
FueliciousFit

**“FueliciousFit”
Dokumentacja projektu
Przedmiot: MAS**

Wersja v0.0001

FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

Historia Zmian

Data	Wersja	Opis zmiany	Autor
30.03.2023	0.0001	MAS dokumentacja	Robert Makłowicz

FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

Spis treści

Spis treści

1.	Wprowadzenie	4
1.1	Definicje, akronimy i skróty	4
1.2	Organizacja dokumentu	4
2.	Opis modelu analitycznego	4
2.1	Opis ogólny (historyjka)	4
2.2	Diagram klas analityczny	6
2.3	Diagram przypadków użycia - wnioski	7
2.4	Diagram przypadków użycia	8
2.5	Scenariusz przypadków użycia	9
2.5.1	[CT-1] Stwórz plan treningowy lub dietetyczny lub przepis	9
2.5.2	Aktorzy	9
2.5.3	Cel i kontekst	9
2.5.4	Zależności	9
2.5.5	Założenia i warunki początkowe	9
2.5.6	Inicjujące zdarzenie biznesowe	9
2.5.7	Przebieg podstawowy	9
2.5.8	Przebiegi alternatywne	9
2.5.9	Punkty rozszerzające	9
2.5.10	Warunki końcowe	9
3.	Analiza dynamiczna	10
3.1	Diagram aktywności dla przypadku użycia	10
3.2	Diagram interakcji (sekwencji) dla przypadku użycia	11
3.3	Diagram stanu dla klasy	12
3.4	Diagram klas projektowy	13
3.5	Analiza dynamiczna skutki	14
3.6	Decyzje projektowe	14
4.	Projekt GUI	16
4.1	Opis projektu GUI	16
4.1.1	Widżet tworzenia planu dietetycznego, treningowego lub przepisu	16
4.1.2	Formularz do dodania danych planu dietetycznego, treningowego lub przepisu	17
4.1.3	Formularz do dodania dania lub ćwiczenia do planu dietetycznego lub treningowego	18
4.1.4	Formularz tworzenia do dania lub ćwiczenia	19

FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

Dokument Projektu na przedmiot MAS

1. Wprowadzenie

Niniejsza dokumentacja projektu ma na celu przedstawienie propozycji aplikacji żywieniowo-fitnessowej, która ma ułatwić implementację oraz śledzenie planu żywieniowego i fitnessowego użytkowników. Aplikacja ta pomoże użytkownikom utrzymać zdrową dietę, śledzić spożywane posiłki oraz monitorować bilans kaloryczny.

Dokumentacja zawiera diagramy analityczne i projektowe aktywności, które przedstawiają różne etapy procesu implementacji aplikacji. Zaprezentowane zostaną również główne funkcjonalności i wymagania systemu.

1.1 Definicje, akronimy i skróty

- Administrator – Pracownik TirRex Logistics mający utworzone specjalne konto z pełnymi uprawnieniami,
- Gość- osoba niezalogowana, korzystająca z systemu,
- Infrastruktura – fizyczne maszyny i oprogramowanie będące własnością TirRex Logistics,
- Użytkownik – osoba zalogowana do systemu z danymi prawami, dzieli się na logistyka, księgowego, konsultanta, mechanika,
- System – aktor odpowiedzialny za wycenę zamówień i przesłanie jej do klienta,
- Zamówienie – jest to usługa przewozu,
- Trasa – dostępne trasy dla kierowcy na dostawę towaru,
- Kopia zapasowa – są to dane do otworzenia bazy danych w przypadku ich utraty lub uszkodzenia.

1.2 Organizacja dokumentu

- W pierwszym punkcie dokumentu znajduje się wprowadzenie, definicje i organizacja dokumentu,
- W drugim punkcie dokumentu znajdują się opis modelu analitycznego,
- W trzecim punkcie dokumentu znajdują się analiza dynamiczna systemu,
- W czwartym punkcie dokumentu znajdują się prezentacja projektu GUI,

2. Opis modelu analitycznego

2.1 Opis ogólny (historyjka)

Firma żywieniowo-fitnessowi określiła swoje wstępne wymagania odnoszące się do systemu informatycznego, który pomógłby w usprawnieniu treningu, dobrych nawyków żywieniowych lub diety.

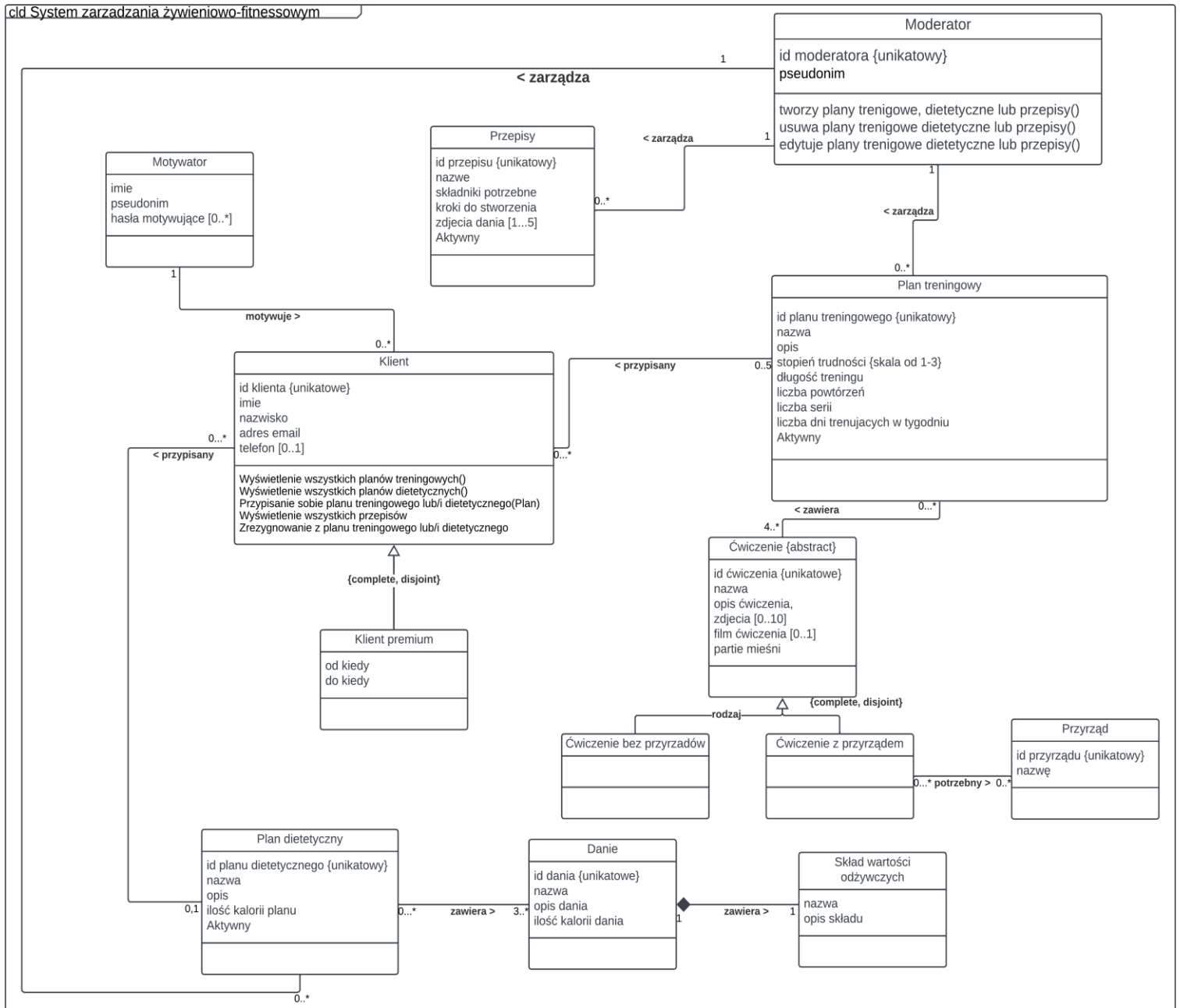
1. W systemie powinny być przechowywane dane kontaktowe klientów (unikatowy indentyfikator użytkownika, Imię, Nazwisko, adres email, telefon (opcjonalny) później lista się zwiększy). Dodatkowy należy zapamiętać dane klientów premium który ma atrybuty od kiedy do kiedy.
2. W systemie powinny być przechowywane dane klientów. Każdy klient może mieć max 5 planów treningowych przypisanych do niego w danym momencie. Dla planu treningowego należy pamiętać: unikatowy indentyfikator planu, opis, zestaw ćwiczeń, stopień trudności (w skali 1-3, przy czym 1 oznacza najłatwiejsza), długość treningu, liczba powtórzeń ćwiczeń, liczba serii ćwiczeń, liczba dni trenujących w tygodniu.

FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

3. Plan treningowy zawiera minimum 4 ćwiczenia. W ćwiczeniach należy pamiętać: unikatowy indefikator ćwiczenia, opis ćwiczenia, zdjęcia (max zdjęć to na razie 10), opcjonalne filmik pokazujący jak wykonać, na jakie partie mięśni wpływa.
4. Ćwiczenia dzielimy ze względu na rodzaj: ćwiczenia bez przyrządów, ćwiczenia z przyrządami należy zapamiętać potrzebny przyrząd (maszynę) do ćwiczeń. Przyrząd do ćwiczeń powinien mieć swój unikatowy indefikator, nazwę.
5. Każde ćwiczenie z przyrządem (maszyną) może posiadać dowolną ilość przyrządów.
6. W systemie powinny być przechowywane także dane o planach dietetycznych: unikatowy indefikator planu dietetycznego, nazwa, opis, ilość kalorii planu, zestaw dan.
7. Plan dietetyczny zawiera minimum 3 dania. W daniach należy pamiętać: unikatowy indefikator dania, opis dania, ilość kalorii dania. Także każde danie zawiera swój własny skład wartości odżywczych, który istnieje tylko do danego dania.
8. Skład wartości odżywczych zawiera nazwa, opis składu
9. Każdy klient może mieć przypisany max jeden plan dietetyczny w danym momencie.
10. W systemie przechowywane będą także przepisy na zdrowe dania. Dla przepisów należy zapamiętać unikatowy indefikator przepisu, nazwę, składniki potrzebne do stworzenia, liste kroków, zdjęcia gotowego dania max 5 zdjęć.
11. Dla każdego klienta przydzielany jest jeden Motywator, który w tym momencie będzie tylko motywował klientów do ćwiczeń i zdrowego trybu życia (później będzie odpowiedzialny za więcej rzeczy zamieni się w AI coach). Motywator posiada swoje imię, pseudonim, lista haseł motywujących.
12. W systemie jest przechowane kompendium wiedzy na temat żywienia i treningu.
13. W systemie znajduje się Moderator, który tworzy, usuwać, edytować plany treningowe i dietetyczne. Moderator powinien mieć unikatowy indefikator, pseudonim. Moderator może zarządzać wieloma planami dietetycznymi jak i także treningowymi.
14. Moderator może zarządzać wieloma przepisami.
15. Moderator wstępnie ma możliwość:
 - tworzyć plany treningowe, dietetyczne lub przepisy
 - usuwać plany treningowe, dietetyczne lub przepisy
 - edytować plany treningowe, dietetyczne lub przepisy
16. System powinien umożliwiać klientowi realizację następujących zadań:
 - Wyświetlenie wszystkich planów treningowych,
 - Wyświetlenie wszystkich planów dietetycznych,
 - Przypisanie sobie planu treningowego lub/i dietetycznego,
 - Wyświetlenie wszystkich przepisów
 - Zrezygnowanie z planu treningowego lub/i dietetycznego

FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

2.2 Diagram klas analityczny



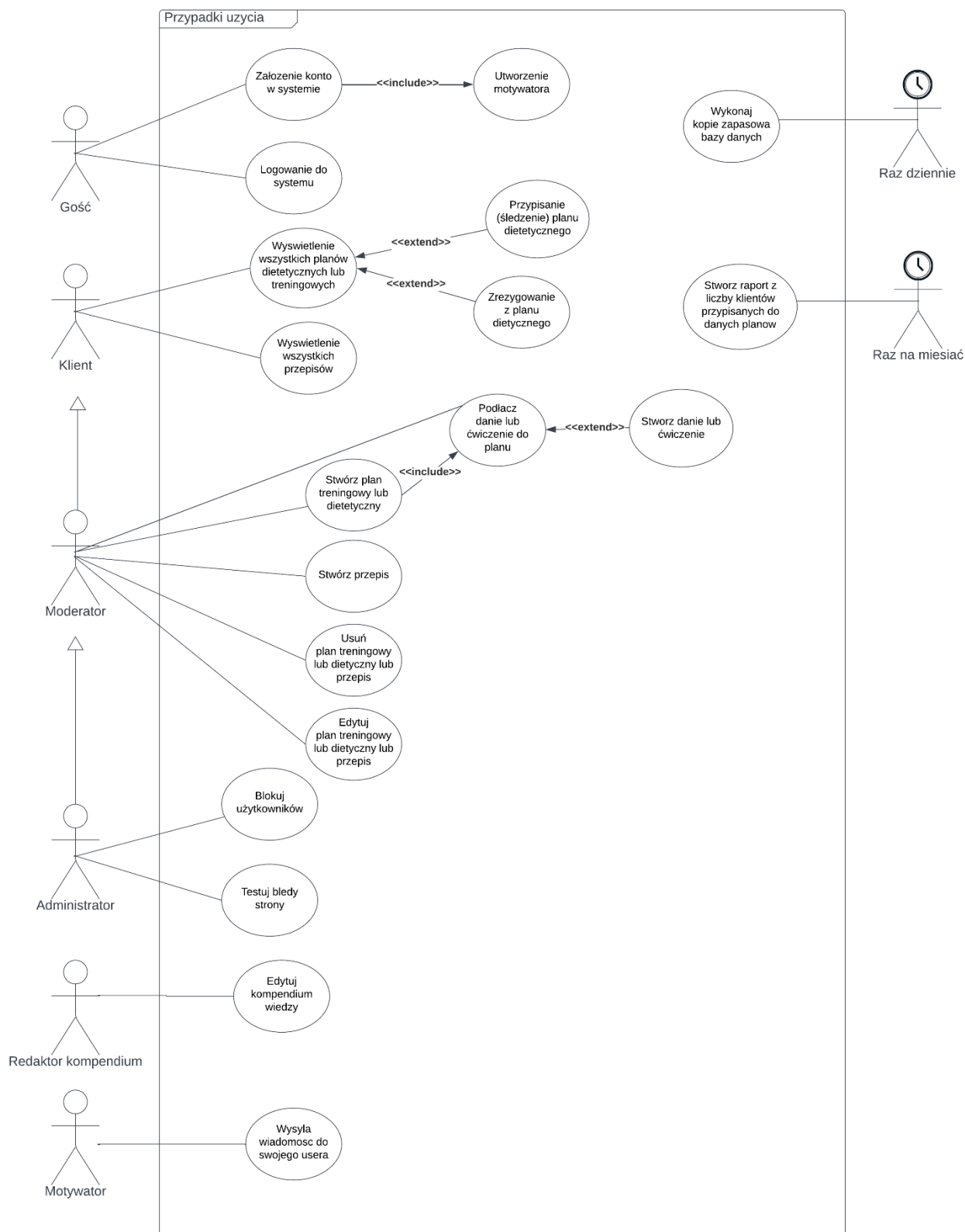
FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

2.3 Diagram przypadków użycia - wnioski

Przedstawiony poniżej diagram przypadków użycia prezentuje opis podanego systemu. W diagramie są zawarte najważniejsze funkcjonalności systemu dla użytkowników oraz wyróżnieni zostali najważniejsi aktorzy systemu. Aktor Gość jest to osoba niezalogowana, która może utworzyć konto w systemie i także się zalogować do niego. Gość po zalogowaniu staje się użytkownikiem czyli osoba zalogowana o danych uprawnieniach i rolach. Użytkownika możemy podzielić na danych aktorów: Klient, Moderator, Administrator, Redaktor kompendium i Motywatora. Aktor klient posiada funkcjonalność do wyświetlania wszystkich planów i przepisów i także śledzenie ich lub anulowanie śledzenia. Administrator wraz z Moderatorem zarządzają planami i przepisy w systemie (dodawanie, edytowanie i usuwanie). Redaktor kompendium edytuje kompendium wiedzy. Motywator wysyła motywacyjne wiadomości do użytkowników (klientów). Administrator posiada także funkcjonalność blokowania użytkowników i testowanie błędów na stronie. W systemie są także dwa aktorzy czasowi: raz dziennie i raz na miesiąc. Aktor Raz dziennie jest odpowiedzialny za wykonywanie kopii zapasowej bazy danych. Drugi aktor czasowy raz na miesiąc tworzy raport z liczby klientów przypisany do danych planów treningowych lub żywieniowych.

FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

2.4 Diagram przypadków użycia



FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

2.5 Scenariusz przypadków użycia

2.5.1 [CT-1] *Stwórz plan treningowy lub dietetyczny lub przepis*

2.5.2 *Aktorzy*

1. Moderator, *Stwórz plan treningowy lub dietetyczny*

2.5.3 *Cel i kontekst*

Osoba tworzy nowy plan treningowy lub dietetyczny

2.5.4 *Zależności*

2.5.4.1 Wymagane przypadki użycia

Podłącz danie lub ćwiczenie do planu

2.5.4.2 Opcjonalne przypadki użycia

Brak.

2.5.5 Założenia i warunki początkowe

1. Czy osoba jest uprawniona do utworzenia nowego planu dietetycznego lub treningowego,

2.5.6 Inicjujące zdarzenie biznesowe

Brak.

2.5.7 Przebieg podstawowy

1. Aktor uruchamia przypadek użycia „Stwórz plan treningowy lub dietetyczny” wybierając odpowiedni przycisk:
 - a) Stwórz plan treningowy,
 - b) Stwórz plan dietetyczny,
2. System wyświetla formularz i prosi o wypełnienie informacji
3. Aktor uzupełnia dane w formularzu i zapisuje
4. System zapisuje dane o nowym planie lub przepisie w bazie
5. System wyświetla okno do podłączania dania lub ćwiczenia do planu
6. Aktor uzupełnia dane w formularzu i zapisuje.
7. System zapisuje stan w bazie danych

2.5.8 Przebiegi alternatywne

2.5.8.1 System informuje o błędnym lub braku wypełnienia jakiś danych w formularzu i prosi o podjęcie działania

- 4a. Aktor kończy przypadek użycia
- 4b. Aktor chce poprawić dane systemu przechodzi do pkt 2

2.5.8.2 System informuje o błędnym lub braku wybrania dania lub ćwiczenia do planu

- 7a. Aktor kończy przypadek użycia
- 7b. Aktor chce poprawić dane systemu przechodzi do pkt 5

2.5.9 Punkty rozszerzające

Brak.

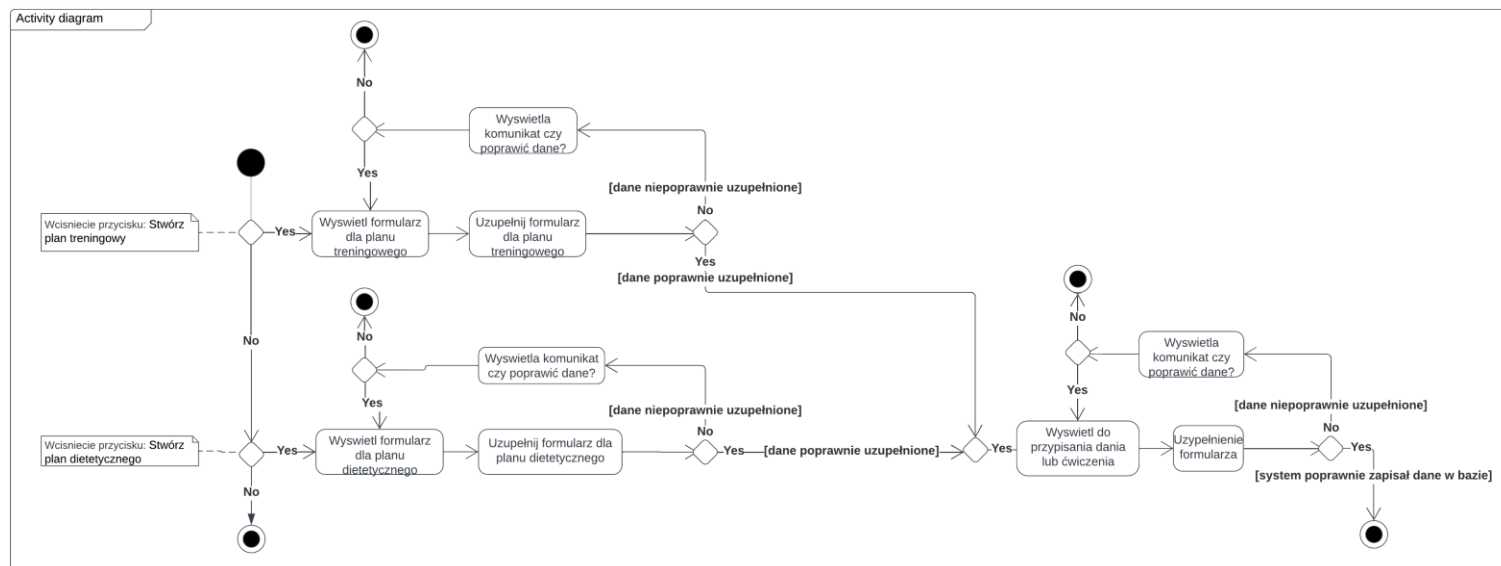
2.5.10 *Warunki końcowe*

1. System tworzy nowy plan w bazie oraz z sukcesem podłącza do niego danie lub ćwiczenie

FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

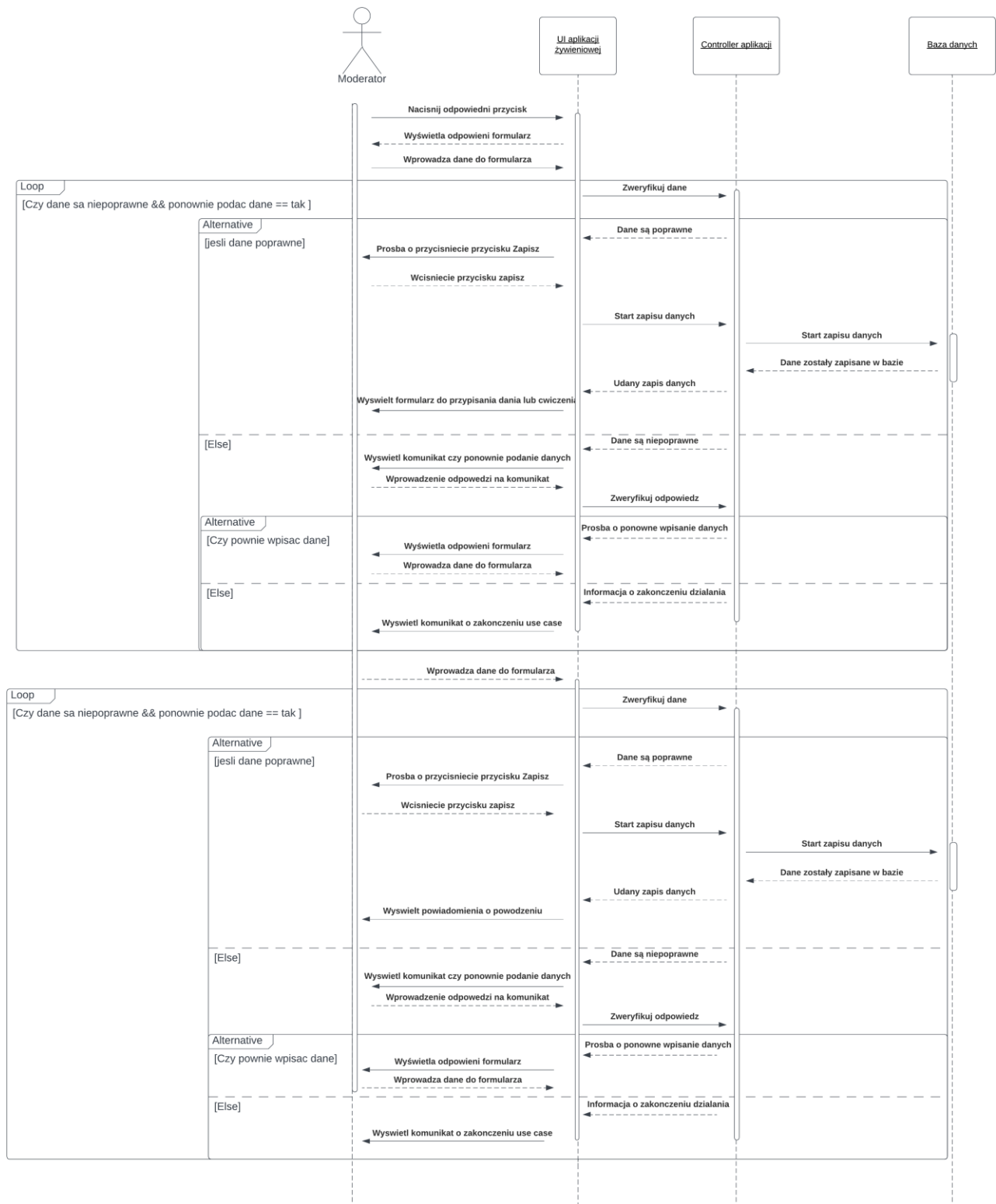
3. Analiza dynamiczna

3.1 Diagram aktywności dla przypadku użycia



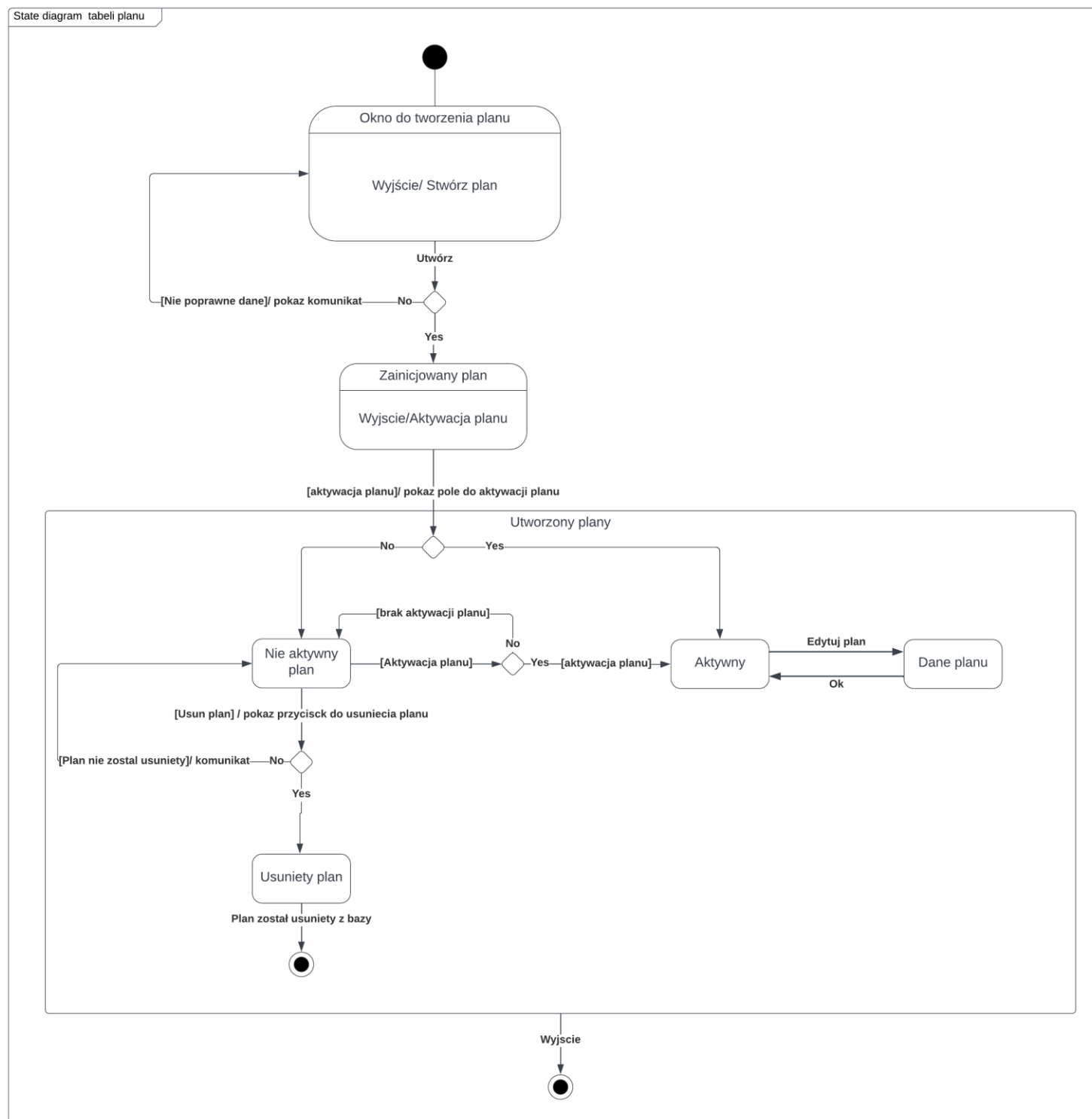
FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

3.2 Diagram interakcji (sekwencji) dla przypadku użycia



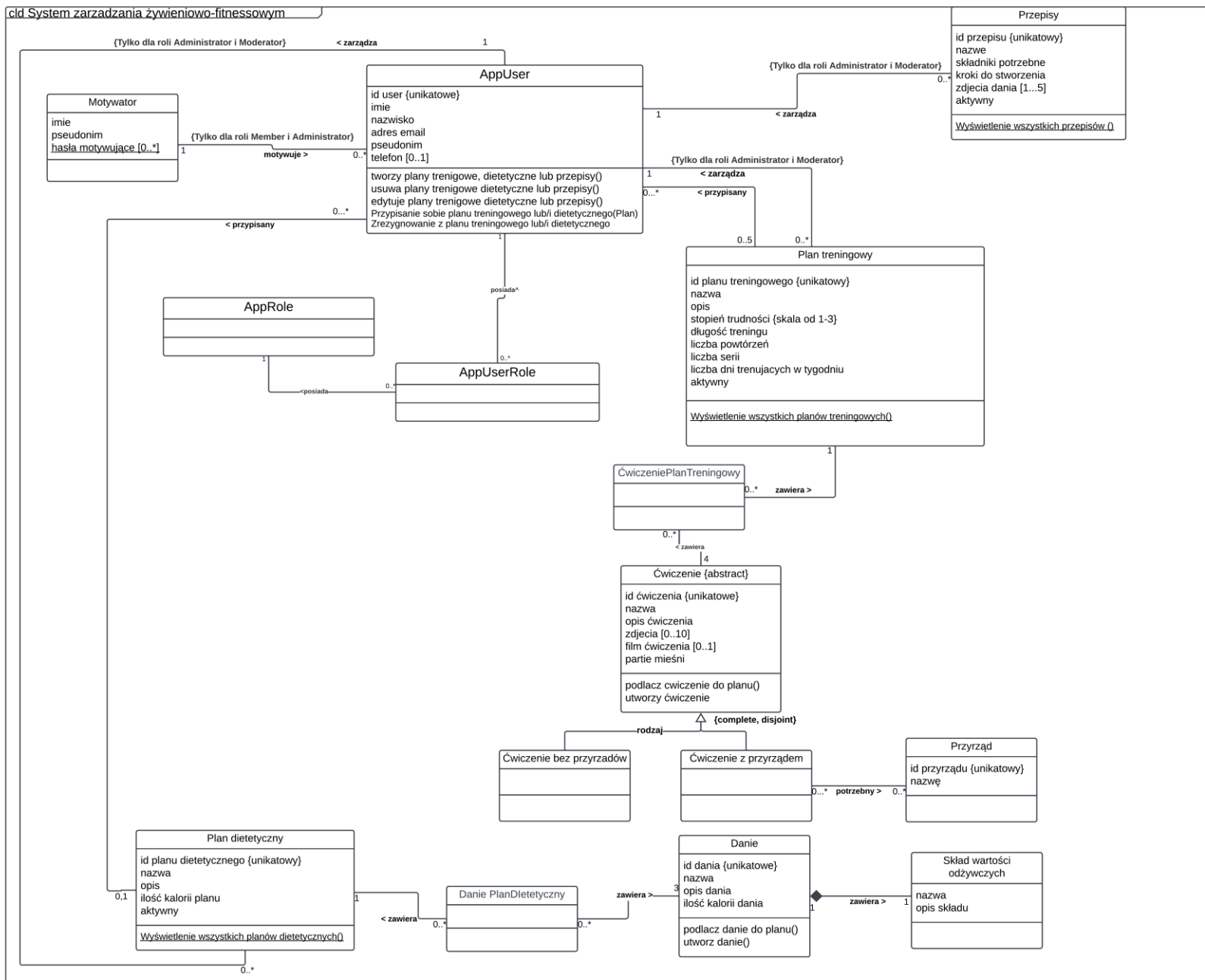
FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

3.3 Diagram stanu dla klasy



FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

3.4 Diagram klas projektowy



FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

3.5 Analiza dynamiczna skutki

W systemie zostanie zastosowany ASP.NET Core Identity, który będzie odpowiedzialny za nadzorowanie i zarządzanie kontami użytkowników. Będzie on posiadać różne role, takie jak Member, Moderator i Administrator. Zostanie stworzony specjalny kontroler, który będzie posiadał odpowiednie metody do tworzenia, edytowania i usuwania planów treningowych, dietetycznych oraz przepisów. System będzie blokował dostęp dla użytkowników bez odpowiednich uprawnień (tylko Moderator i Administrator będą wpuszczani). Każdy Moderator i Administrator będzie posiadał odpowiednie interfejsy użytkownika (UI), umożliwiające tworzenie planu poprzez odpowiednie zapytanie do kontrolera.

Następnie zostanie pokazana opcja dodania do planu dania lub ćwiczenia (także tam będzie można utworzyć danie lub plan). Tworzenie dania lub ćwiczenia będzie rozwiązane poprzez odpowiednie kontrolery dla tych klas, a dodanie będzie połączone z kontrolerem planu treningowego lub dietetycznego. Przy każdym tworzeniu planu dodanie dania lub ćwiczenia będzie wymagane.

Zmiana stanu klasy planu będzie rozwiązana poprzez aktywację planu, dezaktywację lub usunięcie. Podczas tworzenia każdego planu aktor motywator będzie mógł aktywować plan lub ustawić go jako nie aktywny. Aby usunąć plan moderator najpierw musi ustawić go na nie aktywny z co widać na diagramie stanu tej klasy.

Na diagramie projektowym widać nowo utworzone klasy asocjacyjne łączące Plan dietetyczny i danie jak i również plan treningowy i ćwiczenie. Zastosowałem taki rodzaj połączenia aby zminimalizować redundancję danych. Dzięki temu możemy przechować dane dania lub ćwiczenia do wielu planów i odwrotnie.

W diagramie sekwencji można zauważyć, że po dodaniu nowego planu następuje wyświetlenie kolejnego formularza, w którym należy podać dodatkowe dane. W celu optymalizacji tych operacji podczas pracy z bazą danych oraz endpointami w kontrolerze, zostaną wykorzystane metody asynchroniczne. Zastosowanie metod asynchronicznych pozwoli na lepsze zarządzanie czasem wykonywania operacji, co może przyspieszyć działanie aplikacji. Wykorzystanie asynchroniczności może również zwiększyć skalowalność systemu, umożliwiając obsługę większej liczby równoczesnych żądań.

3.6 Decyzje projektowe

1. Ekstensje klas planów, przepisów, dań, ćwiczeń i użytkowników zostaną rozwiązane poprzez wykorzystanie bazy danych z użyciem Entity Framework (EF) z podejściem Code-First. Wybrano to podejście, ponieważ zapewnia ono elastyczność, wygodę i trwałość danych. Baza danych będzie odzwierciedlać strukturę klas i relacje między nimi, umożliwiając łatwe zarządzanie danymi aplikacji.
2. Na diagramie klas można zauważyć podobieństwo między klasami "PlanTreningowy" i "PlanDietetyczny". Choć na pierwszy rzut oka wydaje się, że moglibyśmy zastosować dziedziczenie z ogólnego planu, zdecydowano się na inną strategię. Zamiast tego, zaimplementowano dwie oddzielne klasy, co daje większą swobodę w dołączaniu i rozwoju poszczególnych elementów. Powtarzanie kodu jest praktycznie ograniczone do powtórzenia dwóch atrybutów, a takie rozwiązanie pozwala na bardziej elastyczną rozbudowę aplikacji w przyszłości. Dodawanie, edytowanie i usuwanie planów będzie realizowane poprzez wywołanie odpowiednich endpointów udostępnionych przez kontrolery.
3. Dania będą powiązane relacją kompozycji z klasą "SkładWartosciOdzywczych". W celu zapewnienia bezpieczeństwa i integralności danych, zastosowany zostanie prywatny konstruktor dla klasy "SkładWartosciOdzywczych", co uniemożliwi tworzenie instancji tej klasy spoza klasy dania. Podczas dodawania dania do planu, zostanie automatycznie utworzony i przypisany skład wartości odżywczych odpowiedni dla danego dania.
4. Autoryzacja i uwierzytelnianie będą realizowane za pomocą ASP.NET Core Identity. System wykorzysta wbudowane funkcje Identity, takie jak rejestracja użytkowników, logowanie, resetowanie hasła itp. To zapewni bezpieczny dostęp do aplikacji i kontrolę uprawnień użytkowników.

FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

5. Aby zminimalizować redundancję danych, zdecydowano się na stosowanie relacji wiele-do-wielu między planami treningowymi/dietetycznymi a daniami/ćwiczeniami. Wykorzystanie tabel asocjacyjnych pozwoli na przechowywanie informacji o przynależności dania/ćwiczenia do wielu planów treningowych/dietetycznych oraz odwrotnie.
6. Do generowania interfejsu użytkownika (UI) dla różnych ról (Member, Moderator, Administrator) zastosowany zostanie framework Angular, który zapewni responsywność, modułowość i interaktywność aplikacji. Angular po stronie klienta będzie komunikował się z backendem (implementowanym w ASP.NET Core) za pomocą interfejsów API.
7. W celu optymalizacji wydajności systemu, zostaną zaimplementowane mechanizmy pamięci podręcznej (cache) dla często używanych danych. Na przykład, dane użytkowników mogą być przechowywane w pamięci podręcznej, aby uniknąć częstych zapytań do bazy danych i przyspieszyć operacje uwierzytelniania i autoryzacji.
8. Aby zapewnić integralność danych i uniknąć błędów podczas tworzenia planów treningowych lub dietetycznych, zostaną zastosowane mechanizmy walidacji danych. Przykładowo, można wykorzystać atrybuty walidacji dostępne w ASP.NET Core, takie jak wymagane pola, ograniczenia długości, unikalność itp.
9. W celu poprawy wydajności i responsywności aplikacji, zdecydowano się na szerokie wykorzystanie funkcji asynchronicznych. W miejscach, gdzie operacje mogą być czasochłonne lub wymagają komunikacji z zewnętrznymi serwisami, takimi jak pobieranie danych z bazy danych, wysyłanie żądań HTTP lub przetwarzanie plików, zostaną użyte metody asynchroniczne.
10. Każda asocjacja w projekcie będzie stworzona z automatycznymi połączeniami zwrotnymi.
11. Na diagramie projektowym widać także dużą zmianę dotyczące Moderatora i Klienta. Te klasy zostały usunięte aby zastosować ASP.NET Core Identity. Doszły 3 klasy AppUser, AppRole i klasa asocjacyjna AppUserRole. Każdy użytkownik posiada dzięki temu własną rolę, która nadaje uprawnienia jakie ma i do czego dokładnie ma dostęp. Jest to lepsze rozwiązanie niż wcześniej ponieważ w bardzo łatwy sposób możemy dodać nowego użytkownika z innymi uprawnieniami. Na diagramie nie zostały umieszczone pola hasło użytkownika ponieważ Identity sam automatycznie tworzy takie pola i nie jest wymagane tego umieszczenie.
Dużym plusem zastosowania tej metody także jest brak potrzeby implementacji wielodziedziczenia dotyczące klient premium ponieważ wystarczy utworzyć rolę MemberPremium do niego.
12. Na diagramie projektowym widać ograniczenia do asocjacji, będą one dostępne tylko dla danych użytkowników systemu, którzy mają odpowiednie role. Z takich jak Member, Administrator lub Moderator.

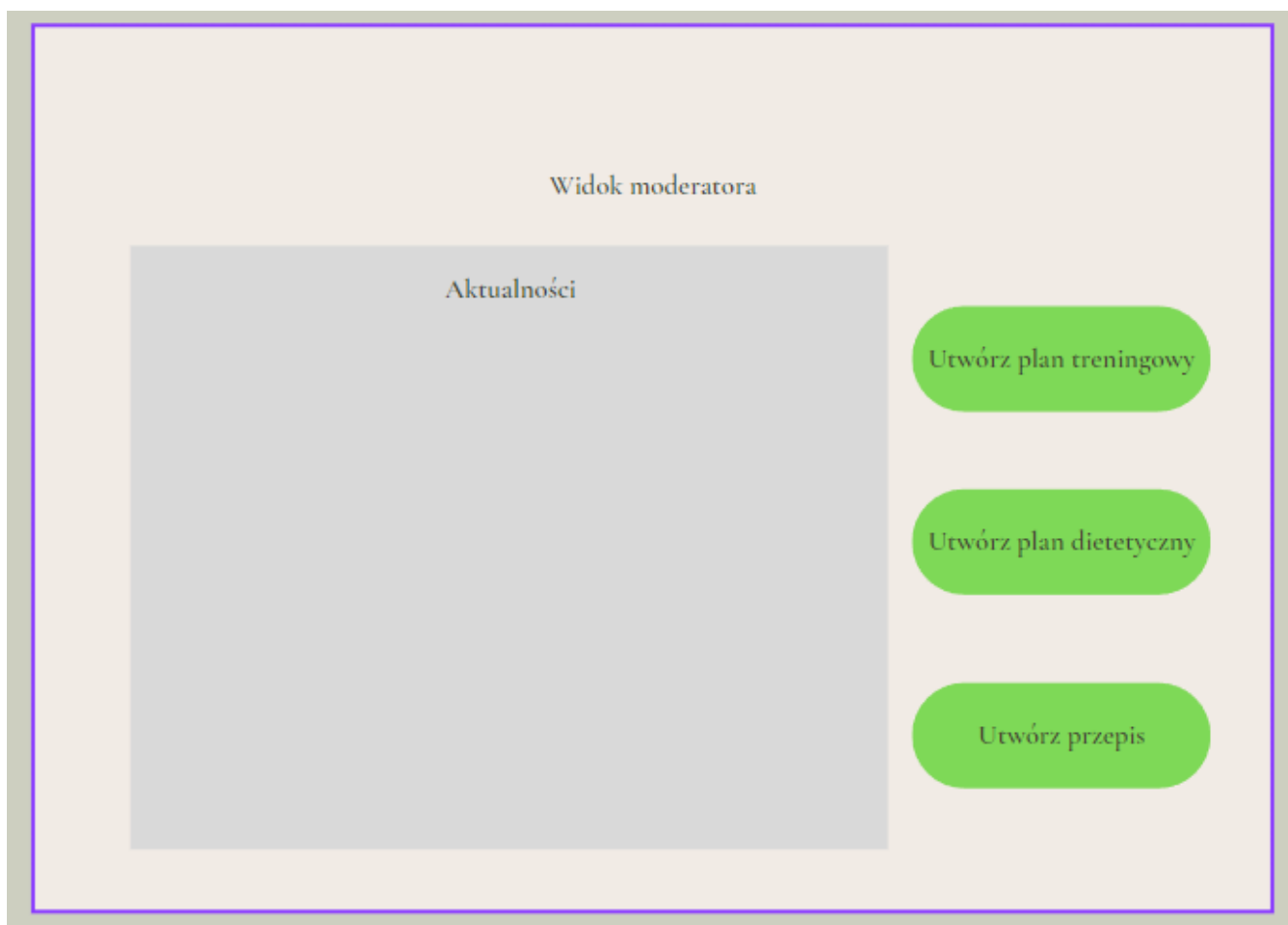
FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

4. Projekt GUI

4.1 Opis projektu GUI

W poniższych zdjęciach zostanie pokazany prototyp interfejsu użytkownika do wyżej opisanego przypadków użycia. Interfejsu posiada przyciski, które przenoszą do kolejnych widżetów. Każdy plan dietetyczny czy treningowy mają inne formularze, które odnoszą się do nich. Potem jest wyświetlany widżet, w którym trzeba wybrać z listy danie odnoszące się do planu dietetycznego lub ćwiczenie odnoszące się do planu treningowego. Można podczas dodania tych opcji stworzyć nowe danie lub ćwiczenie. Przepis posiada tylko formularz do dania danych. Po wpisaniu złych danych w formularzu zostanie wyświetlona na dole informacja o tym pod rubryką gdzie jest błąd.

4.1.1 Widżet tworzenia planu dietetycznego, treningowego lub przepisu



FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

4.1.2 Formularz do dodania danych planu dietetycznego, treningowego lub przepisu

Utwórz plan dietetyczny

Wpisz nazwę

nazwa...

Wpisz opis

opis...

Wpisz ilość kalorii planu

ilosc kalorii...

Aktywuj plan

aktywacja planu...

Anuluj

Zapisz

FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

4.1.3 Formularz do dodania dania lub ćwiczenia do planu dietetycznego lub treningowego

Dodaj danie do plan
dietetyczny

Podaj danie pierwsze

wybierz z listy dania..

Podaj danie drugie

wybierz z listy dania..

Podaj danie trzecie

wybierz z listy dania..

Stwórz danie

Dodaj kolejne danie

Anuluj

Zapisz

FueliciousFit	Wersja: v0.0001
Dokument projektu na MAS	Data: 30.03.2023

4.1.4 Formularz tworzenia do dania lub ćwiczenia

Utwórz danie

Wpisz nazwę

nazwa...

Wpisz opis

opis...

Wpisz ilość kalorii dania

ilosc kalorii...

Anuluj

Zapisz