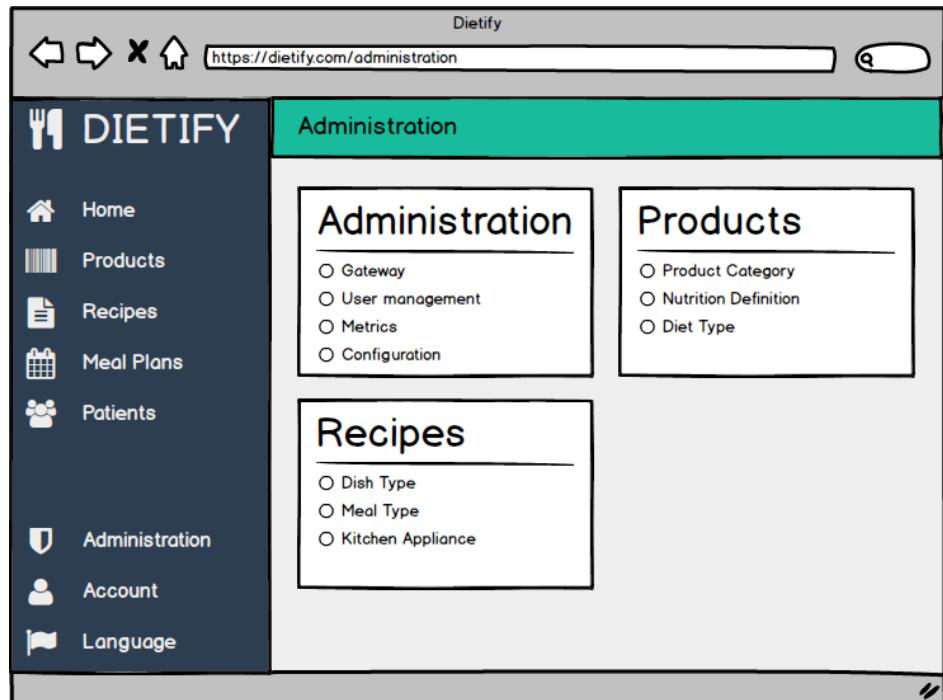
Podobnie szablon strony będzie wyglądał na urządzeniu mobilnym, co przedstawiono na rysunku 3.2.



Rys. 3.2: Prototyp interfejsu - układ strony na urządzeniu mobilnym (opr.wł)

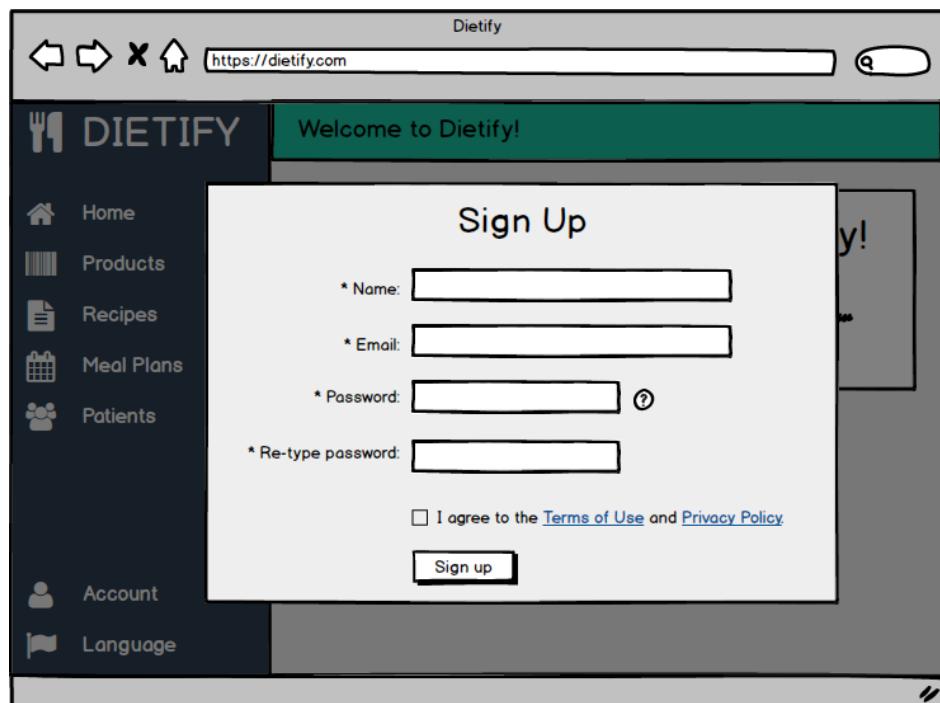
3.1.1. Poddziedzina administracyjna

Na rysunku 3.3 przedstawiono podstawowy widok administratora zawierający łącza do treści zarządzanych przez administratora w poszczególnych poddziedzinach.

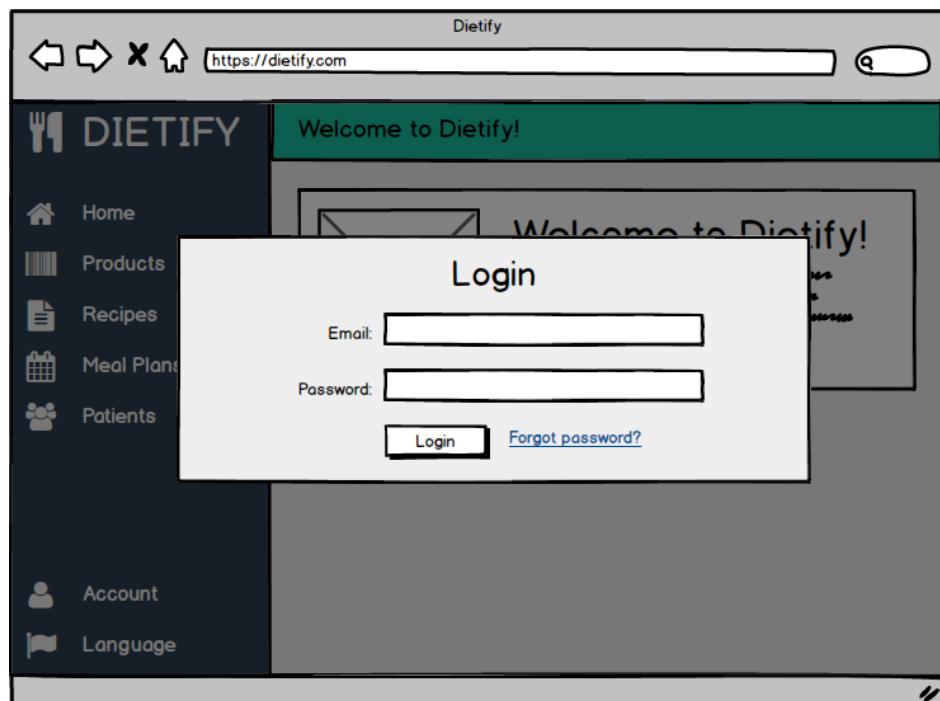


Rys. 3.3: Prototyp interfejsu - widok administratora (opr.wł)

Poddziedzina administracyjna jest związana przede wszystkim z zarządzaniem użytkownikami, przy czym należy również wziąć pod uwagę rejestrację (wysunek 3.4) i logowanie (rysunek 3.5) użytkowników do systemu. Warto zauważyć, że podczas rejestracji użytkownik wyraża zgodę na przetwarzanie swoich danych osobowych poprzez akceptację polityki prywatności.



Rys. 3.4: Prototyp interfejsu - rejestrowanie do systemu (opr.wł)



Rys. 3.5: Prototyp interfejsu - logowanie do systemu (opr.wł)

Widoku zarządzania użytkownikami (rysunek 3.6) jest dostępny tylko dla administratora i z jego poziomu możliwe jest zarządzanie uprawnieniami użytkowników, zmiana statusu ich kont oraz tworzenie nowych użytkowników.

The screenshot shows a web browser window for 'Dietify' at the URL <https://dietify.com/administration/user-management>. The page title is 'Users'. On the left, there is a dark sidebar with the 'DIETIFY' logo and links: Home, Products, Recipes, Meal Plans, Patients, Administration (selected), Account, and Language. The main content area has a green header 'Users' and a 'Create new user' button. A table lists three users:

ID	Login	Email	Status	Profiles
1	admin	admin@localhost	activated	ROLE_ADMIN, ROLE_USER
2	dietitian	dietitian@localhost	activated	ROLE_DIETITIAN, ROLE_USER
3	patient	patient@localhost	activated	ROLE_PATIENT, ROLE_USER

Rys. 3.6: Prototyp interfejsu - zarządzanie użytkownikami (opr.wł)

3.1.2. Poddziedzina produkty

The screenshot shows the Dietify application's product list page. The left sidebar contains navigation links: Home, Products (selected), Recipes, Meal Plans, Patients, Account, and Language. The main content area has a header 'Products' with a 'Create new Product' button. A search bar is followed by a table showing product details. The table columns are: Name, Subcategory, Lang, Energy, Protein, Fat, and Carbs. The data includes:

Name	Subcategory	Lang	Energy	Protein	Fat	Carbs
Acerola	Fruits	EN	23	0.4	0.3	4.8
Asparagus	Vegetables	EN	20	2.2	0.12	3.88
Cheese	Diary	EN	353	21.4	28.74	2.34
Egg	Diary	EN	357	84.08	0.32	4.51

Pagination at the bottom shows pages 1 through 20.

Rys. 3.7: Prototyp interfejsu - lista produktów (opr.wł)

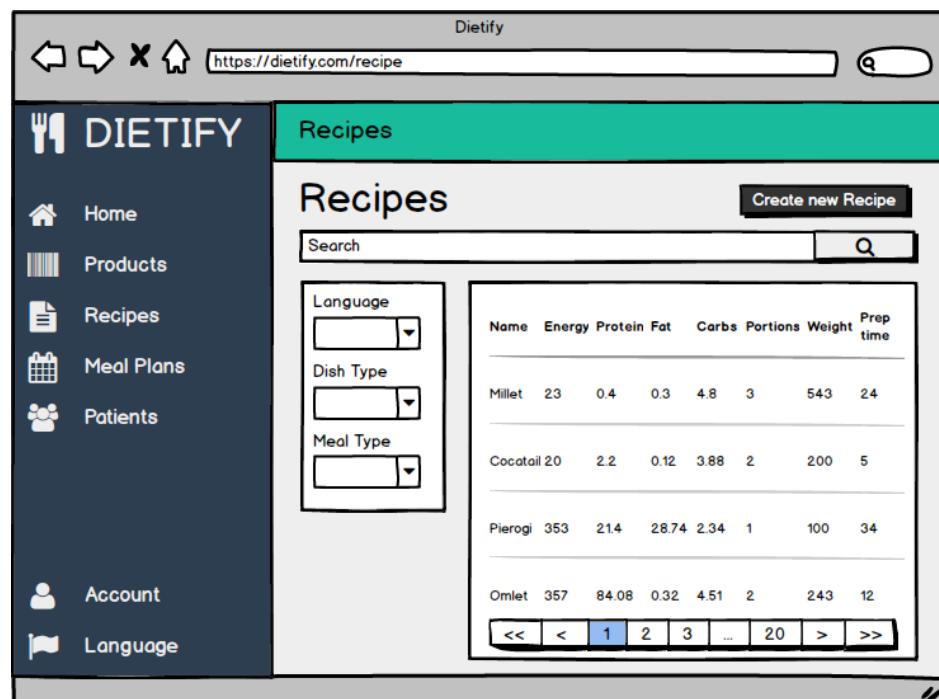
The screenshot shows the 'Create or Edit Product' form. The left sidebar is identical to the previous screen. The main form has tabs: Basic (selected), Nutrition Data, and Household Measures. The 'Basic' tab contains fields for Description, Source, Category (set to Diary), Subcategory (set to Milk), Suitable Diets, and Unsuitable Diets. It also includes a 'Basic Nutrients' section with fields for Energy, Protein, Fat, and Carbohydrates. A 'Language' dropdown is set to English. Buttons for 'Cancel' and 'Save' are at the top right.

Rys. 3.8: Prototyp interfejsu - dodawanie nowego lub edycja istniejącego produktu (opr.wł)



Rys. 3.9: Prototyp interfejsu - szczegóły produktu (opr.wł)

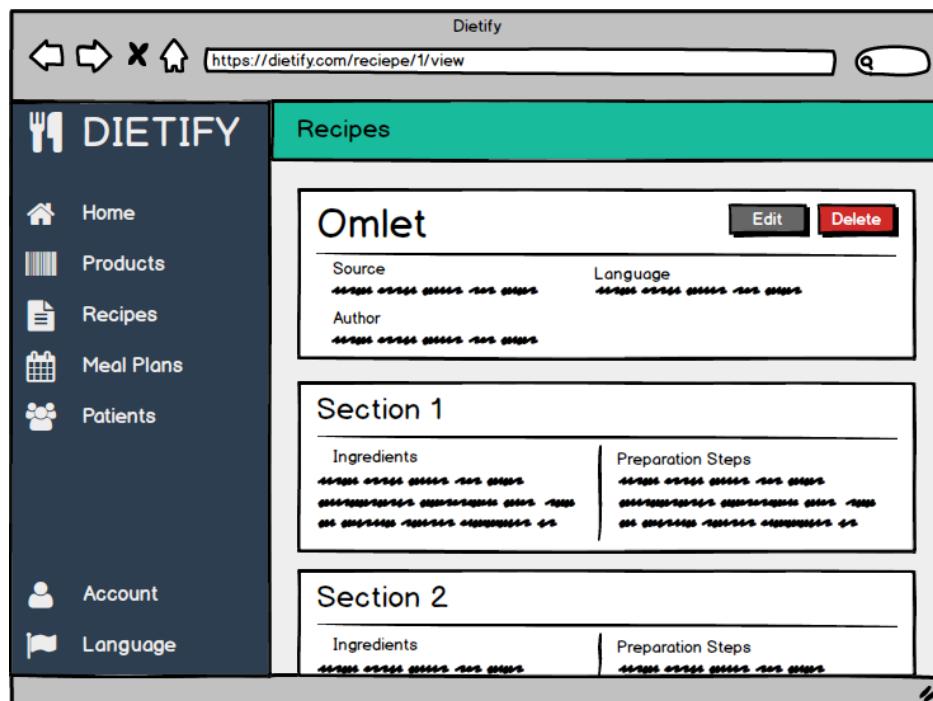
3.1.3. Poddziedzina przepisy



Rys. 3.10: Prototyp interfejsu - lista przepisów (opr.wł)



Rys. 3.11: Prototyp interfejsu - dodawanie nowego lub edycja istniejącego przepisu (opr.wł)



Rys. 3.12: Prototyp interfejsu - szczegóły przepisu (opr.wł)

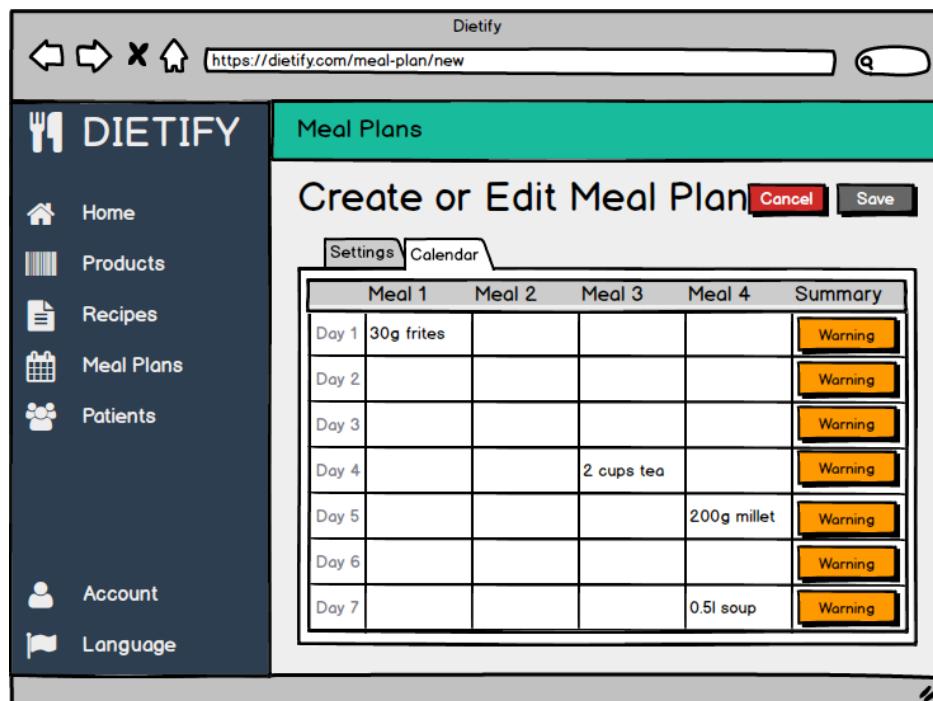
3.1.4. Poddziedzina jadłospisy

The screenshot shows the 'Meal Plans' section of the Dietify application. On the left, a sidebar menu includes 'Home', 'Products', 'Recipes', 'Meal Plans' (selected), and 'Patients'. Below these are 'Account' and 'Language' options. The main area has a green header 'Meal Plans' with a 'Create new Meal Plan' button. A search bar is at the top right. To the left of the table is a form with dropdown menus for 'Language', 'Number of days', 'Meals per day', and 'Kcal per day'. To the right is a table showing meal plans with columns for Name, Number of days, Meals per day, and Kcal per day. The table lists four plans: 'Awesome plan' (5 days, 5 meals, 2100 kcal), 'Strict diet' (7 days, 5 meals, 1400 kcal), 'For allergic' (30 days, 6 meals, 2400 kcal), and 'For Ms. Sunshine' (10 days, 4 meals, 1700 kcal). Navigation arrows at the bottom allow for page navigation.

Rys. 3.13: Prototyp interfejsu - lista jadłospisów (opr.wł)

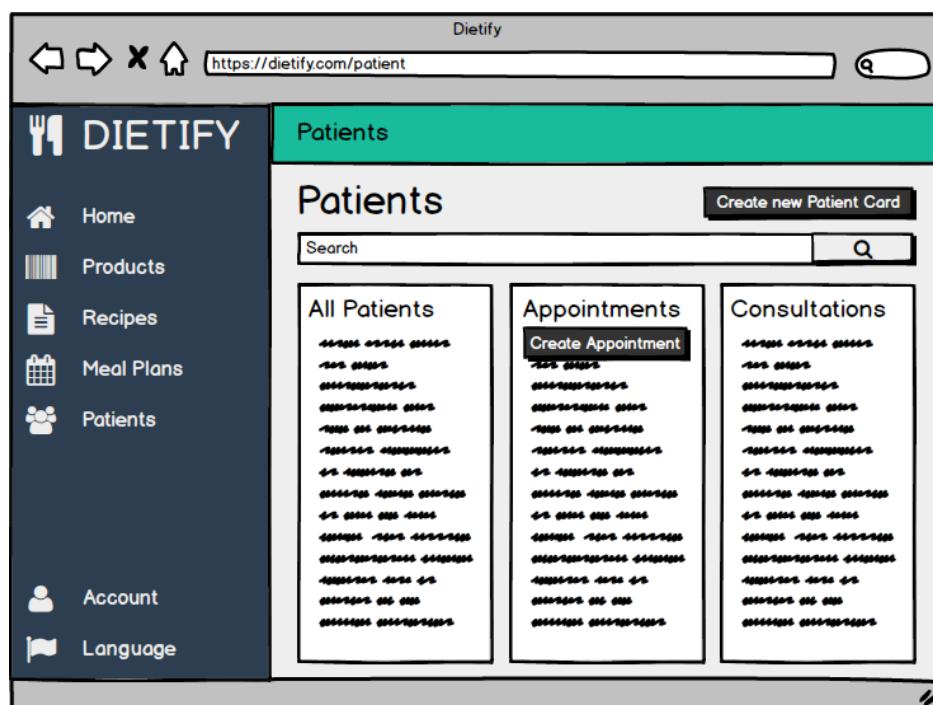
The screenshot shows the 'Create or Edit Meal Plan' dialog. The sidebar is identical to the previous screen. The main area has a green header 'Meal Plans' with 'Create or Edit Meal Plan' subtext. It features two tabs: 'Settings' (selected) and 'Calendar'. The 'Settings' tab contains fields for 'Name' (empty), 'Number of Days' (7), 'Language' (English), 'Number of Meals per Day' (4), 'Meal Definitions' (a table with four rows for meal times and percentages: Breakfast at 07:00 is 25%, followed by three other rows with incomplete data), and 'Energy distribution' (fields for Kcal daily, % Protein, % Fat, and % Carbs). 'Cancel' and 'Save' buttons are at the top right of the dialog.

Rys. 3.14: Prototyp interfejsu - dodawanie nowego lub edycja istniejącego jadłospisu - zakładka ustawień (opr.wł)



Rys. 3.15: Prototyp interfejsu - dodawanie nowego lub edycja istniejącego jadłospisu - zakładka kalendarza (opr.wł)

3.1.5. Poddziedzina wizyty



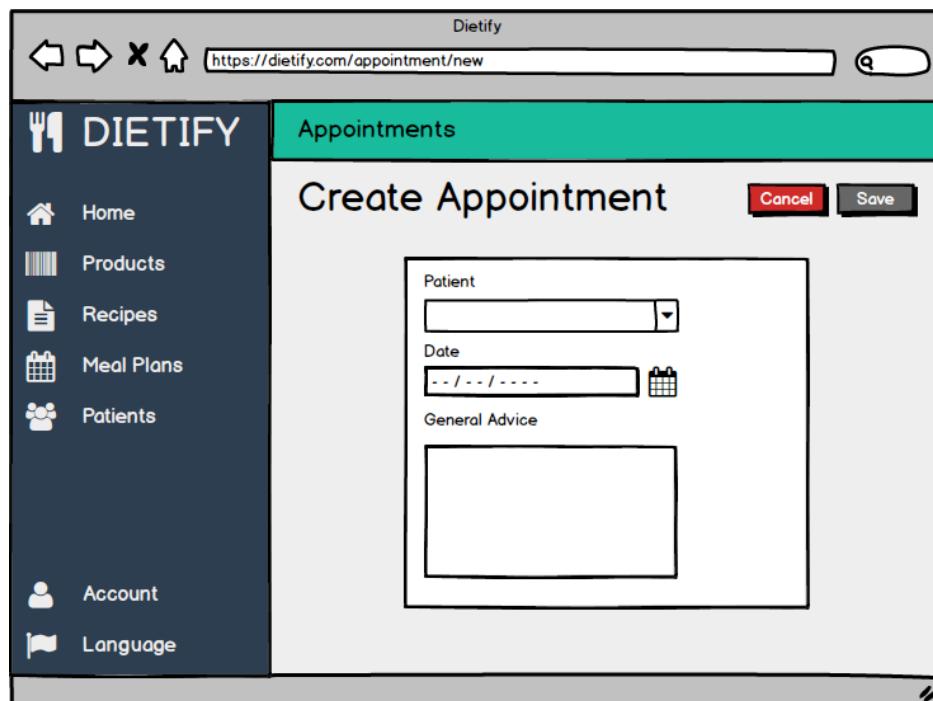
Rys. 3.16: Prototyp interfejsu - lista wizyt (opr.wł)

The screenshot shows a web browser window for the Dietify application. The URL is <https://dietify.com/patient/new>. The page title is "Dietify". On the left, there is a dark sidebar with the Dietify logo and links to Home, Products, Recipes, Meal Plans, Patients, Account, and Language. The main content area has a green header bar with the text "Patients" and "Create Patient Card". Below the header, there are input fields for "Email", "First Name", and "Last Name". A checkbox labeled "I hereby declare that Patient has signed a consent for processing their personal data as described in [Privacy Policy](#)" is present. At the top right are "Cancel" and "Save" buttons.

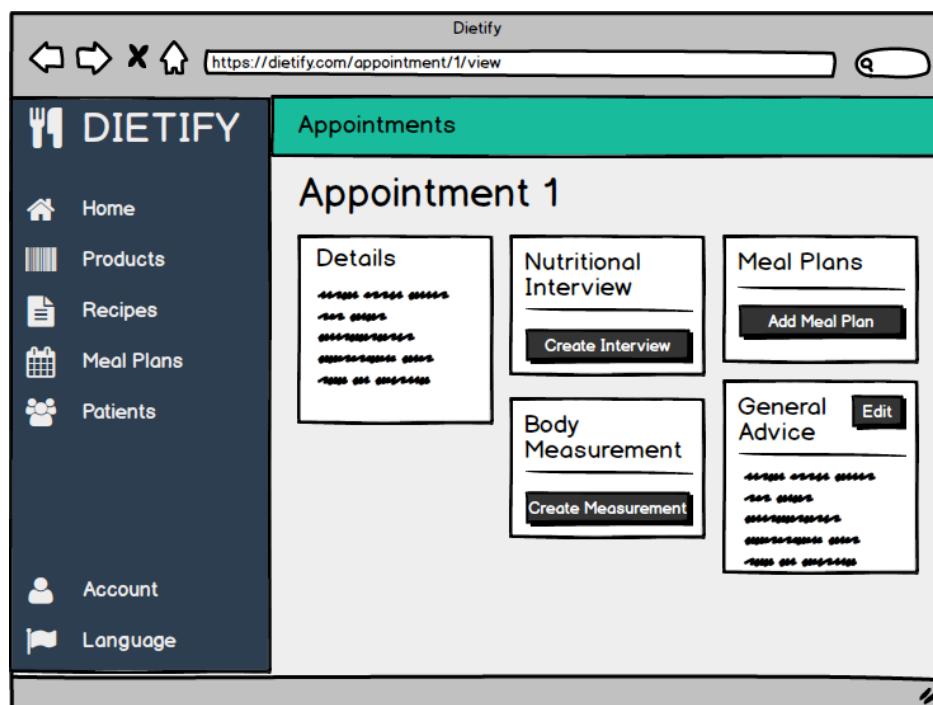
Rys. 3.17: Prototyp interfejsu - dodawanie nowej karty pacjenta (opr.wł)

The screenshot shows a web browser window for the Dietify application. The URL is <https://dietify.com/patient/1/view>. The page title is "Dietify". The sidebar on the left is identical to the one in the previous screenshot. The main content area shows a patient card for "Jan Kowalski". The card includes sections for "Patient Details" (with placeholder text), "Appointments" (with a "Create Appointment" button and a list of entries), and "Consultations" (with a list of entries). An "Edit" button is located at the top right of the card. There is also a small chart titled "BMI" showing a line graph.

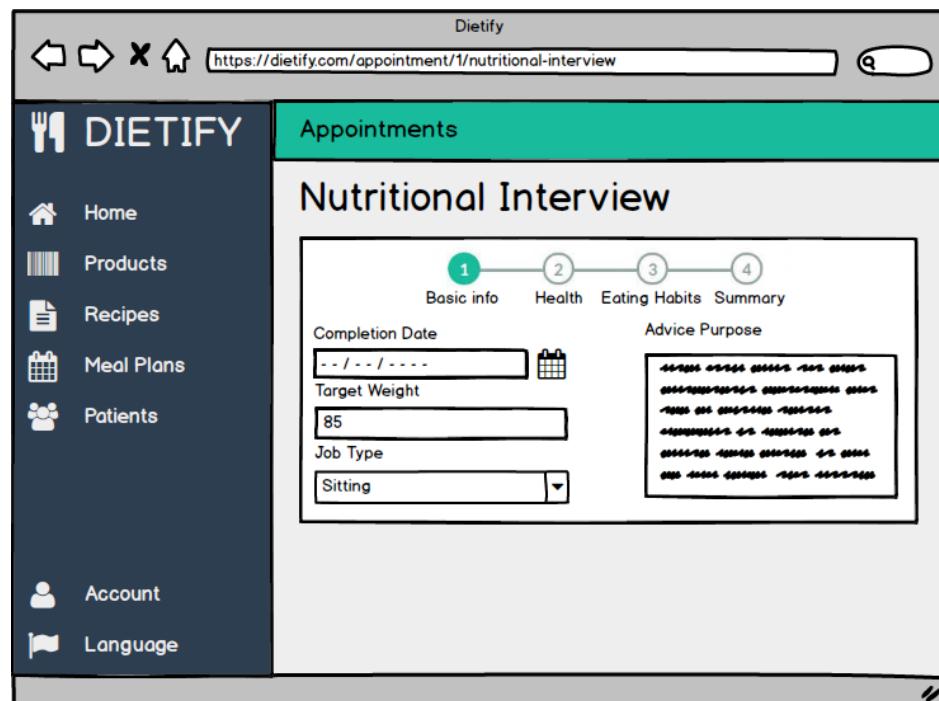
Rys. 3.18: Prototyp interfejsu - szczegóły karty pacjenta (opr.wł)



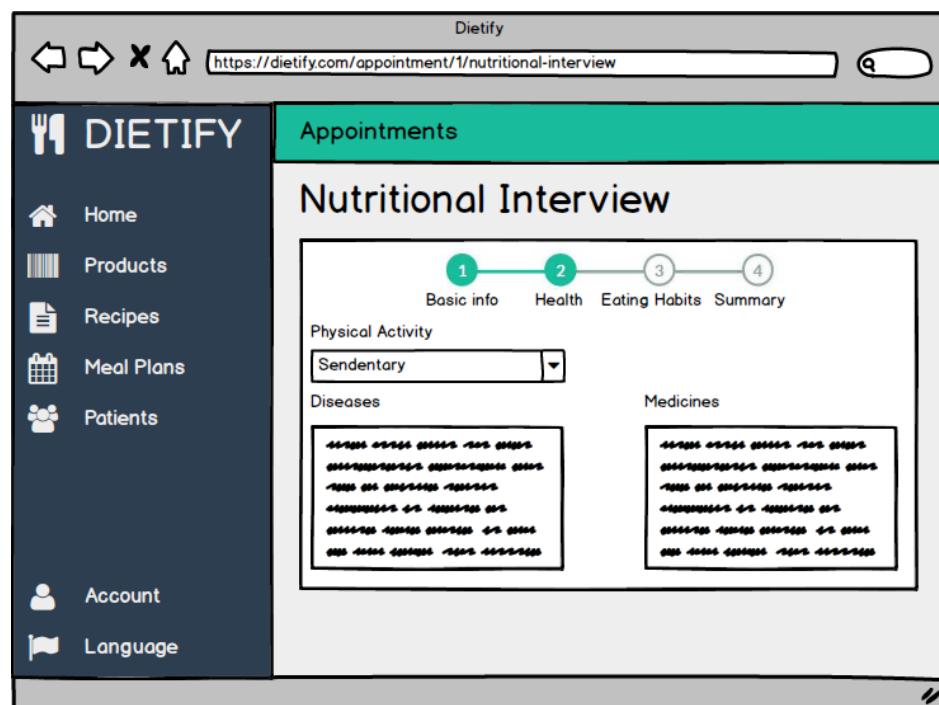
Rys. 3.19: Prototyp interfejsu - dodawanie nowej wizyty (opr.wł)



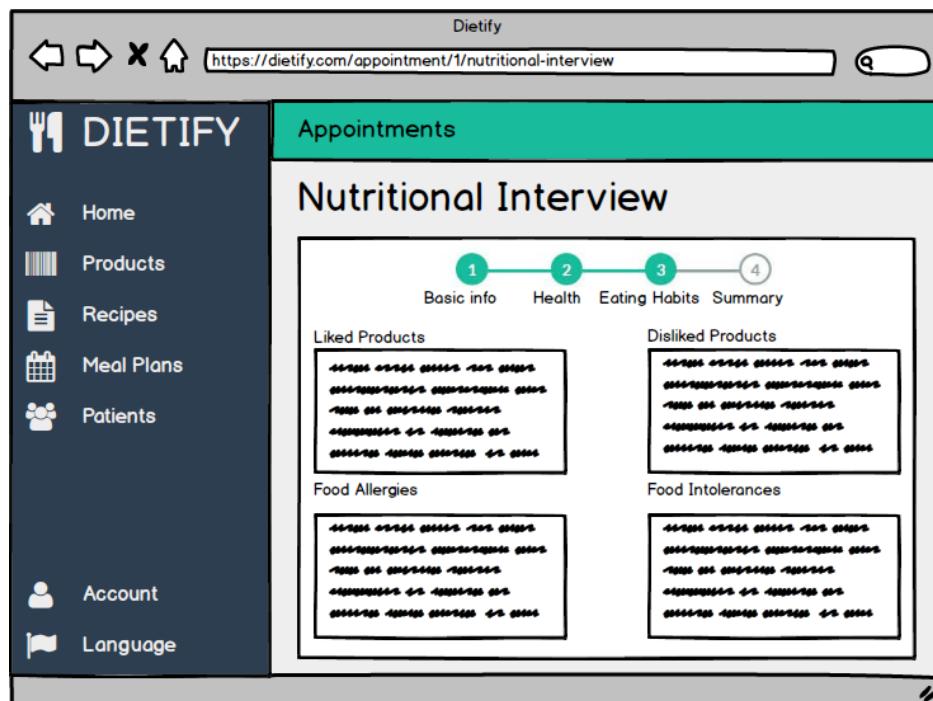
Rys. 3.20: Prototyp interfejsu - szczegółowy wizyty (opr.wł)



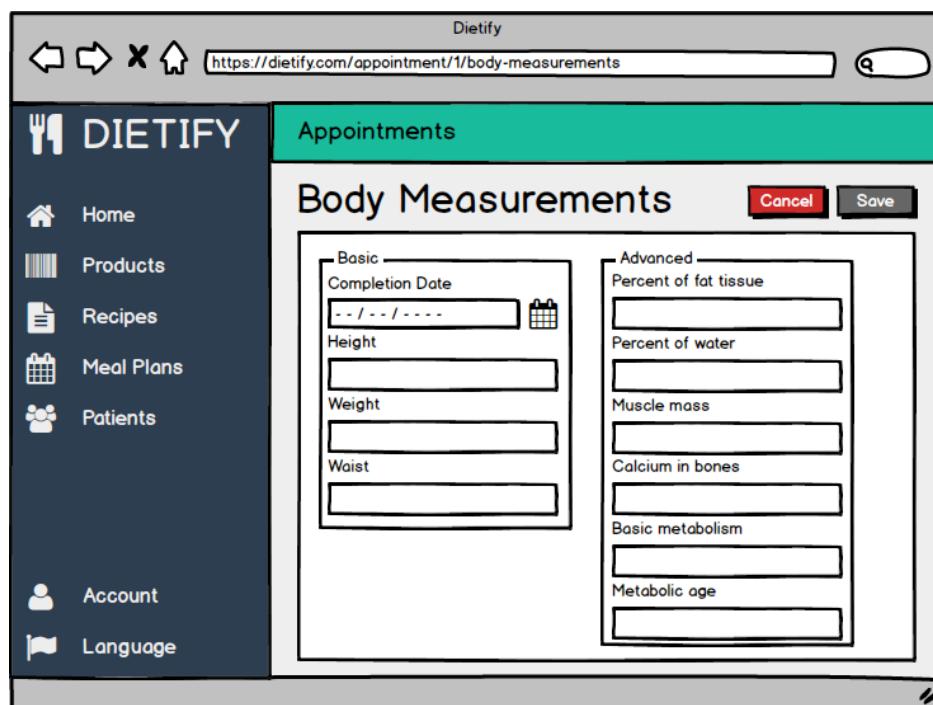
Rys. 3.21: Prototyp interfejsu - wizyta - wywiad żywieniowy - krok 1 (opr.wł)



Rys. 3.22: Prototyp interfejsu - wizyta - wywiad żywieniowy - krok 2 (opr.wł)



Rys. 3.23: Prototyp interfejsu - wizyta - wywiad żywieniowy - krok 3 (opr.wł)

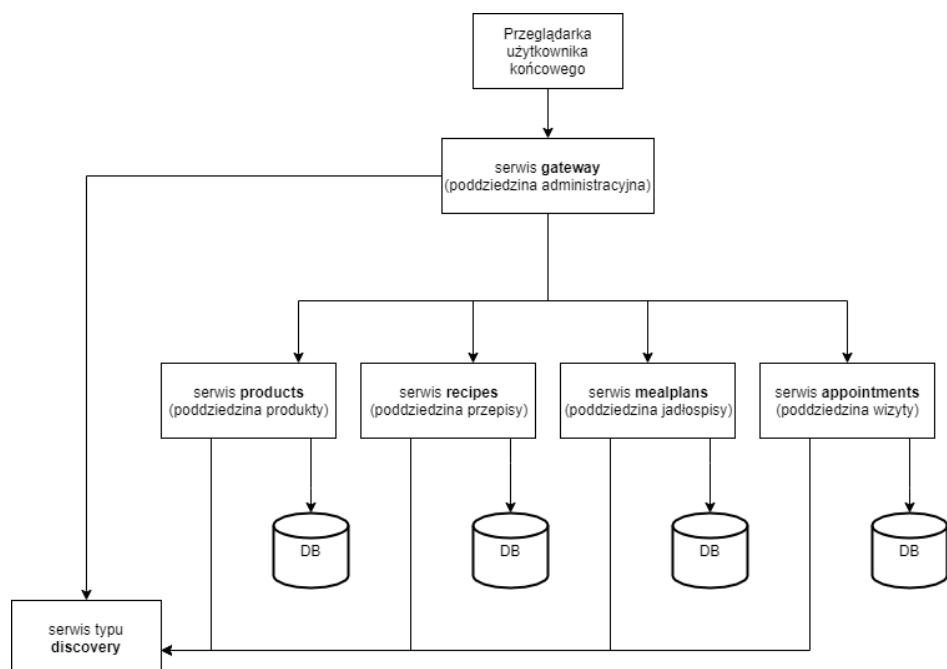


Rys. 3.24: Prototyp interfejsu - wizyta - pomiary ciała (opr.wł)

3.2. Opis podstawowej architektury systemu

Jak wspomniano w rozdziale 1.3, dekompozycja w oparciu o poddzedziny pozwala na proste wyznaczenie mikroserwisów poprzez stworzenie serwisu dla każdej poddzedziny. Każdy mikroserwisy potrzebuje przechowywać dane w bazie danych, która będzie odseparowana od baz innych mikroserwisów przynajmniej na poziomie logicznym. Instancje mikroserwisów mogą być tworzone dynamicznie i w efekcie mogą być dostępne pod różnymi adresami, należy więc uwzględnić mechanizm pozwalający na automatyczną detekcję bieżących lokalizacji mikroserwisów. Zadanie takie może być realizowane przez serwis typu discovery. Dodatkowo potrzebne jest rozwiązań umożliwiające odbieranie żądań użytkownika i przekierowywanie ich do odpowiednich serwisów z uwzględnieniem obciążenia (ang. load balancing). Do tego celu służyć może serwis działający jako brama aplikacji (ang. gateway), przy czym warto zauważyć, że brama aplikacji powinna również być odpowiedzialna za autoryzację żądań użytkowników, dlatego też poddzedzina administracyjna będzie zaimplementowana w ramach serwisu gateway.

Na rysunku 3.25 przedstawiono diagram podstawowej architektury uwzględniający powyższe rozważania.



Rys. 3.25: Podstawowa architektura systemu (opr.wł)

3.3. Projekt bazy danych

Podstawą projektowanego systemu jest przetwarzanie danych związanych z domeną zarządzania dietą, dlatego niezmiernie istotne jest zapewnienie trwałości i spójności danych. Cechy takie może zapewnić relacyjna baza danych dzięki mechanizmowi transakcji[17]. Jednakże w przypadku architektury mikroserwisów, w której każdy serwis korzysta z własnego schematu bazy danych spójność będzie gwarantowana jedynie w obrębie jednego serwisu, gdyż więcej integralności nie będą definiowane pomiędzy danymi znajdującymi się w bazach różnych serwisów. Aby rozwiązać ten problem, dla danych definiowanych przez użytkownika

końcowego (dietetyka), do których odniesienia będą przechowywane w serwisie, który nie jest ich właścicielem, zastosowany zostanie wzorzec "Tymczasowego Obiektu" (ang. Temporal Object)[7] nazywany również wzorcem wersjonowania. Dla raz utworzonej wersji obiektu dane pozostają niezmienne, a w razie potrzeby zmiany danych należy utworzyć nową wersję. Dzięki temu referencję do wersji można bezpiecznie wykorzystywać w środowisku rozproszonym. Należy zauważyć, że konsekwencją zastosowania tego wzorca jest niemożność usuwania obiektów. Zamiast tego należy stworzyć finalną wersję obiektu z flagą pokazującą, że obiekt powinien być traktowany jako usunięty.

W dalszej części niniejszego podrozdziału zostanie przeprowadzone projektowanie konceptualne, w ramach którego dla każdej rozważanej poddziedziny zdefiniowane zostaną kategorie (obiekty) oraz związane z nimi reguły funkcjonowania i ograniczenia dziedzinowe[17], które następnie zostaną podsumowane w formie modelu informacyjnego wyrażonego za pomocą diagramu klas języka UML.

3.3.1. Ogólne cechy dziedziny

Reguły funkcjonowania

Reguły ogólne

- REG/0/01** Przedmiot kompozycji podlega takim samym zasadom dostępu co właściciel kompozycji pod warunkiem, że przedmiot kompozycji nie definiuje własnych reguł dostępu

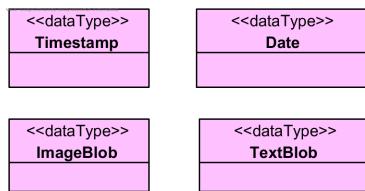
Ograniczenia dziedzinowe

Ograniczenia ogólne

- OGR/0/01** Wszystkie **id** muszą być unikalne
OGR/0/02 Wszystkie **id** są wymagane
OGR/0/03 Wszystkie **id** są liczbami całkowitymi dodatnimi tworzonymi przez SZBD za pomocą autonumerowania
OGR/0/04 Wszystkie atrybuty **language** są wymagane
OGR/0/05 Wszystkie **language** są ciągami znaków o długości 2 znaków spełniającymi normę ISO 639-1
OGR/0/06 Wszystkie **stemple czasowe** są w formacie YYYY:MM:DD HH:MI:SS
OGR/0/07 Wszystkie **daty** są w formacie YYYY:MM:DD
OGR/0/08 Ciągi znaków bez dodatkowych ograniczeń mogą zawierać dowolne znaki dopuszczalne w systemie kodowania UTF-8

Model informacyjny

Na rysunku 3.26 przedstawiono definicje niestandardowych typów danych potrzebnych przy prezentacji modelu informacyjnego.



Rys. 3.26: Diagram klas - typy danych (opr.wł)

3.3.2. Poddziedzina administracyjna

Kategorie

KAT/1/01 User (Użytkownik)

Opis: Konto użytkownika aplikacji. Każdy zalogowany użytkownik musi mieć konto użytkownika

Atrybuty:

- | | |
|--------------------|---|
| • id | - identyfikator |
| • login | - login użytkownika |
| • passwordHash | - reprezentacja hasła stworzona przez nałożenie na hasło funkcji skrótu |
| • firstName | - imię użytkownika |
| • lastName | - nazwisko użytkownika |
| • email | - adres e-mail |
| • image | - zdjęcie profilowe użytkownika |
| • activated | - flaga pokazująca czy konto użytkownika zostało aktywowane |
| • language | - język użytkownika w postaci kodu ISO 639-1 |
| • activationKey | - klucz wymagany podczas aktywacji konta użytkownika |
| • resetKey | - klucz wymagany podczas resetowania hasła do konta użytkownika |
| • createdDate | - data utworzenia konta |
| • resetDate | - data ostatniego resetowania hasła do konta |
| • lastModifiedDate | - data ostatniej modyfikacji konta |

KAT/1/02 Authority (Rola)

Opis: Rola użytkownika od której zależy zakres uprawnień użytkownika

Atrybuty:

- | | |
|--------|--------------|
| • name | - nazwa roli |
|--------|--------------|

KAT/1/03 UserExtraInfo (Dodatkowe Informacje Użytkownika)

Opis: Dodatkowe informacje o użytkowniku

Atrybuty:

- | | |
|----------|-----------------|
| • id | - identyfikator |
| • gender | - płeć |

• dateOfBirth	- data urodzenia
• phoneNumber	- numer telefonu, najlepiej w formacie (+00) 000-000-000
• streetAddress	- adres zamieszkania
• postalCode	- kod pocztowy
• city	- miasto
• country	- państwo
• personalDescription	- krótki opis osobisty. W przypadku dietetyka może zawierać dodatkowe informacje o prowadzonej praktyce dietetycznej

Reguły funkcjonowania

Reguły dla KAT/1/01 User

- REG/1/01** Użytkownik (**KAT/1/01 User**) nie musi mieć żadnych dodatkowych informacji (**KAT/1/03 UserExtraInfo**)
- REG/1/02** Użytkownik (**KAT/1/01 User**) może mieć maksymalnie jedne dodatkowe informacje (**KAT/1/03 UserExtraInfo**)
- REG/1/03** Użytkownik (**KAT/1/01 User**) musi mieć przynajmniej jedną rolę (**KAT/1/02 Authority**)
- REG/1/04** Użytkownik (**KAT/1/01 User**) może mieć wiele ról (**KAT/1/02 Authority**)
- REG/1/05** Użytkownik (**KAT/1/01 User**) nie musi mieć autora (**KAT/1/01 User**)
- REG/1/06** Użytkownik (**KAT/1/01 User**) może mieć maksymalnie jednego autora (**KAT/1/01 User**)
- REG/1/07** Użytkownik (**KAT/1/01 User**) nie musi mieć ostatniego edytora (**KAT/1/01 User**)
- REG/1/08** Użytkownik (**KAT/1/01 User**) może mieć maksymalnie jednego ostatniego edytora (**KAT/1/01 User**)
- REG/1/09** Gość może dodawać nowego użytkownika (**KAT/1/01 User**)
- REG/1/10** Użytkownik może wyświetlać, edytować i usuwać swoje dane użytkownika (**KAT/1/01 User**)
- REG/1/11** Dietetyk może wyświetlać dane *Pacjenta*, którego kartotekę prowadzi
- REG/1/12** Administrator może wyświetlać i usuwać dane użytkownika (**KAT/1/01 User**)

Reguły dla KAT/1/02 Authority

- REG/1/13** Administrator może dodawać, wyświetlać, edytować i usuwać dane roli (**KAT/1/02 Authority**)

Reguły dla KAT/1/03 UserExtraInfo

- REG/1/14** Dodatkowe informacje (**KAT/1/03 UserExtraInfo**) muszą być przypisane do dokładnie jednego użytkownika (**KAT/1/01 User**)
- REG/1/15** Dodatkowe informacje (**KAT/1/03 UserExtraInfo**) są przedmiotem kompozycji ze strony użytkownika (**KAT/1/01 User**)
- REG/1/16** Administrator nie może wyświetlać i usuwać dodatkowych informacji (**KAT/1/03 UserExtraInfo**) innych użytkowników

Ograniczenia dziedzinowe

Ograniczenia dla KAT/1/01 User

- OGR/1/01** Atrybut **login** jest wymagany
- OGR/1/02** Atrybut **passwordHash** jest wymagany
- OGR/1/03** Atrybut flagę **activated** jest wymagany
- OGR/1/04** Atrybut **createdDate** jest wymagany
- OGR/1/05** Atrybut **login** ma unikalną wartość
- OGR/1/06** Atrybut **email** ma unikalną wartość
- OGR/1/07** Atrybut **login** jest ciągiem znaków składającym się z liter, cyfr i dodatkowo mogącym zawierać znaki ".", "-", "@" o długości od 1 do 50 znaków
- OGR/1/08** Atrybut **passwordHash** jest ciągiem znaków o długości 60 znaków
- OGR/1/09** Atrybut **firstName** jest ciągiem znaków o długości do 50 znaków
- OGR/1/10** Atrybut **lastName** jest ciągiem znaków o długości do 50 znaków
- OGR/1/11** Atrybut **email** jest ciągiem znaków o długości od 5 do 254 znaków
- OGR/1/12** Atrybut **activated** jest typem logicznym
- OGR/1/13** Atrybut **image** jest ciągiem znaków o długości do 256 znaków tworzącym poprawny adres URL
- OGR/1/14** Atrybut **activationKey** jest ciągiem znaków o długości 20 znaków
- OGR/1/15** Atrybut **resetKey** jest ciągiem znaków o długości 20 znaków
- OGR/1/16** Atrybut **resetDate** jest stemplem czasowym
- OGR/1/17** Atrybut **createdDate** jest stemplem czasowym
- OGR/1/18** Atrybut **lastModifiedDate** jest stemplem czasowym

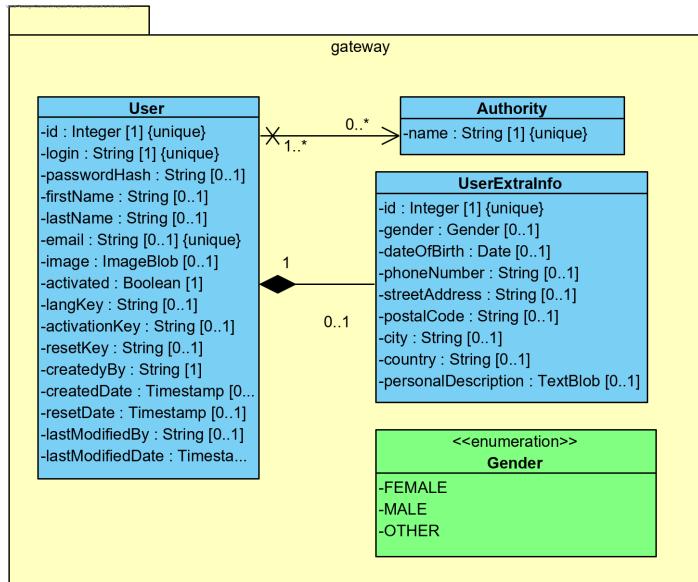
Ograniczenia dla KAT/1/02 Authority

- OGR/1/19** Atrybut **name** jest wymagany
- OGR/1/20** Atrybut **name** ma unikalną wartość
- OGR/1/21** Atrybut **name** jest ciągiem znaków składającym się z liter i znaków "_" o długości od 1 do 255 znaków

Ograniczenia dla KAT/1/03 UserExtraInfo

- OGR/1/22** Atrybut **gender** jest typu wyliczeniowego i może przyjmować wartości "FEMALE", "MALE", "OTHER"
- OGR/1/23** Atrybut **dateOfBirth** jest datą
- OGR/1/24** Atrybut **phoneNumber** jest ciągiem znaków o długości od 1 do 50 znaków
- OGR/1/25** Atrybut **streetAddress** jest ciągiem znaków o długości od 1 do 255 znaków
- OGR/1/26** Atrybut **postalCode** jest ciągiem znaków o długości od 1 do 20 znaków
- OGR/1/27** Atrybut **city** jest ciągiem znaków o długości od 1 do 50 znaków
- OGR/1/28** Atrybut **country** jest ciągiem znaków o długości od 1 do 50 znaków
- OGR/1/29** Atrybut **personalDescription** jest ciągiem znaków

Model informacyjny



Rys. 3.27: Diagram klas - poddziedzina administracyjna (opr.wł)

3.3.3. Poddziedzina produkty

Kategorie

KAT/2/01 Product (Produkt)

Opis: Produkt żywieniowy

Atrybuty:

- | | |
|------------|--|
| • id | - identyfikator |
| • source | - źródło produktu, jeśli produkt jest importowany; preferowany format adresu URL |
| • isPublic | - flaga określająca czy produkt jest widoczny publicznie |
| • language | - język tłumaczenia w postaci kodu ISO 639-1 |

KAT/2/02 ProductVersion (Wersja Produktu)

Opis: Wersja produktu. Podczas każdej edycji powstaje nowa wersja. Wersji nie można usunąć, można ją jedynie zastąpić nową wersją

Atrybuty:

- | | |
|---------------|--|
| • id | - identyfikator |
| • createdDate | - czas utworzenia wersji |
| • description | - krótki opis produktu w języku produktu |

KAT/2/03 ProductBasicNutritionData (Podstawowe Składniki Odżywcze Produktu)

Opis: Podstawowe składniki odżywcze produktu

Atrybuty:

• id	- identyfikator
• energy	- energia w kilokaloriach (kcal) na 100 gramów produktu
• protein	- białko w gramach na 100 gramów produktu
• fat	- tłuszcz w gramach na 100 gramów produktu
• carbohydrates	- węglowodany w gramach na 100 gramów produktu

KAT/2/04 NutritionData (Wartość Odżywczna)

Opis: Wartość wartości odżywczej dla konkretnego produktu

Atrybuty:

• id	- identyfikator
• nutritionValue	- ilość składnika odżywczego w jednostkach wyspecyfikowanych w definicji wartości odżywczej

KAT/2/05 NutritionDefinition (Definicja Wartości Odżywczej)

Opis: Definicja wartości odżywczej

Atrybuty:

• id	- identyfikator
• tag	- krótki znacznik reprezentujący wartość odżywczą
• description	- krótki opis wartości odżywczej w języku angielskim
• units	- jednostki wykorzystywane do pomiaru wartości odżywczej, np. "g", "kcal", "ml"
• decimalPlaces	- liczba miejsc dziesiętnych do których wartość składnika odżywczego powinna być zaokrąglana

KAT/2/06 NutritionDefinitionTranslation (Tłumaczenie Definicji Wartości Odżywczej)

Opis: Tłumaczenie definicji wartości odżywczej

Atrybuty:

• id	- identyfikator
• translation	- przetłumaczony opis definicji wartości odżywczej
• language	- język tłumaczenia w postaci kodu ISO 639-1

KAT/2/07 HouseholdMeasure (Miara Domowa)

Opis: Miara domowa produktu z masą w gramach

Atrybuty:

• id	- identyfikator
• description	- krótki opis miary domowej w języku produktu, np. "szklanka" lub "łyżeczka"
• gramsWeight	- masa w gramach 1 jednostki miary domowej
• isVisible	- flaga określająca czy miara jest widoczna podczas wyświetlania produktu

KAT/2/08 ProductSubcategory (Podkategoria Produktu)

Opis: Podkategoria produktu

Atrybuty:

• id	- identyfikator
------	-----------------

- description
 - krótki opis podkategorii w języku produktu
- KAT/2/09 ProductCategory (Kategoria Produktu)**
- Opis:** Główna kategoria produktu
- Atrybuty:**
- id
 - identyfikator
 - description
 - krótki opis kategorii produktu w języku angielskim

- KAT/2/10 ProductCategoryTranslation (Tłumaczenie Kategorii Produktu)**
- Opis:** Tłumaczenie kategorii produktu
- Atrybuty:**
- id
 - identyfikator
 - translation
 - przetłumaczona nazwa kategorii produktu
 - language
 - język tłumaczenia w postaci kodu ISO 639-1

- KAT/2/11 DietType (Typ Diety)**
- Opis:** Typ diety
- Atrybuty:**
- id
 - identyfikator
 - name
 - krótki opis typu diety w języku angielskim

- KAT/2/12 DietTypeTranslation (Tłumaczenie Typu Diety)**
- Opis:** Tłumaczenie typu diety
- Atrybuty:**
- id
 - identyfikator
 - translation
 - tłumaczenie nazwy typu diety
 - language
 - język tłumaczenia w postaci kodu ISO 639-1

Reguły funkcjonowania

Reguły dla KAT/2/01 Product

- REG/2/01** Produkt (**KAT/2/01 Product**) musi mieć przynajmniej jedną wersję (**KAT/2/02 ProductVersion**)
- REG/2/02** Produkt (**KAT/2/01 Product**) może mieć wiele wersji (**KAT/2/02 ProductVersion**)
- REG/2/03** Produkt (**KAT/2/01 Product**) nie musi mieć zdefiniowanego autora (**KAT/1/01 User**)
- REG/2/04** Produkt (**KAT/2/01 Product**) może mieć maksymalnie jednego autora (**KAT/1/01 User**)
- REG/2/05** *Dietetyk* może wyświetlać publiczne produkty (**KAT/2/01 Product**)
- REG/2/06** *Dietetyk* może dodawać, wyświetlać, edytować i usuwać własne produkty (**KAT/2/01 Product**)
- REG/2/07** *Administrator* może wyświetlać i usuwać produkty (**KAT/2/01 Product**)

Reguły dla KAT/2/02 ProductVersion

- REG/2/08** Wersja produktu (**KAT/2/02 ProductVersion**) musi być przypisana do dokładnie jednego produktu (**KAT/2/01 Product**)
- REG/2/09** Wersja produktu (**KAT/2/02 ProductVersion**) musi być przypisana do dokładnie jednych podstawowych wartości odżywcznych (**KAT/2/03 ProductBasicNutritionData**)
- REG/2/10** Wersja produktu (**KAT/2/02 ProductVersion**) nie musi mieć zdefiniowanych żadnych wartości odżywcznych (**KAT/2/04 NutritionData**)
- REG/2/11** Wersja produktu (**KAT/2/02 ProductVersion**) może mieć zdefiniowane wiele wartości odżywcznych (**KAT/2/04 NutritionData**)
- REG/2/12** Wersja produktu (**KAT/2/02 ProductVersion**) nie musi mieć zdefiniowanych żadnych miar domowych (**KAT/2/07 HouseholdMeasure**)
- REG/2/13** Wersja produktu (**KAT/2/02 ProductVersion**) może mieć zdefiniowane wiele miar domowych (**KAT/2/07 HouseholdMeasure**)
- REG/2/14** Wersja produktu (**KAT/2/02 ProductVersion**) musi należeć do dokładnie jednej podkategorii (**KAT/2/08 ProductSubcategory**)
- REG/2/15** Wersja produktu (**KAT/2/02 ProductVersion**) nie musi mieć przypisanego żadnego odpowiedniego typu diety (**KAT/2/11 DietType**)
- REG/2/16** Wersja produktu (**KAT/2/02 ProductVersion**) może mieć przypisanych wiele odpowiednich typów diety (**KAT/2/11 DietType**)
- REG/2/17** Wersja produktu (**KAT/2/02 ProductVersion**) nie musi mieć przypisanego żadnego nieodpowiedniego typu diety (**KAT/2/11 DietType**)
- REG/2/18** Wersja produktu (**KAT/2/02 ProductVersion**) może mieć przypisanych wiele nieodpowiednich typów diety (**KAT/2/11 DietType**)
- REG/2/19** Wersja produktu (**KAT/2/02 ProductVersion**) jest przedmiotem kompozycji ze strony produktu (**KAT/2/01 Product**)
- REG/2/20** Wersja produktu (**KAT/2/02 ProductVersion**) po utworzeniu nie może być edytowana ani usuwana

Reguły dla KAT/2/03 ProductBasicNutritionData

- REG/2/21** Podstawowe wartości odżywcze produktu (**KAT/2/03 ProductBasicNutritionData**) muszą być przypisane do dokładnie jednej wersji produktu (**KAT/2/02 ProductVersion**)
- REG/2/22** Podstawowe wartości odżywcze produktu (**KAT/2/03 ProductBasicNutritionData**) są przedmiotem kompozycji ze strony wersji produktu (**KAT/2/02 ProductVersion**)

Reguły dla KAT/2/04 NutritionData

- REG/2/23** Wartość odżywca (**KAT/2/04 NutritionData**) musi być przypisana do dokładnie jednej wersji produktu (**KAT/2/02 ProductVersion**)
- REG/2/24** Wartość odżywca (**KAT/2/04 NutritionData**) musi być przypisana do dokładnie jednej definicji wartości odżywczej (**KAT/2/05 NutritionDefinition**)
- REG/2/25** Wartość odżywca (**KAT/2/04 NutritionData**) jest przedmiotem kompozycji ze strony wersji produktu (**KAT/2/02 ProductVersion**)

Reguły dla KAT/2/05 NutritionDefinition

- REG/2/26** Definicja wartości odżywczej (**KAT/2/05 NutritionDefinition**) nie musi mieć zdefiniowanego żadnego tłumaczenia (**KAT/2/06 NutritionDefinitionTranslation**)
- REG/2/27** Definicja wartości odżywczej (**KAT/2/05 NutritionDefinition**) może mieć zdefiniowanych wiele tłumaczeń (**KAT/2/06 NutritionDefinitionTranslation**)
- REG/2/28** *Dietetyk* może wyświetlać definicję wartości odżywczej (**KAT/2/05 NutritionDefinition**)
- REG/2/29** *Administrator* może dodawać, wyświetlać, edytować i usuwać definicję wartości odżywczej (**KAT/2/05 NutritionDefinition**)

Reguły dla KAT/2/06 NutritionDefinitionTranslation

- REG/2/30** Tłumaczenie definicji wartości odżywczej (**KAT/2/06 NutritionDefinitionTranslation**) musi być przypisane do dokładnie jednej definicji wartości odżywczej (**KAT/2/05 NutritionDefinition**)
- REG/2/31** Tłumaczenie definicji wartości odżywczej (**KAT/2/06 NutritionDefinitionTranslation**) jest przedmiotem kompozycji ze strony definicji wartości odżywczej (**KAT/2/05 NutritionDefinition**)

Reguły dla KAT/2/07 HouseholdMeasure

- REG/2/32** Miara domowa (**KAT/2/07 HouseholdMeasure**) musi być przypisana do dokładnie jednej wersji produktu (**KAT/2/02 ProductVersion**)
- REG/2/33** Miara domowa (**KAT/2/07 HouseholdMeasure**) jest przedmiotem kompozycji ze strony wersji produktu (**KAT/2/02 ProductVersion**)

Reguły dla KAT/2/08 ProductSubcategory

- REG/2/34** Podkategoria produktu (**KAT/2/08 ProductSubcategory**) musi być przypisana do co najmniej jednej wersji produktu (**KAT/2/02 ProductVersion**)
- REG/2/35** Podkategoria produktu (**KAT/2/08 ProductSubcategory**) może być przypisana do wielu wersji produktu (**KAT/2/02 ProductVersion**)
- REG/2/36** Podkategoria produktu (**KAT/2/08 ProductSubcategory**) musi być przypisana do dokładnie jednej kategorii (**KAT/2/09 ProductCategory**)
- REG/2/37** *Dietetyk* może dodawać i wyświetlać podkategorię produktu (**KAT/2/08 ProductSubcategory**)
- REG/2/38** *Administrator* może wyświetlać podkategorię produktu (**KAT/2/08 ProductSubcategory**)

Reguły dla KAT/2/09 ProductCategory

- REG/2/39** Kategoria produktu (**KAT/2/09 ProductCategory**) nie musi mieć przypisanego żadnego tłumaczenia (**KAT/2/10 ProductCategoryTranslation**)
- REG/2/40** Kategoria produktu (**KAT/2/09 ProductCategory**) może mieć przypisanych wiele tłumaczeń (**KAT/2/10 ProductCategoryTranslation**)
- REG/2/41** *Dietetyk* może wyświetlać kategorię produktu (**KAT/2/09 ProductCategory**)

REG/2/42 *Administrator* może dodawać, wyświetlać, edytować i usuwać kategorię produktu (**KAT/2/09 ProductCategory**)

Reguły dla KAT/2/10 ProductCategoryTranslation

- REG/2/43** Tłumaczenie kategorii produktu (**KAT/2/10 ProductCategoryTranslation**) musi być przypisane do dokładnie jednej kategorii (**KAT/2/09 ProductCategory**)
- REG/2/44** Tłumaczenie kategorii produktu (**KAT/2/10 ProductCategoryTranslation**) jest przedmiotem kompozycji ze strony kategorii (**KAT/2/09 ProductCategory**)

Reguły dla KAT/2/11 DietType

- REG/2/45** Typ diety (**KAT/2/11 DietType**) nie musi mieć zdefiniowanego żadnego tłumaczenia (**KAT/2/12 DietTypeTranslation**)
- REG/2/46** Typ diety (**KAT/2/11 DietType**) może mieć zdefiniowanych wiele tłumaczeń (**KAT/2/12 DietTypeTranslation**)
- REG/2/47** *Dietetyk* może wyświetlać typ diety (**KAT/2/11 DietType**)
- REG/2/48** *Administrator* może dodawać, wyświetlać, edytować i usuwać typ diety (**KAT/2/11 DietType**)

Reguły dla KAT/2/12 DietTypeTranslation

- REG/2/49** Tłumaczenie typu diety (**KAT/2/12 DietTypeTranslation**) musi być przypisane do dokładnie jednego typu diety (**KAT/2/11 DietType**)
- REG/2/50** Tłumaczenie typu diety (**KAT/2/12 DietTypeTranslation**) jest przedmiotem kompozycji ze strony typu diety (**KAT/2/11 DietType**)

Ograniczenia dziedzinowe

Ograniczenia dla KAT/2/01 Product

- OGR/2/01** Atrybut **isPublic** jest wymagany
- OGR/2/02** Atrybut **source** jest ciągiem znaków o długości od 1 do 255 znaków
- OGR/2/03** Atrybut **isPublic** jest typu logicznego

Ograniczenia dla KAT/2/02 ProductVersion

- OGR/2/04** Atrybut **createdDate** jest wymagany
- OGR/2/05** Atrybut **description** jest wymagany
- OGR/2/06** Atrybut **createdDate** jest stemplem czasowym
- OGR/2/07** Atrybut **description** jest ciągiem znaków o długości od 1 do 255 znaków

Ograniczenia dla KAT/2/03 ProductBasicNutritionData

- OGR/2/08** Atrybut **energy** jest wymagany
- OGR/2/09** Atrybut **protein** jest wymagany
- OGR/2/10** Atrybut **fat** jest wymagany
- OGR/2/11** Atrybut **carbohydrates** jest wymagany
- OGR/2/12** Atrybut **energy** jest liczbą rzeczywistą nie mniejszą niż 0
- OGR/2/13** Atrybut **protein** jest liczbą rzeczywistą nie mniejszą niż 0

OGR/2/14 Atrybut **fat** jest liczbą rzeczywistą nie mniejszą niż 0

OGR/2/15 Atrybut **carbohydrates** jest liczbą rzeczywistą nie mniejszą niż 0

Ograniczenia dla KAT/2/04 NutritionData

OGR/2/16 Atrybut **nutritionValue** jest wymagany

OGR/2/17 Atrybut **nutritionValue** jest liczbą rzeczywistą nie mniejszą niż 0

Ograniczenia dla KAT/2/05 NutritionDefinition

OGR/2/18 Atrybut **tag** jest wymagany

OGR/2/19 Atrybut **description** jest wymagany

OGR/2/20 Atrybut **units** jest wymagany

OGR/2/21 Atrybut **decimalPlaces** jest wymagany

OGR/2/22 Atrybut **tag** ma unikalną wartość

OGR/2/23 Atrybut **tag** jest ciągiem znaków o długości od 1 do 20 znaków

OGR/2/24 Atrybut **description** jest ciągiem znaków o długości od 1 do 255 znaków

OGR/2/25 Atrybut **units** jest ciągiem znaków o długości od 1 do 10 znaków

OGR/2/26 Atrybut **decimalPlaces** jest liczbą całkowitą nie mniejszą niż 0

Ograniczenia dla KAT/2/06 NutritionDefinitionTranslation

OGR/2/27 Atrybut **translation** jest wymagany

OGR/2/28 Atrybut **translation** jest ciągiem znaków o długości od 1 do 255 znaków

Ograniczenia dla KAT/2/07 HouseholdMeasure

OGR/2/29 Atrybut **description** jest wymagany

OGR/2/30 Atrybut **gramsWeight** jest wymagany

OGR/2/31 Atrybut **isVisible** jest wymagany

OGR/2/32 Atrybut **description** jest ciągiem znaków o długości od 1 do 255 znaków

OGR/2/33 Atrybut **gramsWeight** jest liczbą rzeczywistą nie mniejszą niż 0

OGR/2/34 Atrybut **isVisible** jest typu logicznego

Ograniczenia dla KAT/2/08 ProductSubcategory

OGR/2/35 Atrybut **description** jest wymagany

OGR/2/36 Atrybut **description** jest ciągiem znaków o długości od 1 do 255 znaków

Ograniczenia dla KAT/2/09 ProductCategory

OGR/2/37 Atrybut **description** jest wymagany

OGR/2/38 Atrybut **description** ma unikalną wartość

OGR/2/39 Atrybut **description** jest ciągiem znaków o długości od 1 do 255 znaków

Ograniczenia dla KAT/2/10 ProductCategoryTranslation

OGR/2/40 Atrybut **translation** jest wymagany

OGR/2/41 Atrybut **translation** jest ciągiem znaków o długości od 1 do 255 znaków

Ograniczenia dla KAT/2/11 DietType

OGR/2/42 Atrybut **name** jest wymagany

OGR/2/43 Atrybut **name** ma unikalną wartość

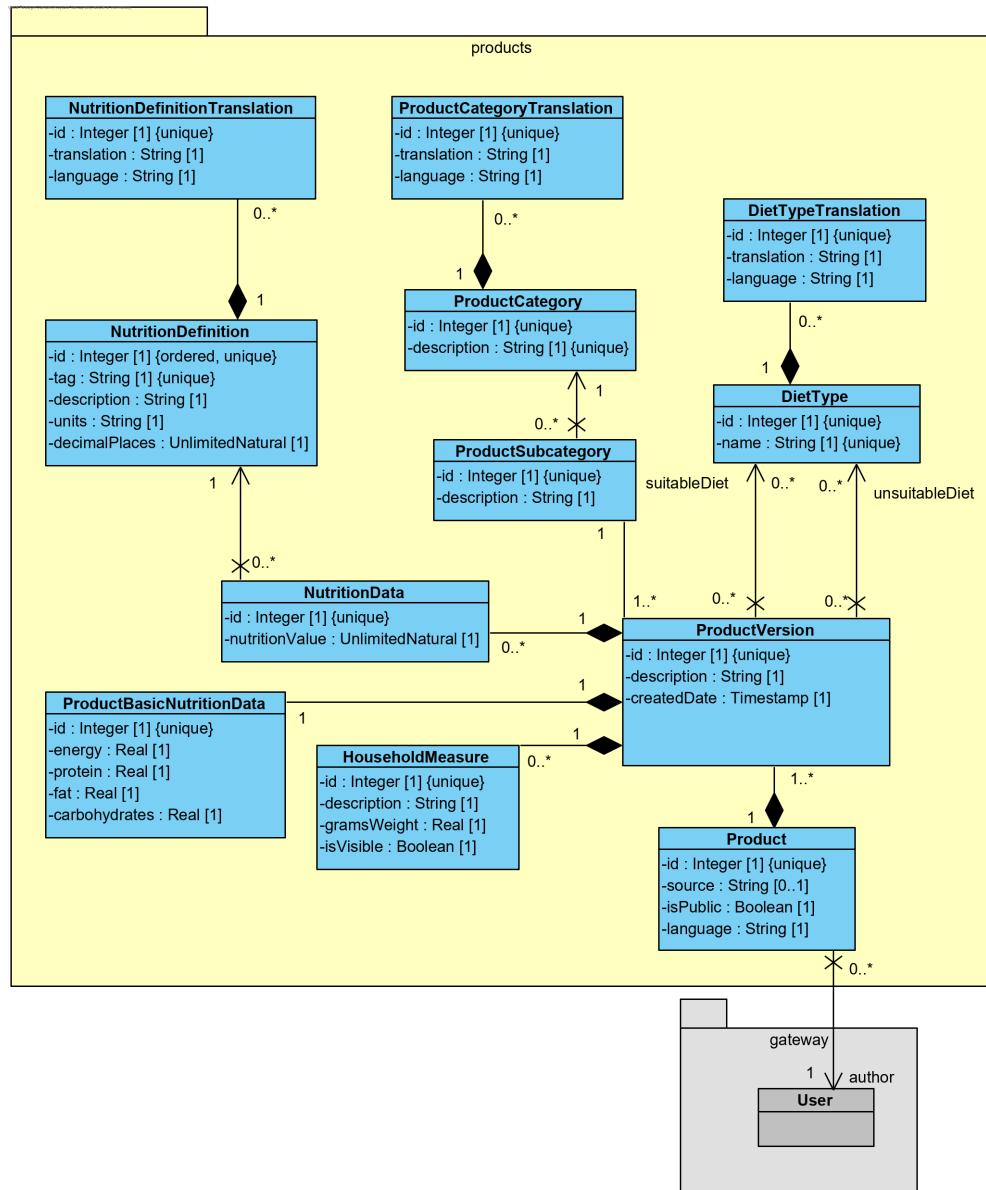
OGR/2/44 Atrybut **name** jest ciągiem znaków o długości od 1 do 255 znaków

Ograniczenia dla KAT/2/12 DietTypeTranslation

OGR/2/45 Atrybut **translation** jest wymagany

OGR/2/46 Atrybut **translation** jest ciągiem znaków o długości od 1 do 255 znaków

Model informacyjny



Rys. 3.28: Diagram klas - poddziedzina produkty (opr.wł)

3.3.4. Poddziedzina przepisy

Kategorie

KAT/3/01 Recipe (Przepis)

Opis: Przepis

Atrybuty:

- | | |
|--|---|
| <ul style="list-style-type: none">• id• isPublic• language | <ul style="list-style-type: none">- identyfikator- flaga określająca czy przepis jest widoczny publicznie- język tłumaczenia w postaci kodu ISO 639-1 |
|--|---|

KAT/3/02 RecipeVersion (Wersja Przepisu)

Opis: Wersja przepisu. Podczas każdej edycji powstaje nowa wersja. Wersji nie można usunąć, można ją jedynie zastąpić nową wersją

Atrybuty:

- | | |
|--|---|
| <ul style="list-style-type: none">• id• editTimestamp• name• preparationTimeMinutes• numberOfPortions• image• totalGramsWeight | <ul style="list-style-type: none">- identyfikator- czas utworzenia wersji- nazwa przepisu w języku przepisu- średni czas potrzebny na całkowite przygotowanie przepisu, zdefiniowany w minutach- liczba porcji dla których przepis jest zdefiniowany- opcjonalne zdjęcie dania przygotowanego na podstawie przepisu- całkowita masa w gramach dania przygotowanego z przepisu |
|--|---|

KAT/3/03 RecipeBasicNutritionData (Podstawowe Wartości Odżywcze Przepisu)

Opis: Podstawowe wartości odżywcze przepisu

Atrybuty:

- | | |
|---|--|
| <ul style="list-style-type: none">• id• energy• protein• fat• carbohydrates | <ul style="list-style-type: none">- identyfikator- energia w kilokaloriach (kcal) na 100 gramów posiłku przygotowanego z użyciem produktów w przepisie- białko w gramach na 100 gramów posiłku przygotowanego z użyciem produktów w przepisie- tłuszcze w gramach na 100 gramów posiłku przygotowanego z użyciem produktów w przepisie- węglowodany w gramach na 100 gramów posiłku przygotowanego z użyciem produktów w przepisie |
|---|--|

KAT/3/04 RecipeSection (Sekcja Przepisu)

Opis: Sekcja przepisu, np. sernik może mieć 3 sekcje odpowiednio dla ciasta, masy i polewy

Atrybuty:

- | | |
|--|--|
| <ul style="list-style-type: none">• id• sectionName | <ul style="list-style-type: none">- identyfikator- nazwa przepisu w języku przepisu |
|--|--|

KAT/3/05 ProductPortion (Porcja Produktu)

Opis: Porcja produktu wykorzystywana w przepisie

Atrybuty:

- | | |
|---|---|
| <ul style="list-style-type: none">● id● amount | <ul style="list-style-type: none">- identyfikator- ilość produktu w jednostkach miary domowej lub w gramach jeśli miara domowa nie jest zdefiniowana |
|---|---|

KAT/3/06 PreparationStep (Krok Przygotowania)

Opis: Krok przygotowania w przepisie

Atrybuty:

- | | |
|--|--|
| <ul style="list-style-type: none">● id● ordinalNumber● stepDescription | <ul style="list-style-type: none">- identyfikator- liczba porządkowa kroku przygotowania- w miarę możliwości krótki opis kroku przygotowania |
|--|--|

KAT/3/07 KitchenAppliance (Sprzęt Kuchenny)

Opis: Definicja sprzętu kuchennego

Atrybuty:

- | | |
|---|--|
| <ul style="list-style-type: none">● id● name | <ul style="list-style-type: none">- identyfikator- nazwa sprzętu kuchennego w języku angielskim |
|---|--|

KAT/3/08 KitchenApplianceTranslation (Tłumaczenie Sprzętu Kuchennego)

Opis: Tłumaczenie sprzętu kuchennego

Atrybuty:

- | | |
|---|--|
| <ul style="list-style-type: none">● id● translation● language | <ul style="list-style-type: none">- identyfikator- przetłumaczona nazwa sprzętu kuchennego- język tłumaczenia w postaci kodu ISO 639-1 |
|---|--|

KAT/3/09 DishType (Typ Dania)

Opis: Typ dania, np. sałatka lub zupa

Atrybuty:

- | | |
|--|---|
| <ul style="list-style-type: none">● id● description | <ul style="list-style-type: none">- identyfikator- opis typu dania w języku angielskim |
|--|---|

KAT/3/10 DishTypeTranslation (Tłumaczenie Typu Dania)

Opis: Tłumaczenie typu dania

Atrybuty:

- | | |
|---|--|
| <ul style="list-style-type: none">● id● translation● language | <ul style="list-style-type: none">- identyfikator- przetłumaczona nazwa typu dania- język tłumaczenia w postaci kodu ISO 639-1 |
|---|--|

KAT/3/11 MealType (Typ Posiłku)

Opis: Typ posiłku, np. śniadanie lub obiad

Atrybuty:

- | | |
|---|--|
| <ul style="list-style-type: none">● id● name | <ul style="list-style-type: none">- identyfikator- nazwa typu posiłku w języku angielskim |
|---|--|

KAT/3/12 MealTypeTranslation (Tłumaczenie Typu Posiłku)

Opis: Meal type translation

Atrybuty:

- | | |
|---------------|--|
| • id | - identyfikator |
| • translation | - przetłumaczona nazwa typu posiłku |
| • language | - język tłumaczenia w postaci kodu ISO 639-1 |

Reguły funkcjonowania

Reguły dla KAT/3/01 Recipe

- REG/3/01** Przepis (**KAT/3/01 Recipe**) nie musi mieć zdefiniowanego żadnego przepisu źródłowego (**KAT/3/01 Recipe**)
- REG/3/02** Przepis (**KAT/3/01 Recipe**) może mieć zdefiniowany maksymalnie jeden przepis źródłowy (**KAT/3/01 Recipe**)
- REG/3/03** Przepis (**KAT/3/01 Recipe**) musi mieć przynajmniej jedną wersję (**KAT/3/02 RecipeVersion**)
- REG/3/04** Przepis (**KAT/3/01 Recipe**) może mieć wiele wersji (**KAT/3/02 RecipeVersion**)
- REG/3/05** Przepis (**KAT/3/01 Recipe**) nie musi mieć zdefiniowanego autora (**KAT/1/01 User**)
- REG/3/06** Przepis (**KAT/3/01 Recipe**) może mieć maksymalnie jednego autora (**KAT/1/01 User**)
- REG/3/07** *Dietetyk* może wyświetlać publiczne przepisy (**KAT/3/01 Recipe**)
- REG/3/08** *Dietetyk* może dodawać, wyświetlać, edytować i usuwać własne przepisy (**KAT/3/01 Recipe**)
- REG/3/09** *Administrator* może wyświetlać i usuwać przepisy (**KAT/3/01 Recipe**)

Reguły dla KAT/3/02 RecipeVersion

- REG/3/10** Wersja przepisu (**KAT/3/02 RecipeVersion**) musi mieć dokładnie jedne podstawowe wartości odżywcze przepisu (**KAT/3/03 RecipeBasicNutritionData**)
- REG/3/11** Wersja przepisu (**KAT/3/02 RecipeVersion**) musi mieć przynajmniej jedną sekcję (**KAT/3/04 RecipeSection**)
- REG/3/12** Wersja przepisu (**KAT/3/02 RecipeVersion**) może mieć wiele sekcji (**KAT/3/04 RecipeSection**)
- REG/3/13** Wersja przepisu (**KAT/3/02 RecipeVersion**) nie musi mieć przypisanego żadnego sprzętu kuchennego (**KAT/3/07 KitchenAppliance**)
- REG/3/14** Wersja przepisu (**KAT/3/02 RecipeVersion**) może mieć przypisanych wiele sprzętów kuchennych (**KAT/3/07 KitchenAppliance**)
- REG/3/15** Wersja przepisu (**KAT/3/02 RecipeVersion**) nie musi mieć przypisanego żadnego typu dania (**KAT/3/09 DishType**)
- REG/3/16** Wersja przepisu (**KAT/3/02 RecipeVersion**) może mieć przypisanych wiele typów dań (**KAT/3/09 DishType**)
- REG/3/17** Wersja przepisu (**KAT/3/02 RecipeVersion**) nie musi mieć przypisanego żadnego typu posiłku (**KAT/3/11 MealType**)
- REG/3/18** Wersja przepisu (**KAT/3/02 RecipeVersion**) może mieć przypisanych wiele typów posiłków (**KAT/3/11 MealType**)

- REG/3/19** Wersja przepisu (**KAT/3/02 RecipeVersion**) nie musi mieć przypisanego żadnego odpowiedniego typu diety (**KAT/2/11 DietType**)
- REG/3/20** Wersja przepisu (**KAT/3/02 RecipeVersion**) może mieć przypisanych wiele odpowiednich typów diety (**KAT/2/11 DietType**)
- REG/3/21** Wersja przepisu (**KAT/3/02 RecipeVersion**) nie musi mieć przypisanego żadnego nieodpowiedniego typu diety (**KAT/2/11 DietType**)
- REG/3/22** Wersja przepisu (**KAT/3/02 RecipeVersion**) może mieć przypisanych wiele nieodpowiednich typów diety (**KAT/2/11 DietType**)
- REG/3/23** Wersja przepisu (**KAT/3/02 RecipeVersion**) jest przedmiotem kompozycji ze strony przepisu (**KAT/3/01 Recipe**)
- REG/3/24** Wersja przepisu (**KAT/3/02 RecipeVersion**) po utworzeniu nie może być edytowana ani usuwana

Reguły dla KAT/3/03 RecipeBasicNutritionData

- REG/3/25** Podstawowe wartości odżywcze przepisu (**KAT/3/03 RecipeBasicNutritionData**) muszą być przypisane do dokładnie jednej wersji przepisu (**KAT/3/02 RecipeVersion**)
- REG/3/26** Podstawowe wartości odżywcze przepisu (**KAT/3/03 RecipeBasicNutritionData**) są przedmiotem kompozycji ze strony wersji przepisu (**KAT/3/02 RecipeVersion**)

Reguły dla KAT/3/04 RecipeSection

- REG/3/27** Sekcja przepisu (**KAT/3/04 RecipeSection**) musi być przypisana do dokładniej jednej wersji przepisu (**KAT/3/02 RecipeVersion**)
- REG/3/28** Sekcja przepisu (**KAT/3/04 RecipeSection**) musi mieć przypisaną przynajmniej jedną porcję produktu (**KAT/3/05 ProductPortion**)
- REG/3/29** Sekcja przepisu (**KAT/3/04 RecipeSection**) może mieć przypisanych wiele porcji produktu (**KAT/3/05 ProductPortion**)
- REG/3/30** Sekcja przepisu (**KAT/3/04 RecipeSection**) musi mieć przypisany przynajmniej jeden krok przygotowania (**KAT/3/06 PreparationStep**)
- REG/3/31** Sekcja przepisu (**KAT/3/04 RecipeSection**) może mieć zdefiniowanych wiele kroków przygotowania (**KAT/3/06 PreparationStep**)
- REG/3/32** Sekcja przepisu (**KAT/3/04 RecipeSection**) jest przedmiotem kompozycji ze strony wersji przepisu (**KAT/3/02 RecipeVersion**)

Reguły dla KAT/3/05 ProductPortion

- REG/3/33** Porcja produktu (**KAT/3/05 ProductPortion**) musi być przypisana do dokładnie jednej sekcji przepisu (**KAT/3/04 RecipeSection**)
- REG/3/34** Porcja produktu (**KAT/3/05 ProductPortion**) musi mieć przypisany dokładnie jeden produkt (**KAT/2/01 Product**)
- REG/3/35** Porcja produktu (**KAT/3/05 ProductPortion**) nie musi mieć przypisanej miary domowej (**KAT/2/07 HouseholdMeasure**)
- REG/3/36** Porcja produktu (**KAT/3/05 ProductPortion**) może mieć przypisaną maksymalnie jedną miarę domową (**KAT/2/07 HouseholdMeasure**)
- REG/3/37** Porcja produktu (**KAT/3/05 ProductPortion**) jest przedmiotem kompozycji ze strony sekcji przepisu (**KAT/3/04 RecipeSection**)

Reguły dla KAT/3/06 PreparationStep

- REG/3/38** Krok przygotowania (**KAT/3/06 PreparationStep**) musi być przypisany do dokładnie jednej sekcji przepisu (**KAT/3/04 RecipeSection**)
- REG/3/39** Krok przygotowania (**KAT/3/06 PreparationStep**) jest przedmiotem kompozycji ze strony sekcji przepisu (**KAT/3/04 RecipeSection**)

Reguły dla KAT/3/07 KitchenAppliance

- REG/3/40** Sprzęt kuchenny (**KAT/3/07 KitchenAppliance**) nie musi mieć zdefiniowanego żadnego tłumaczenia (**KAT/3/08 KitchenApplianceTranslation**)
- REG/3/41** Sprzęt kuchenny (**KAT/3/07 KitchenAppliance**) może mieć zdefiniowanych wiele tłumaczeń (**KAT/3/08 KitchenApplianceTranslation**)
- REG/3/42** *Pacjent* może wyświetlać sprzęt kuchenny (**KAT/3/07 KitchenAppliance**)
- REG/3/43** *Dietetyk* może wyświetlać sprzęt kuchenny (**KAT/3/07 KitchenAppliance**)
- REG/3/44** *Administrator* może dodawać, wyświetlać, edytować i usuwać sprzęt kuchenny (**KAT/3/07 KitchenAppliance**)

Reguły dla KAT/3/08 KitchenApplianceTranslation

- REG/3/45** Tłumaczenie sprzętu kuchennego (**KAT/3/08 KitchenApplianceTranslation**) musi być przypisane do dokładnie jednego sprzętu kuchennego (**KAT/3/07 KitchenAppliance**)
- REG/3/46** Tłumaczenie sprzętu kuchennego (**KAT/3/08 KitchenApplianceTranslation**) jest przedmiotem kompozycji ze strony sprzętu kuchennego (**KAT/3/07 KitchenAppliance**)

Reguły dla KAT/3/09 DishType

- REG/3/47** Typ dania (**KAT/3/09 DishType**) nie musi mieć zdefiniowanego żadnego tłumaczenia (**KAT/3/10 DishTypeTranslation**)
- REG/3/48** Typ dania (**KAT/3/09 DishType**) może mieć zdefiniowanych wiele tłumaczeń (**KAT/3/10 DishTypeTranslation**)
- REG/3/49** *Dietetyk* może wyświetlać typ dania (**KAT/3/09 DishType**)
- REG/3/50** *Administrator* może dodawać, wyświetlać, edytować i usuwać typ dania (**KAT/3/09 DishType**)

Reguły dla KAT/3/10 DishTypeTranslation

- REG/3/51** Tłumaczenie typu dania (**KAT/3/10 DishTypeTranslation**) musi być przypisane do dokładnie jednego typu dania (**KAT/3/09 DishType**)
- REG/3/52** Tłumaczenie typu dania (**KAT/3/10 DishTypeTranslation**) jest przedmiotem kompozycji ze strony typu dania (**KAT/3/09 DishType**)

Reguły dla KAT/3/11 MealType

- REG/3/53** Typ posiłku (**KAT/3/11 MealType**) nie musi mieć zdefiniowanego żadnego tłumaczenia (**KAT/3/12 MealTypeTranslation**)
- REG/3/54** Typ posiłku (**KAT/3/11 MealType**) może mieć zdefiniowanych wiele tłumaczeń (**KAT/3/12 MealTypeTranslation**)
- REG/3/55** *Pacjent* może wyświetlać typ posiłku (**KAT/3/11 MealType**)
- REG/3/56** *Dietetyk* może wyświetlać typ posiłku (**KAT/3/11 MealType**)

REG/3/57 *Administrator* może dodawać, wyświetlać, edytować i usuwać typ posiłku
(KAT/3/11 MealType)

Reguły dla KAT/3/12 MealTypeTranslation

- REG/3/58** Tłumaczenie typu posiłku (**KAT/3/12 MealTypeTranslation**) musi być przypisane do dokładnie jednego typu posiłku (**KAT/3/11 MealType**)
- REG/3/59** Tłumaczenie typu posiłku (**KAT/3/12 MealTypeTranslation**) jest przedmiotem kompozycji ze strony typu posiłku (**KAT/3/11 MealType**)

Ograniczenia dziedzinowe

Ograniczenia dla KAT/3/01 Recipe

- OGR/3/01** Atrybut **isPublic** jest wymagany
- OGR/3/02** Atrybut **isPublic** jest typu logicznego

Ograniczenia dla KAT/3/02 RecipeVersion

- OGR/3/03** Atrybut **name** jest wymagany
- OGR/3/04** Atrybut **preparationTimeMinutes** jest wymagany
- OGR/3/05** Atrybut **numberOfPortions** jest wymagany
- OGR/3/06** Atrybut **totalGramsWeight** jest wymagany
- OGR/3/07** Atrybut **editTimestamp** jest stemplem czasowym
- OGR/3/08** Atrybut **name** jest ciągiem znaków o długości od 1 do 255 znaków
- OGR/3/09** Atrybut **preparationTimeMinutes** jest liczbą całkowitą nie mniejszą niż 0
- OGR/3/10** Atrybut **numberOfPortions** jest liczbą rzeczywistą nie mniejszą niż 0
- OGR/3/11** Atrybut **image** jest zdjęciem o maksymalnym rozmiarze 5000000 bajtów
- OGR/3/12** Atrybut **totalGramsWeight** jest liczbą rzeczywistą nie mniejszą niż 0

Ograniczenia dla KAT/3/03 RecipeBasicNutritionData

- OGR/3/13** Atrybut **energy** jest wymagany
- OGR/3/14** Atrybut **protein** jest wymagany
- OGR/3/15** Atrybut **fat** jest wymagany
- OGR/3/16** Atrybut **carbohydrates** jest wymagany
- OGR/3/17** Atrybut **energy** jest liczbą całkowitą nie mniejszą niż 0
- OGR/3/18** Atrybut **protein** jest liczbą całkowitą nie mniejszą niż 0
- OGR/3/19** Atrybut **fat** jest liczbą całkowitą nie mniejszą niż 0
- OGR/3/20** Atrybut **carbohydrates** jest liczbą całkowitą nie mniejszą niż 0

Ograniczenia dla KAT/3/04 RecipeSection

- OGR/3/21** Atrybut **sectionName** jest ciągiem znaków o długości od 1 do 255 znaków

Ograniczenia dla KAT/3/05 ProductPortion

- OGR/3/22** Atrybut **amount** jest wymagany
- OGR/3/23** Atrybut **amount** jest liczbą rzeczywistą nie mniejszą niż 0

Ograniczenia dla KAT/3/06 PreparationStep

- OGR/3/24** Atrybut **ordinalNumber** jest wymagany
- OGR/3/25** Atrybut **ordinalNumber** jest liczbą całkowitą nie mniejszą niż 1
- OGR/3/26** Atrybut **stepDescription** jest ciągiem znaków

Ograniczenia dla KAT/3/07 KitchenAppliance

- OGR/3/27** Atrybut **name** jest wymagany
- OGR/3/28** Atrybut **name** ma unikalną wartość
- OGR/3/29** Atrybut **name** jest ciągiem znaków o długości od 1 do 255 znaków

Ograniczenia dla KAT/3/08 KitchenApplianceTranslation

- OGR/3/30** Atrybut **translation** jest wymagany
- OGR/3/31** Atrybut **translation** jest ciągiem znaków o długości od 1 do 255 znaków

Ograniczenia dla KAT/3/09 DishType

- OGR/3/32** Atrybut **description** jest wymagany
- OGR/3/33** Atrybut **description** ma unikalną wartość
- OGR/3/34** Atrybut **description** jest ciągiem znaków o długości od 1 do 255 znaków

Ograniczenia dla KAT/3/10 DishTypeTranslation

- OGR/3/35** Atrybut **translation** jest wymagany
- OGR/3/36** Atrybut **translation** jest ciągiem znaków o długości od 1 do 255 znaków

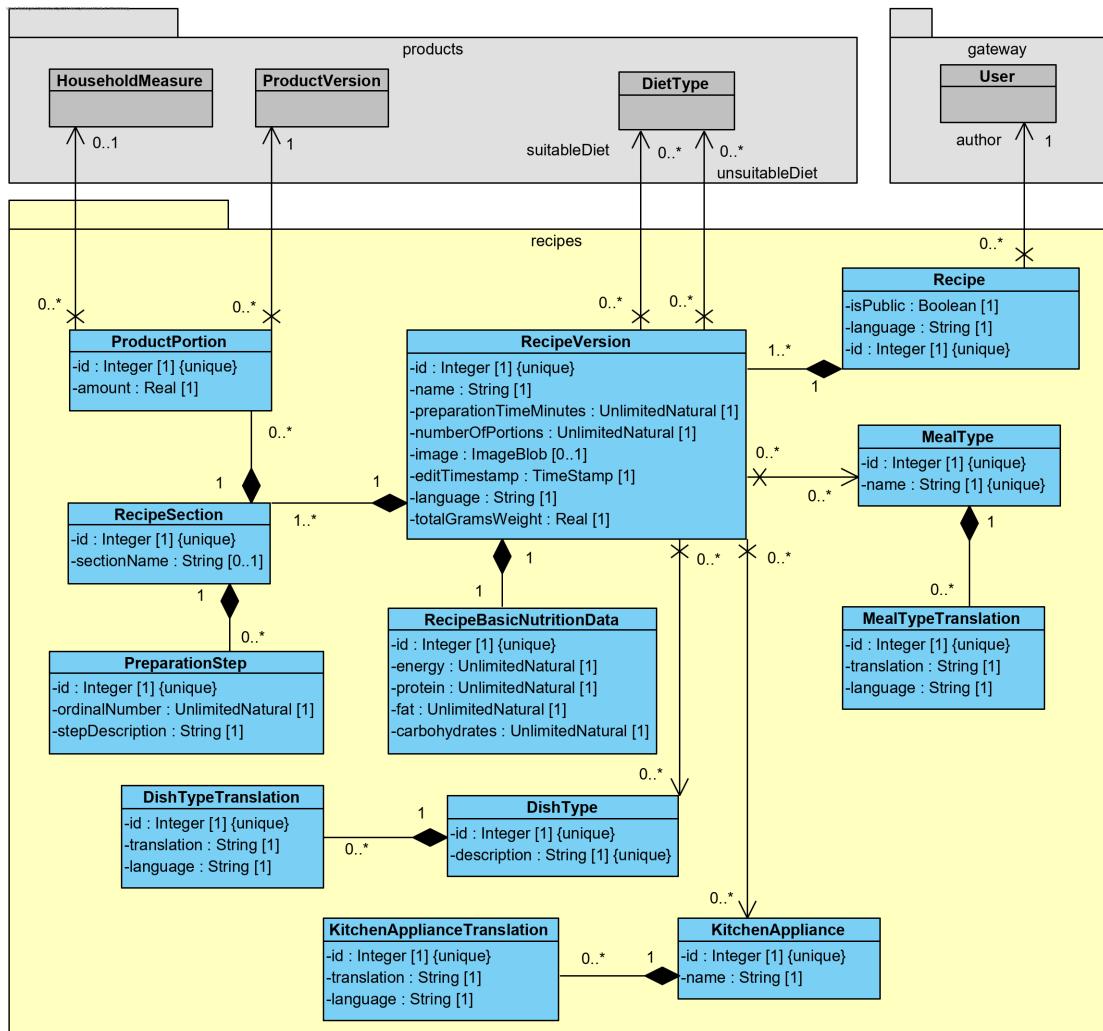
Ograniczenia dla KAT/3/11 MealType

- OGR/3/37** Atrybut **name** jest wymagany
- OGR/3/38** Atrybut **name** ma unikalną wartość
- OGR/3/39** Atrybut **name** jest ciągiem znaków o długości od 1 do 255 znaków

Ograniczenia dla KAT/3/12 MealTypeTranslation

- OGR/3/40** Atrybut **translation** jest wymagany
- OGR/3/41** Atrybut **translation** jest ciągiem znaków o długości od 1 do 255 znaków

Model informacyjny



Rys. 3.29: Diagram klas - poddziedzina przepisy (opr.wł)

3.3.5. Poddziedzina jadłospisy

Kategorie

KAT/4/01 MealPlan (Jadłospis)

Opis: Jadłospis; plan posiłków z podziałem na dnie i posiłki

Atrybuty:

- | | |
|---------------------|--|
| • id | - identyfikator |
| • creationTimestamp | - czas utworzenia jadłospisu |
| • editTimestamp | - czas ostatniej edycji jadłospisu |
| • name | - nazwa jadłospisu |
| • isVisible | - flaga określająca czy jadłospis jest widoczny na liście jadłospisów autora |
| • language | - język tłumaczenia w postaci kodu ISO 639-1 |

- | | |
|---------------------------------------|---|
| • <code>numberOfDays</code> | - liczba dni planu |
| • <code>numberOfMealsPerDay</code> | - liczba posiłków w ciągu dnia |
| • <code>totalDailyEnergy</code> | - całkowita oczekiwana energia w ciągu dnia w kilokaloriach |
| • <code>percentOfProtein</code> | - procent białka w całkowitej dziennej energii |
| • <code>percentOfFat</code> | - procent tłuszczy w całkowitej dziennej energii |
| • <code>percentOfCarbohydrates</code> | - procent węglowodanów w całkowitej dziennej energii |

KAT/4/02 MealPlanDay (Dzień Jadłospisu)

Opis: Dzień w jadłospisie

Atrybuty:

- | | |
|------------------------------|-------------------------|
| • <code>id</code> | - identyfikator |
| • <code>ordinalNumber</code> | - numer porządkowy dnia |

KAT/4/03 Meal (Posiłek)

Opis: Posiłek

Atrybuty:

- | | |
|------------------------------|----------------------------|
| • <code>id</code> | - identyfikator |
| • <code>ordinalNumber</code> | - numer porządkowy posiłku |

KAT/4/04 MealRecipe (Przepis Posiłku)

Opis: Przepis przypisany do posiłku

Atrybuty:

- | | |
|-----------------------|----------------------------|
| • <code>id</code> | - identyfikator |
| • <code>amount</code> | - ilość przepisu w gramach |

KAT/4/05 MealProduct (Produkt Posiłku)

Opis: Produkt przypisany do posiłku

Atrybuty:

- | | |
|-----------------------|---|
| • <code>id</code> | - identyfikator |
| • <code>amount</code> | - ilość produktu w jednostkach miary domowej lub w gramach jeśli miara domowa nie jest zdefiniowana |

KAT/4/06 MealDefinition (Definicja Posiłku)

Opis: Definicja posiłku wykorzystywana do określenia właściwości każdego posiłku w ciągu dnia

Atrybuty:

- | | |
|--------------------------------|---|
| • <code>id</code> | - identyfikator |
| • <code>ordinalNumber</code> | - dzienny numer porządkowy posiłku |
| • <code>timeOfDay</code> | - typowy czas posiłku w formacie 24h w postaci: HH:mm |
| • <code>percentOfEnergy</code> | - część dziennej całkowitej dziennej energii w procentach |

Reguły funkcjonowania

Reguły dla KAT/4/01 MealPlan

- REG/4/01** Jadłospis (**KAT/4/01 MealPlan**) musi mieć przypisany przynajmniej jeden dzień (**KAT/4/02 MealPlanDay**)
- REG/4/02** Jadłospis (**KAT/4/01 MealPlan**) może mieć przypisanych maksymalnie 31 dni (**KAT/4/02 MealPlanDay**)
- REG/4/03** Jadłospis (**KAT/4/01 MealPlan**) musi mieć przypisaną przynajmniej jedną definicję posiłku (**KAT/4/06 MealDefinition**)
- REG/4/04** Jadłospis (**KAT/4/01 MealPlan**) może mieć przypisanych maksymalnie 10 definicji posiłków (**KAT/4/06 MealDefinition**)
- REG/4/05** Jadłospis (**KAT/4/01 MealPlan**) nie musi mieć przypisanego żadnego odpowiedniego typu diety (**KAT/2/11 DietType**)
- REG/4/06** Jadłospis (**KAT/4/01 MealPlan**) może mieć przypisanych wiele odpowiednich typów diety (**KAT/2/11 DietType**)
- REG/4/07** Jadłospis (**KAT/4/01 MealPlan**) nie musi mieć przypisanego żadnego nieodpowiedniego typu diety (**KAT/2/11 DietType**)
- REG/4/08** Jadłospis (**KAT/4/01 MealPlan**) może mieć przypisanych wiele nieodpowiednich typów diety (**KAT/2/11 DietType**)
- REG/4/09** Jadłospis (**KAT/4/01 MealPlan**) musi mieć dokładnie jednego autora (**KAT/1/01 User**)
- REG/4/10** Dietetyk może wyświetlać publiczne jadłospisy (**KAT/4/01 MealPlan**)
- REG/4/11** Dietetyk może dodawać, wyświetlać, edytować i usuwać własne jadłospisy (**KAT/4/01 MealPlan**)
- REG/4/12** Administrator może wyświetlać i usuwać jadłospisy (**KAT/4/01 MealPlan**)

Reguły dla KAT/4/02 MealPlanDay

- REG/4/13** Dzień jadłospisu (**KAT/4/02 MealPlanDay**) musi być przypisany do dokładnie jednego jadłospisu (**KAT/4/01 MealPlan**)
- REG/4/14** Dzień jadłospisu (**KAT/4/02 MealPlanDay**) nie musi mieć przypisanego żadnego posiłku (**KAT/4/03 Meal**)
- REG/4/15** Dzień jadłospisu (**KAT/4/02 MealPlanDay**) może mieć przypisanych maksymalnie 10 posiłków (**KAT/4/03 Meal**)
- REG/4/16** Dzień jadłospisu (**KAT/4/02 MealPlanDay**) jest przedmiotem kompozycji ze strony jadłospisu (**KAT/4/01 MealPlan**)

Reguły dla KAT/4/03 Meal

- REG/4/17** Posiłek (**KAT/4/03 Meal**) musi być przypisany do dokładnie jednego dnia jadłospisu (**KAT/4/02 MealPlanDay**)
- REG/4/18** Posiłek (**KAT/4/03 Meal**) nie musi mieć przypisanego żadnego produktu (**KAT/4/05 MealProduct**)
- REG/4/19** Posiłek (**KAT/4/03 Meal**) może mieć przypisanych wiele produktów (**KAT/4/05 MealProduct**)
- REG/4/20** Posiłek (**KAT/4/03 Meal**) nie musi mieć przypisanego żadnego przepisu (**KAT/4/04 MealRecipe**)
- REG/4/21** Posiłek (**KAT/4/03 Meal**) może mieć przypisanych wiele przepisów (**KAT/4/04 MealRecipe**)

REG/4/22 Posiłek (**KAT/4/03 Meal**) jest przedmiotem kompozycji ze strony dnia jadłospisu (**KAT/4/02 MealPlanDay**)

Reguły dla KAT/4/04 MealRecipe

- REG/4/23** Przepis posiłku (**KAT/4/04 MealRecipe**) musi być przypisany do dokładnie jednego posiłku (**KAT/4/03 Meal**)
- REG/4/24** Przepis posiłku (**KAT/4/04 MealRecipe**) musi mieć przypisany dokładnie jeden przepis (**KAT/3/01 Recipe**)
- REG/4/25** Przepis posiłku (**KAT/4/04 MealRecipe**) jest przedmiotem kompozycji ze strony posiłku (**KAT/4/03 Meal**)

Reguły dla KAT/4/05 MealProduct

- REG/4/26** Produkt posiłku (**KAT/4/05 MealProduct**) musi być przypisany do dokładnie jednego posiłku (**KAT/4/03 Meal**)
- REG/4/27** Produkt posiłku (**KAT/4/05 MealProduct**) musi mieć przypisany dokładnie jeden produkt (**KAT/2/01 Product**)
- REG/4/28** Produkt posiłku (**KAT/4/05 MealProduct**) nie musi mieć przypisanej żadnej miary domowej (**KAT/2/07 HouseholdMeasure**)
- REG/4/29** Produkt posiłku (**KAT/4/05 MealProduct**) musi mieć przypisaną maksymalnie jedną miarę domową (**KAT/2/07 HouseholdMeasure**)
- REG/4/30** Produkt posiłku (**KAT/4/05 MealProduct**) jest przedmiotem kompozycji ze strony posiłku (**KAT/4/03 Meal**)

Reguły dla KAT/4/06 MealDefinition

- REG/4/31** Definicja posiłku (**KAT/4/06 MealDefinition**) musi być przypisana do dokładnie jednego jadłospisu (**KAT/4/01 MealPlan**)
- REG/4/32** Definicja posiłku (**KAT/4/06 MealDefinition**) musi mieć przypisany dokładnie jeden typ posiłku (**KAT/3/11 MealType**)
- REG/4/33** Definicja posiłku (**KAT/4/06 MealDefinition**) jest przedmiotem kompozycji ze strony jadłospisu (**KAT/4/01 MealPlan**)

Ograniczenia dziedzinowe

Ograniczenia dla KAT/4/01 MealPlan

- OGR/4/01** Atrybut **creationTimestamp** jest wymagany
- OGR/4/02** Atrybut **editTimestamp** jest wymagany
- OGR/4/03** Atrybut **isVisible** jest wymagany
- OGR/4/04** Atrybut **numberOfDays** jest wymagany
- OGR/4/05** Atrybut **numberOfMealsPerDay** jest wymagany
- OGR/4/06** Atrybut **totalDailyEnergy** jest wymagany
- OGR/4/07** Atrybut **percentOfProtein** jest wymagany
- OGR/4/08** Atrybut **percentOfFat** jest wymagany
- OGR/4/09** Atrybut **percentOfCarbohydrates** jest wymagany
- OGR/4/10** Atrybut **creationTimestamp** jest stemplem czasowym
- OGR/4/11** Atrybut **editTimestamp** jest stemplem czasowym
- OGR/4/12** Atrybut **name** jest ciągiem znaków o długości od 1 do 255 znaków
- OGR/4/13** Atrybut **isVisible** jest typu logicznego

- OGR/4/14** Atrybut **numberOfDays** jest liczbą całkowitą nie mniejszą niż 1 i nie większą niż 30
- OGR/4/15** Atrybut **numberOfMealsPerDay** jest liczbą całkowitą nie mniejszą niż 1 i nie większą niż 10
- OGR/4/16** Atrybut **totalDailyEnergy** jest liczbą całkowitą nie mniejszą niż 1
- OGR/4/17** Atrybut **percentOfProtein** jest liczbą całkowitą nie mniejszą niż 0 i nie większą niż 100
- OGR/4/18** Atrybut **percentOfFat** jest liczbą całkowitą nie mniejszą niż 0 i nie większą niż 100
- OGR/4/19** Atrybut **percentOfCarbohydrates** jest liczbą całkowitą nie mniejszą niż 0 i nie większą niż 100
- OGR/4/20** Suma wartości atrybutów **percentOfProtein**, **percentOfFat**, **percentOfCarbohydrates** nie może przekraczać 100

Ograniczenia dla KAT/4/02 MealPlanDay

- OGR/4/21** Atrybut **ordinalNumber** jest wymagany
- OGR/4/22** Atrybut **ordinalNumber** jest liczbą całkowitą nie mniejszą niż 1

Ograniczenia dla KAT/4/03 Meal

- OGR/4/23** Atrybut **ordinalNumber** jest wymagany
- OGR/4/24** Atrybut **ordinalNumber** jest liczbą całkowitą nie mniejszą niż 1

Ograniczenia dla KAT/4/04 MealRecipe

- OGR/4/25** Atrybut **amount** jest wymagany
- OGR/4/26** Atrybut **amount** jest liczbą całkowitą nie mniejszą niż 0

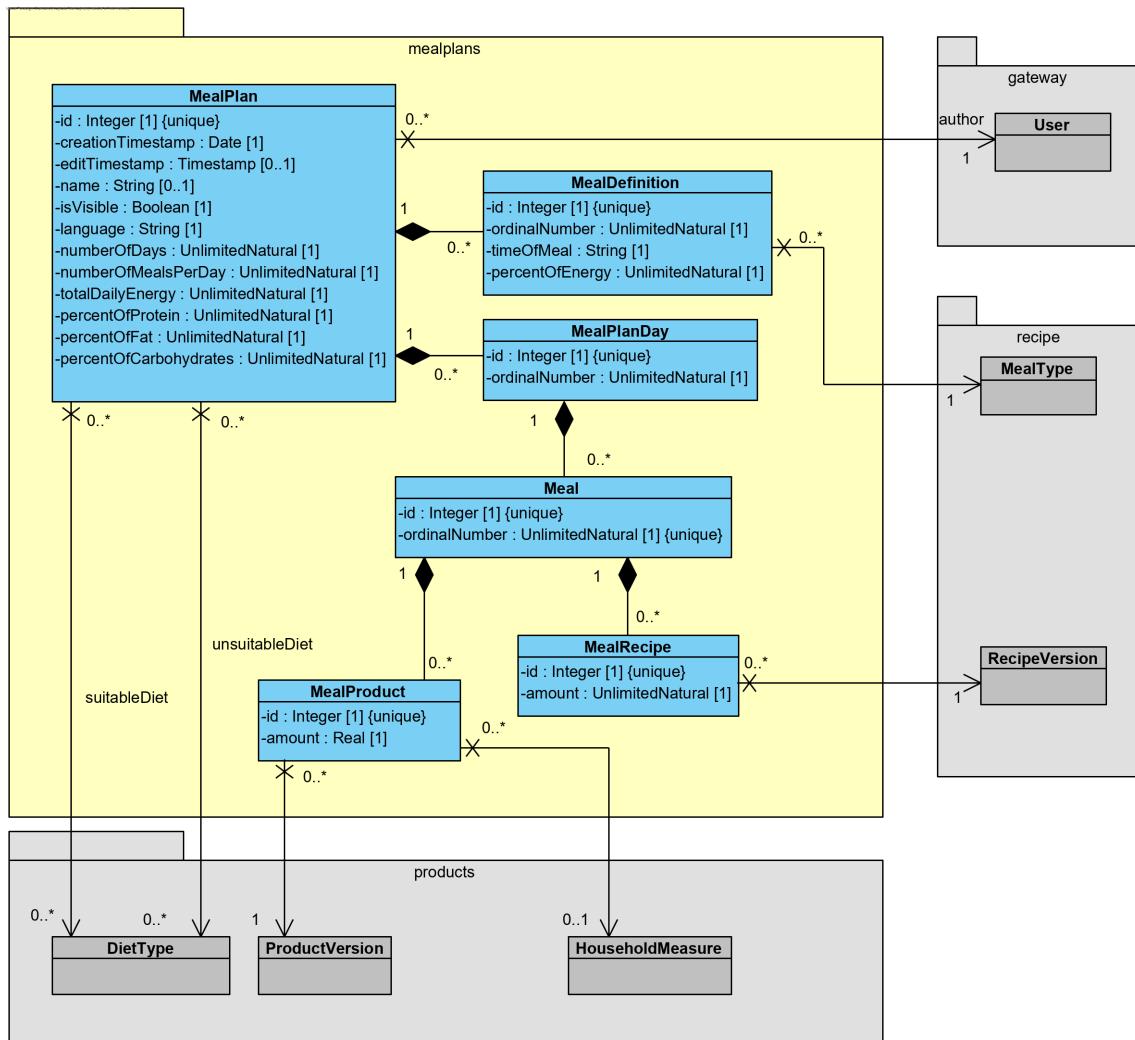
Ograniczenia dla KAT/4/05 MealProduct

- OGR/4/27** Atrybut **amount** jest wymagany
- OGR/4/28** Atrybut **amount** jest liczbą rzeczywistą nie mniejszą niż 0

Ograniczenia dla KAT/4/06 MealDefinition

- OGR/4/29** Atrybut **ordinalNumber** jest wymagany
- OGR/4/30** Atrybut **timeOfMeal** jest wymagany
- OGR/4/31** Atrybut **percentOfEnergy** jest wymagany
- OGR/4/32** Atrybut **ordinalNumber** jest liczbą całkowitą nie mniejszą niż 1
- OGR/4/33** Atrybut **timeOfMeal** jest ciągiem znaków w postaci HH:MI
- OGR/4/34** Atrybut **percentOfEnergy** jest liczbą całkowitą nie mniejszą niż 0 i nie większą niż 100
- OGR/4/35** Suma wartości wszystkich atrybutów **percentOfEnergy** w jednym jadłospisie musi być równa 100

Model informacyjny



Rys. 3.30: Diagram klas - poddziedzina jadłospisy (opr.wł)

3.3.6. Poddziedzina wizyty

Kategorie

KAT/5/01 Appointment (Wizyta)

Opis: Wizyta dietetyczna

Atrybuty:

- | | |
|--|---|
| <ul style="list-style-type: none"> • id • appointmentDate • appointmentState • generalAdvice | <ul style="list-style-type: none"> - identyfikator - data i godzina wizyty - stan wizyty - ogólna porada po wizycie |
|--|---|

KAT/5/02 PatientCard (Karta Pacjenta)

Opis: Karta pacjenta

Atrybuty:

- id - identyfikator
- creationDate - data rejestracji pacjenta do dietetyka

KAT/5/03 BodyMeasurement (Pomiar Ciała)**Opis:** Pomiar ciała**Atrybuty:**

- id - identyfikator
- completionDate - data przeprowadzenia pomiaru
- height - wzrost pacjenta; razem z wagą wykorzystywany do obliczania współczynnika BMI
- weight - waga pacjenta; razem z wzrostem wykorzystywany do obliczania współczynnika BMI
- waist - obwód pasa pacjenta
- percentOfFatTissue - procent tkanki tłuszczowej w ciele pacjenta; norma dla kobiet: 16-20; norma dla mężczyzn: 15-18
- percentOfWater - procent wody w ciele pacjenta; norma dla kobiet: 45-60; norma dla mężczyzn: 50-65
- muscleMass - masa tkanki mięśniowej w ciele pacjenta w kilogramach
- physicalMark - ocena fizyczna; norma: 5
- calciumInBones - poziom wapnia w kościach pacjenta w kilogramach; norma: 2.4kg
- basicMetabolism - podstawowy metabolizm w kilokaloriach
- metabolicAge - wiek metaboliczny w latach
- visceralFatLevel - poziom tłuszcza trzewnego; norma: 1-12

KAT/5/04 NutritionalInterview (Wywiad żywieniowy)**Opis:** Wywiad żywieniowy**Atrybuty:**

- id - identyfikator
- completionDate - czas przeprowadzenia wywiadu
- targetWeight - docelowa waga pacjenta w kilogramach
- advicePurpose - cel wizyty podsumowujący co pacjent pragnie osiągnąć poprzez terapię dietetyczną
- physicalActivity - poziom aktywności fizycznej pacjenta
- diseases - choroby pacjenta
- medicines - leki przyjmowane przez pacjenta
- jobType - typ pracy pacjenta
- likedProducts - produkty spożywcze, które pacjent lubi
- dislikedProducts - produkty spożywcze, których pacjent nie lubi
- foodAllergies - produkty spożywcze na które pacjent jest uczulony
- foodIntolerances - nietolerancje pokarmowe pacjenta

KAT/5/05 CustomNutritionalInterviewQuestion (Niestandardowe Pytanie Wywiadu żywieniowego)**Opis:** Niestandardowe pytanie wywiadu żywieniowego

Atrybuty:

- | | |
|-----------------|----------------------------|
| • id | - identyfikator |
| • ordinalNumber | - numer porządkowy pytania |
| • question | - pytanie |
| • answer | - odpowiedź na pytanie |

KAT/5/06 AssignedMealPlan (Przypisany Jadłospis)**Opis:** Przypisany jadłospis**Atrybuty:**

- | | |
|-----------------|-------------------------------|
| • id | - identyfikator |
| • assigmentTime | - czas przypisania jadłospisu |

Reguły funkcjonowania**Reguły dla KAT/5/01 Appointment**

- REG/5/01** Wizyta (**KAT/5/01 Appointment**) musi być przypisana do dokładnie jednej karty pacjenta (**KAT/5/02 PatientCard**)
- REG/5/02** Wizyta (**KAT/5/01 Appointment**) nie musi mieć przypisanego żadnych pomiarów ciała (**KAT/5/03 BodyMeasurement**)
- REG/5/03** Wizyta (**KAT/5/01 Appointment**) może mieć przypisane maksymalnie jedne pomiary ciała (**KAT/5/03 BodyMeasurement**)
- REG/5/04** Wizyta (**KAT/5/01 Appointment**) nie musi mieć przypisanego żadnego wywiadu żywieniowego (**KAT/5/04 NutritionalInterview**)
- REG/5/05** Wizyta (**KAT/5/01 Appointment**) może mieć przypisany maksymalnie jeden wywiad żywieniowy (**KAT/5/04 NutritionalInterview**)
- REG/5/06** Wizyta (**KAT/5/01 Appointment**) nie musi mieć przypisanego żadnego jadłospisu (**KAT/5/06 AssignedMealPlan**)
- REG/5/07** Wizyta (**KAT/5/01 Appointment**) może mieć przypisanych wiele jadłospisów (**KAT/5/06 AssignedMealPlan**)
- REG/5/08** *Pacjent* może wyświetlać swoją wizytę (**KAT/5/01 Appointment**)
- REG/5/09** *Dietetyk* może dodawać nową wizytę (**KAT/5/01 Appointment**)
- REG/5/10** *Dietetyk* może wyświetlać i edytować swoją wizytę (**KAT/5/01 Appointment**)

Reguły dla KAT/5/02 PatientCard

- REG/5/11** Karta pacjenta (**KAT/5/02 PatientCard**) nie musi mieć przypisanej żadnej wizyty (**KAT/5/01 Appointment**)
- REG/5/12** Karta pacjenta (**KAT/5/02 PatientCard**) może mieć przypisanych wiele wizyt (**KAT/5/01 Appointment**)
- REG/5/13** Karta pacjenta (**KAT/5/02 PatientCard**) musi mieć przypisanego dokładnie jednego pacjenta (**KAT/1/01 User**)
- REG/5/14** Karta pacjenta (**KAT/5/02 PatientCard**) musi mieć przypisanego dokładnie jednego dietetyka (**KAT/1/01 User**)
- REG/5/15** *Pacjent* może wyświetlać swoją kartę pacjenta (**KAT/5/02 PatientCard**)
- REG/5/16** *Dietetyk* może dodawać nową kartę pacjenta (**KAT/5/02 PatientCard**)
- REG/5/17** *Dietetyk* może wyświetlać i edytować karty pacjenta (**KAT/5/02 PatientCard**), którymi zarządza

Reguły dla KAT/5/03 BodyMeasurement

- REG/5/18** Pomiary ciała (**KAT/5/03 BodyMeasurement**) muszą być przypisane do dokładnie jednej wizyty (**KAT/5/01 Appointment**)
- REG/5/19** Pomiary ciała (**KAT/5/03 BodyMeasurement**) są przedmiotem kompozycji ze strony wizyty (**KAT/5/01 Appointment**)

Reguły dla KAT/5/04 NutritionalInterview

- REG/5/20** Wywiad żywieniowy (**KAT/5/04 NutritionalInterview**) musi być przypisany do dokładnie jednej wizyty (**KAT/5/01 Appointment**)
- REG/5/21** Wywiad żywieniowy (**KAT/5/04 NutritionalInterview**) nie musi mieć przypisanego żadnego niestandardowego pytania (**KAT/5/05 CustomNutritionalInterviewQuestion**)
- REG/5/22** Wywiad żywieniowy (**KAT/5/04 NutritionalInterview**) może mieć przypisanych wiele niestandardowych pytań (**KAT/5/05 CustomNutritionalInterviewQuestion**)
- REG/5/23** Wywiad żywieniowy (**KAT/5/04 NutritionalInterview**) nie musi mieć przypisanych żadnych posiadanych sprzętów kuchennych (**KAT/3/07 KitchenAppliance**)
- REG/5/24** Wywiad żywieniowy (**KAT/5/04 NutritionalInterview**) może mieć przypisanych wiele posiadanych sprzętów kuchennych (**KAT/3/07 KitchenAppliance**)
- REG/5/25** Wywiad żywieniowy (**KAT/5/04 NutritionalInterview**) jest przedmiotem kompozycji ze strony wizyty (**KAT/5/01 Appointment**)

Reguły dla KAT/5/05 CustomNutritionalInterviewQuestion

- REG/5/26** Niestandardowe pytanie żywieniowe (**KAT/5/05 CustomNutritionalInterviewQuestion**) musi być przypisane do dokładnie jednego wywiadu żywieniowego (**KAT/5/04 NutritionalInterview**)
- REG/5/27** Niestandardowe pytanie żywieniowe (**KAT/5/05 CustomNutritionalInterviewQuestion**) jest przedmiotem kompozycji ze strony wywiadu żywieniowego (**KAT/5/04 NutritionalInterview**)

Reguły dla KAT/5/06 AssignedMealPlan

- REG/5/28** Przypisany jadłospis (**KAT/5/06 AssignedMealPlan**) musi mieć przydzieloną dokładnie jedną wizytę (**KAT/5/01 Appointment**)
- REG/5/29** Przypisany jadłospis (**KAT/5/06 AssignedMealPlan**) musi mieć przydzielony dokładnie jeden jadłospis (**KAT/4/01 MealPlan**)
- REG/5/30** Przypisany jadłospis (**KAT/5/06 AssignedMealPlan**) jest przedmiotem kompozycji ze strony wizyty (**KAT/5/01 Appointment**)

Ograniczenia dziedzinowe

Ograniczenia dla KAT/5/01 Appointment

- OGR/5/01** Atrybut **appointmentDate** jest wymagany
- OGR/5/02** Atrybut **appointmentState** jest wymagany
- OGR/5/03** Atrybut **appointmentDate** jest stemplem czasowym

- OGR/5/04** Atrybut **appointmentState** jest typu wyliczeniowego i może przyjmować wartości "PLANNED", "CANCELED", "TOOK_PLACE", "COMPLETED"
OGR/5/05 Atrybut **generalAdvice** jest ciągiem znaków

Ograniczenia dla KAT/5/02 PatientCard

- OGR/5/06** Atrybut **creationDate** jest wymagany
OGR/5/07 Atrybut **creationDate** jest stemplem czasowym

Ograniczenia dla KAT/5/03 BodyMeasurement

- OGR/5/08** Atrybut **completionDate** jest wymagany
OGR/5/09 Atrybut **height** jest wymagany
OGR/5/10 Atrybut **weight** jest wymagany
OGR/5/11 Atrybut **waist** jest wymagany
OGR/5/12 Atrybut **completionDate** jest stemplem czasowym
OGR/5/13 Atrybut **height** jest liczbą całkowitą
OGR/5/14 Atrybut **weight** jest liczbą całkowitą
OGR/5/15 Atrybut **waist** jest liczbą rzeczywistą
OGR/5/16 Atrybut **percentOfFatTissue** jest liczbą rzeczywistą nie mniejszą niż 0 i nie większą niż 100
OGR/5/17 Atrybut **percentOfWater** jest liczbą rzeczywistą nie mniejszą niż 0 i nie większą niż 100
OGR/5/18 Atrybut **muscleMass** jest liczbą rzeczywistą
OGR/5/19 Atrybut **physicalMark** jest liczbą rzeczywistą
OGR/5/20 Atrybut **calciumInBones** jest liczbą rzeczywistą
OGR/5/21 Atrybut **basicMetabolism** jest liczbą całkowitą
OGR/5/22 Atrybut **metabolicAge** jest liczbą rzeczywistą
OGR/5/23 Atrybut **visceralFatLevel** jest liczbą rzeczywistą

Ograniczenia dla KAT/5/04 NutritionalInterview

- OGR/5/24** Atrybut **completionDate** jest wymagany
OGR/5/25 Atrybut **targetWeight** jest wymagany
OGR/5/26 Atrybut **advicePurpose** jest wymagany
OGR/5/27 Atrybut **physicalActivity** jest wymagany
OGR/5/28 Atrybut **completionDate** jest stemplem czasowym
OGR/5/29 Atrybut **targetWeight** jest liczbą całkowitą
OGR/5/30 Atrybut **advicePurpose** jest ciągiem znaków
OGR/5/31 Atrybut **physicalActivity** jest typu wyliczeniowego i może przyjmować wartości "EXTREMELY_INACTIVE", "SEDENTARY", "MODERATELY_ACTIVE", "VIGOROUSLY_ACTIVE", "EXTREMELY_ACTIVE"
OGR/5/32 Atrybut **diseases** jest ciągiem znaków
OGR/5/33 Atrybut **medicines** jest ciągiem znaków
OGR/5/34 Atrybut **jobType** jest typu wyliczeniowego i może przyjmować wartości "SITTING", "STANDING", "MIXED"
OGR/5/35 Atrybut **likedProducts** jest ciągiem znaków
OGR/5/36 Atrybut **dislikedProducts** jest ciągiem znaków
OGR/5/37 Atrybut **foodAllergies** jest ciągiem znaków

OGR/5/38 Atrybut **foodIntolerances** jest ciągiem znaków

Ograniczenia dla KAT/5/05 CustomNutritionalInterviewQuestion

OGR/5/39 Atrybut **question** jest wymagany

OGR/5/40 Atrybut **ordinalNumber** jest liczbą całkowitą nie mniejszą niż 1

OGR/5/41 Atrybut **question** jest ciągiem znaków

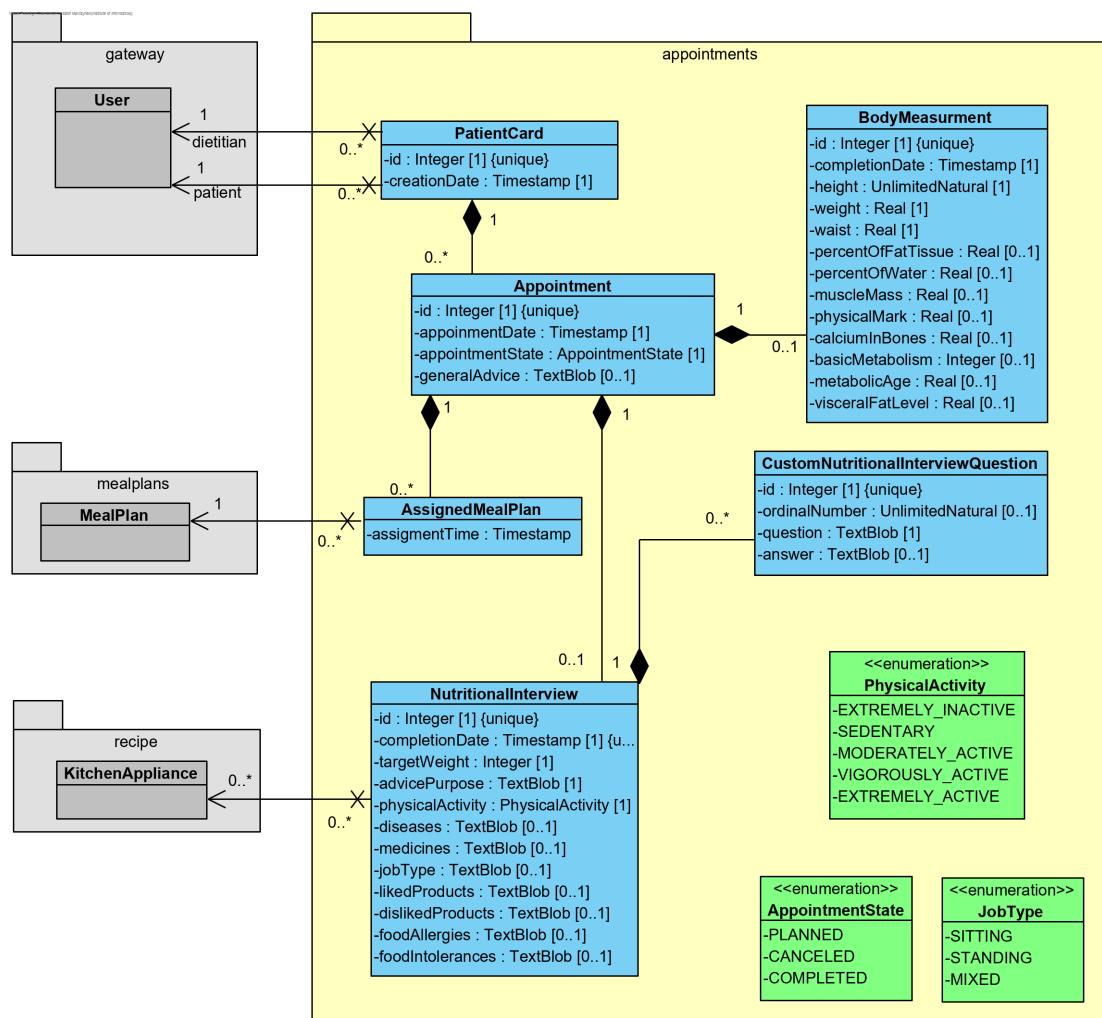
OGR/5/42 Atrybut **answer** jest ciągiem znaków

Ograniczenia dla KAT/5/06 AssignedMealPlan

OGR/5/43 Atrybut **assigmentTime** jest wymagany

OGR/5/44 Atrybut **assigmentTime** jest stemplem czasowym

Model informacyjny



Rys. 3.31: Diagram klas - poddziedzina wizyty (opr.wł)

4. Implementacja

4.1. Wykorzystywane środowiska i narzędzia programistyczne

(zupełnić, opisać, zacytować narzędzia)

Podczas wyboru języków programowania, z użyciem których zostanie zaimplementowany system, postawiono następujące kryteria:

- Ścisła kontrola typów
- Dobre wsparcie dla paradymatu programowania obiektowego
- Niezależność języka od platformy

Wybrane języki spełniające te kryteria to:

- W warstwie backendu Java[20]
- W warstwie frontendu Typescript[19]

Powyższy wybór zaowocował decyzją o zastosowaniu Angulara[9] jako wiodącego frameworka frontendowego i Springa[22] jako wiodącego frameworka backendowego. Frameworki te cieszą się bardzo dużą popularnością, a ich dojrzałość sprawia, że nadają się zarówno przy tworzeniu niewielkich aplikacji jak i systemów klasy enterprise. ([angular](#), [spring](#))

(html, bootstrap, bootswatch)

System został zaprojektowany tak, żeby wykorzystać cechy relacyjnych baz danych, więc podczas wyboru systemu zarządzania bazą danych pod uwagę wzięto tylko relacyjne bazy danych. Rozważano przede wszystkim systemy PostgreSQL[21] i MySQL[27]. Z punktu widzenia funkcjonalności potrzebnych w implementowanej aplikacji oba systemy systemy zarządzania relacyjną bazą danych (ang. Relational Database Management System - RDBMS) wypadają równie dobrze, jednakże ostatecznie wybrano PostgreSQL ze względu na mniej restrykcyjną licencję wykorzystania systemu nawet w rozwiązańach komercyjnych o zamkniętym kodzie. ([liquibase](#))

(Netflix oss, jhipster)

(docker, docker compose, gitlab pipelines)

4.2. Architektura systemu

(opisać stack netflix oss)

(diagram rozmieszczenia)

4.3. Instalacja oprogramowania

4.3.1. Wymagania wstępne

Przed przystąpieniem do wykonywania kolejny kroków należy się upewnić, że następujące narzędzia są zainstalowane:

- Open JDK 11 (<https://adoptopenjdk.net/?variant=openjdk11>)
- Node.js 10 lub nowsza wersja LTS (<https://nodejs.org/en/>)
- Docker 19.03 + Docker Compose 2 (<https://docs.docker.com/install/>)

4.3.2. Instalacja

Aby zbudować i uruchomić projekt na Dockerze należy z poziomu głównego katalogu projektu wykonać polecenia przedstawione na listingu 4.1.

```
1   cd gateway
2   npm install
3   sh gradlew bootJar -Pprod jibDockerBuild
4   cd ../products
5   sh gradlew bootJar -Pprod jibDockerBuild
6   cd ../recipes
7   sh gradlew bootJar -Pprod jibDockerBuild
8   cd ../mealplans
9   sh gradlew bootJar -Pprod jibDockerBuild
10  cd ../appointments
11  sh gradlew bootJar -Pprod jibDockerBuild
12  cd ../docker-compose
13  sh docker-compose up
```

Kod 4.1: Skrypt komplilujący wszystkie mikroserwisy i uruchamiający aplikację na Dockerze (opr. wł.)

Alternatywnie, dla celów deweloperskich można zastosować uproszczony proces nie wykorzystujący Dockera. W tym celu należy najpierw uruchomić JHipster Registry wykonując polecenie z poziomu głównego katalogu projektu wykonać polecenia przedstawione na listingu 4.2.

```
1   cd service-discovery
2   java -jar ./jhipster-registry-5.0.2.jar --spring.profiles.active=dev
      --spring.security.user.password=admin
      --spring.cloud.config.server.composite.0.type=git
      --spring.cloud.config.server.composite.0.uri=
      https://github.com/jhipster/jhipster-registry-sample-config
```

Kod 4.2: Uruchamianie JHipster Registry (opr. wł.)

Następnie z poziomu katalogu każdego z serwisów (gateway, products, recipes, mealplans, appointments) należy wykonać polecenie uruchamiające Gradle Wrapper przedstawione na listingu 4.3.

```
1 ./gradlew
```

Kod 4.3: Uruchamianie Gradle Wrapper (opr. wł.)

Po uruchomieniu wszystkich serwisów aplikacja będzie dostępna pod adresem *localhost:8080*.

4.4. Prezentacja aplikacji

(podstawowy opis poruszania się po aplikacji, zrzuty ekranu z kilku najważniejszych widoków)

4.5. Dokumentacja kodu

(javadoc, swagger, jdl)

4.6. Testy

(przykładowe testy jednostkowe i integracyjne)

(testy użyteczności)

Zakończenie

4.7. Podsumowanie pracy

(W pracy udało mi się dużo zrobić. Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam congue neque id dolor.)

4.8. Możliwości dalszego rozwoju

(Mnóstwo innych rzeczy da się poprawić i rozwinąć. Donec et nisl id sapien blandit mattis. Aenean dictum odio sit amet risus. Morbi purus. Nulla a est sit amet purus venenatis iaculis. Vivamus viverra purus vel magna. Donec in justo sed odio malesuada dapibus. Nunc ultrices aliquam nunc. Vivamus facilisis pellentesque velit. Nulla nunc velit, vulputate dapibus, vulputate id, mattis ac, justo. Nam mattis elit dapibus purus. Quisque enim risus, congue non, elementum ut, mattis quis, sem. Quisque elit.)

Bibliografia

- [1] Anmarsoft, *Aliant Kalkulator dietetyczny*, <https://aliant.com.pl>. Dostęp 13.09.2019.
- [2] AURA GROUP, *Dietetyk Pro*, <https://program.dietetykpro.pl/>. Dostęp 13.09.2019.
- [3] Ciborowska, H., Rudnicka, A., *Dietetyka. Żywienie zdrowego i chorego człowieka*, 4 wyd. (PZWL, Warszawa, 2019).
- [4] Evans, E., *Domain-Driven Design. Zapanuj nad złożonym systemem informatycznym* (Helion, Gliwice, 2015).
- [5] Food and Agriculture Organization of the United Nations, *Energy requirements of adults*, <http://www.fao.org/3/y5686e/y5686e07.htm>. Dostęp 28.10.2019.
- [6] Fowler, M., *MonolithFirst*, <https://martinfowler.com/bliki/MonolithFirst.html>. Dostęp 27.10.2019.
- [7] Fowler, M., *Temporal Object Pattern*, <https://martinfowler.com/eaaDev/TemporalObject.html>. Dostęp 10.09.2019.
- [8] Głabska, D., Kozłowska, L., Lange, E., Włodarek, D., *Dietoterapia* (PZWL, Warszawa, 2017).
- [9] Google, *Angular 7.2.4*, <https://angular.io>. Dostęp 10.05.2019.
- [10] Google, *Google Trends*, <https://trends.google.pl>. Dostęp 10.09.2019.
- [11] Google, *YouTube*, <https://youtube.com>. Dostęp 10.09.2019.
- [12] HERMAX sp. z o.o., *Kcalmar PRO*, <https://kcalmar.com/>. Dostęp 13.09.2019.
- [13] Instytut Żywności i Żywienia, *Regulamin dostępu do bazy "Wartość odżywcza produktów spożywczych i potraw"*, <http://www.izz.waw.pl/warto-odywcza-produktow-spojwczych-i-potraw/regulamin-dostpu-do-bazy>. Dostęp 28.10.2019.
- [14] Jarosz, M., Bułhak-Jachymczyk, B., *Normy żywienia człowieka. Podstawy prewencji otyłości i chorób niezakaźnych* (PZWL, Warszawa, 2008).
- [15] Kaczorowski, B., Baturo, W., Smochowska, Z., Działoszyński, B., Kamiński, H., Milusińska, A., Rossa, A., Uba, M., Wilińska, M., Ziemińska, M., *Encyklopedia Powszechna PWN*, tom 7 (Wydawnictwo Naukowe PWN, Warszawa, 2009).
- [16] Kunachowicz, H., Przygoda, B., Nadolna, I., Iwanow, K., *Tabele składu i wartości odżywczej żywności* (PZWL, Warszawa, 2017).
- [17] Mazur, H., Mazur, Z., *Projektowanie relacyjnych baz danych* (Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 2004).
- [18] Microsoft, *Common web application architectures*, <https://docs.microsoft.com/en-us/dotnet/architecture/modern-web-apps-azure/common-web-application-architectures>. Dostęp 27.10.2019.
- [19] Microsoft, *TypeScript 3.4*, <https://www.typescriptlang.org/>. Dostęp 10.05.2019.
- [20] Oracle Corporation, *Java 8 SE*, <https://docs.oracle.com/javase/8/docs/>. Dostęp 10.05.2019.
- [21] Oracle Corporation, *MySQL 8.0*, <https://www.mysql.com/>. Dostęp 27.10.2019.
- [22] Pivotal Software, *Spring Framework 5*, <https://spring.io/>. Dostęp 10.05.2019.
- [23] Redakcja dietetycy.org.pl, *Programy dla dietetyków 2018 – porównanie*, <https://dietetycy.org.pl/programy-dla-dietetykow-2018-porownanie/>. Dostęp 28.10.2019.
- [24] Richardson, C., *Microservices Patterns* (Packt Publishing, Birmingham, 2018).
- [25] Sharma, S., *Mastering Microservices with Java 9*, 2 wyd. (Packt Publishing, Birmingham, 2017).
- [26] Spotbeans sp. z o.o., *TiqDiet*, <https://tiqdiet.com/pl/>. Dostęp 13.09.2019.

- [27] The PostgreSQL Global Development Group, *PostgreSQL 11.5*, <https://www.postgresql.org/>. Dostęp 10.09.2019.
- [28] Thomasworks, *Vitme*, <https://vitme.pl/>. Dostęp 13.09.2019.
- [29] de la Torre, C., Wagner, B., Rousos, M., *.NET Microservices: Architecture for Containerized .NET Applications* (Microsoft, Redmond, 2019).
- [30] USDA, *USDA National Nutrient Database for Standard Reference, Legacy*, <http://www.ars.usda.gov/Services/docs.htm?docid=8964>. Dostęp 28.10.2019.
- [31] Vernon, V., *DDD. Kompendium Wiedzy* (Helion, Gliwice, 2018).
- [32] Walczak, M., Krasowska-Walczak, G., *Dietetyk na rynku usług medycznych w Polsce i wybranych krajach*, Zdrowie Publiczne i Zarządzanie. 2015, tom 13 (2): 204–215.
- [33] XLTEAM, *Dietico*, <https://dietico.pl/>. Dostęp 13.09.2019.

Spis rysunków

1.	Zainteresowanie hasłem "dietetyk" w ujęciu czasowym [10]	1
2.1.	Diagram przypadków użycia - użytkownicy (opr.wł)	13
2.2.	Diagram przypadków użycia - poddziedzina administracyjna (opr.wł)	15
2.3.	Diagram przypadków użycia - poddziedzina produkty (opr.wł)	17
2.4.	Diagram przypadków użycia - poddziedzina przepisy (opr.wł)	19
2.5.	Diagram przypadków użycia - poddziedzina jadłospisy (opr.wł)	21
2.6.	Diagram przypadków użycia - poddziedzina wizyty (opr.wł)	23
3.1.	Prototyp interfejsu - strona startowa (opr.wł)	25
3.2.	Prototyp interfejsu - układ strony na urządzeniu mobilnym (opr.wł)	26
3.3.	Prototyp interfejsu - widok administratora (opr.wł)	27
3.4.	Prototyp interfejsu - rejestrowanie do systemu (opr.wł)	28
3.5.	Prototyp interfejsu - logowanie do systemu (opr.wł)	28
3.6.	Prototyp interfejsu - zarządzanie użytkownikami (opr.wł)	29
3.7.	Prototyp interfejsu - lista produktów (opr.wł)	30
3.8.	Prototyp interfejsu - dodawanie nowego lub edycja istniejącego produktu (opr.wł)	30
3.9.	Prototyp interfejsu - szczegóły produktu (opr.wł)	31
3.10.	Prototyp interfejsu - lista przepisów (opr.wł)	31
3.11.	Prototyp interfejsu - dodawanie nowego lub edycja istniejącego przepisu (opr.wł)	32
3.12.	Prototyp interfejsu - szczegóły przepisu (opr.wł)	32
3.13.	Prototyp interfejsu - lista jadłospisów (opr.wł)	33
3.14.	Prototyp interfejsu - dodawanie nowego lub edycja istniejącego jadłospisu - zakładka ustawień (opr.wł)	33
3.15.	Prototyp interfejsu - dodawanie nowego lub edycja istniejącego jadłospisu - zakładka kalendarza (opr.wł)	34
3.16.	Prototyp interfejsu - lista wizyt (opr.wł)	34
3.17.	Prototyp interfejsu - dodawanie nowej karty pacjenta (opr.wł)	35
3.18.	Prototyp interfejsu - szczegóły karty pacjenta (opr.wł)	35
3.19.	Prototyp interfejsu - dodawanie nowej wizyty (opr.wł)	36
3.20.	Prototyp interfejsu - szczegóły wizyty (opr.wł)	36
3.21.	Prototyp interfejsu - wizyta - wywiad żywieniowy - krok 1 (opr.wł)	37
3.22.	Prototyp interfejsu - wizyta - wywiad żywieniowy - krok 2 (opr.wł)	37
3.23.	Prototyp interfejsu - wizyta - wywiad żywieniowy - krok 3 (opr.wł)	38
3.24.	Prototyp interfejsu - wizyta - pomiary ciała (opr.wł)	38
3.25.	Podstawowa architektura systemu (opr.wł)	39
3.26.	Diagram klas - typy danych (opr.wł)	41
3.27.	Diagram klas - poddziedzina administracyjna (opr.wł)	44
3.28.	Diagram klas - poddziedzina produkty (opr.wł)	51

3.29. Diagram klas - poddziedzina przepisy (opr.wł)	59
3.30. Diagram klas - poddziedzina jadłospisy (opr.wł)	64
3.31. Diagram klas - poddziedzina wizyty (opr.wł)	69

Spis tabel

1.1.	Rozwiązania konkurencyjne - cechy funkcjonalne (opr.wł)	5
1.2.	Rozwiązania konkurencyjne - cechy niefunkcjonalne (opr.wł)	6
2.1.	Sformułowanie problemu (opr.wł)	11
2.2.	Pozycjonowanie produktu (opr.wł)	12
2.3.	Użytkownicy (opr.wł)	13
2.4.	Wymagania funkcjonalne - poddziedzina administracyjna (opr.wł)	14
2.5.	Wymagania funkcjonalne - poddziedzina produkty (opr.wł)	16
2.6.	Wymagania funkcjonalne - poddziedzina przepisy (opr.wł)	18
2.7.	Wymagania funkcjonalne - poddziedzina jadłospisy (opr.wł)	20
2.8.	Wymagania funkcjonalne - poddziedzina wizyty (opr.wł)	22

Spis kodów źródłowych

4.1	Skrypt komplilujący wszystkie mikroserwisy i uruchamiający aplikację na Dockerze (opr. wł.)	71
4.2	Uruchamianie JHipster Registry (opr. wł.)	71
4.3	Uruchamianie Gradle Wrapper (opr. wł.)	72
A.1	Definicja mikroserwisów w języku JDL (opr. wł.)	84

A. JDL

```
1 // <<START gateway.jh>>
2 application {
3     config {
4         baseName gateway
5         packageName pl.marczynski.dietify.gateway
6
7         applicationType gateway
8         serverPort 8080
9
10        authenticationType jwt
11        buildTool gradle
12        serviceDiscoveryType eureka
13
14        databaseType sql
15        devDatabaseType h2Disk
16        prodDatabaseType postgresql
17        cacheProvider hazelcast
18        enableHibernateCache true
19
20        enableTranslation true
21        nativeLanguage en
22        languages [en, pl]
23
24        useSass true
25        clientFramework angularX
26        //clientTheme flatly
27        //clientThemeVariant primary
28    }
29    entities *
30 }
31
32 deployment {
33     deploymentType docker-compose
34
35     appsFolders [gateway, products, recipes, mealplans, appointments]
36     dockerRepositoryName "dietify"
37     monitoring elk
38 }
39
40 service * with serviceImpl
41 noFluentMethod *
```

```

42 // <<END gateway.jh>>
43
44 // <<START products.jh>>
45 application {
46     config {
47         baseName products
48         packageName pl.marczynski.dietify.products
49
50         applicationType microservice
51         serverPort 8081
52
53         authenticationType jwt
54         buildTool gradle
55         serviceDiscoveryType eureka
56
57         databaseType sql
58         devDatabaseType h2Disk
59         prodDatabaseType postgresql
60         cacheProvider hazelcast
61         enableHibernateCache true
62
63         enableTranslation true
64         nativeLanguage en
65         languages [en, pl]
66     }
67     entities Product, ProductSubcategory, ProductCategory,
68     ↳ ProductCategoryTranslation, NutritionData, NutritionDefinition,
69     ↳ NutritionDefinitionTranslation, HouseholdMeasure, DietType,
70     ↳ DietTypeTranslation, ProductBasicNutritionData
71 }
72 // <<END products.jh>>
73
74 // <<START recipes.jh>>
75 application {
76     config {
77         baseName recipes
78         packageName pl.marczynski.dietify.recipes
79
80         applicationType microservice
81         serverPort 8082
82
83         authenticationType jwt
84         buildTool gradle
85         serviceDiscoveryType eureka
86
87         databaseType sql
88         devDatabaseType h2Disk
89         prodDatabaseType postgresql
         cacheProvider hazelcast
         enableHibernateCache true

```

```

90         enableTranslation true
91         nativeLanguage en
92         languages [en, pl]
93
94         testFrameworks [gatling, cucumber]
95     }
96     entities Recipe, MealType, MealTypeTranslation, DishType,
97     ↳ DishTypeTranslation, KitchenAppliance, KitchenApplianceTranslation,
98     ↳ RecipeSuitableForDiet, RecipeUnsuitableForDiet, RecipeSection,
99     ↳ ProductPortion, PreparationStep, RecipeBasicNutritionData
100    }
101   // <<END recipes.jh>>
102
103  // <<START mealPlans.jh>>
104  application {
105    config {
106      baseUrl mealplans
107      packageName pl.marczynski.dietify.mealplans
108
109      applicationType microservice
110      serverPort 8083
111
112      authenticationType jwt
113      buildTool gradle
114      serviceDiscoveryType eureka
115
116      databaseType sql
117      devDatabaseType h2Disk
118      prodDatabaseType postgresql
119      cacheProvider hazelcast
120      enableHibernateCache true
121
122      enableTranslation true
123      nativeLanguage en
124      languages [en, pl]
125    }
126    entities MealPlan, MealDefinition, MealPlanSuitableForDiet,
127    ↳ MealPlanUnsuitableForDiet, MealPlanDay, Meal, MealRecipe, MealProduct
128  }
129 // <<END mealPlans.jh>>
130
131 // <<START appointments.jh>>
132 application {
133   config {
134     baseUrl appointments
135     packageName pl.marczynski.dietify.appointments
136
137     applicationType microservice
138     serverPort 8084
139
140     authenticationType jwt

```

```
137     buildTool gradle
138     serviceDiscoveryType eureka
139
140     databaseType sql
141     devDatabaseType h2Disk
142     prodDatabaseType postgresql
143     cacheProvider hazelcast
144     enableHibernateCache true
145
146     enableTranslation true
147     nativeLanguage en
148     languages [en, pl]
149 }
150 entities Appointment, BodyMeasurement, PatientCard, NutritionalInterview,
151     ↳ AssignedMealPlan, OwnedKitchenAppliance,
152     ↳ CustomNutritionalInterviewQuestion, AppointmentEvaluation
153 }
154 // <<END appointments.jh>>
```

Kod A.1: Definicja mikroserwisów w języku JDL (opr. wł.)