Software Requirements Specification

for

Aplikacja webowa do wyszukiwania przepisów

**Version 1.2 approved**

**Prepared by Ewa Swatowska & Paweł Goliszewski**

**Uniwersytet Jagielloński**

**11.06.2019**

**Table of Contents**

[1. Introduction 4](#_Toc11183686)

[1.1 Purpose 4](#_Toc11183687)

[1.2 Document Conventions 4](#_Toc11183688)

[1.3 Intended Audience and Reading Suggestions 4](#_Toc11183689)

[1.4 Product Scope 4](#_Toc11183690)

[1.5 References 5](#_Toc11183691)

[2. Overall Description 5](#_Toc11183692)

[2.1 Product Perspective 5](#_Toc11183693)

[2.2 Product Functions 5](#_Toc11183694)

[2.3 User Classes and Characteristics 5](#_Toc11183695)

[2.4 Operating Environment 6](#_Toc11183696)

[2.5 Design and Implementation Constraints 6](#_Toc11183697)

[2.6 User Documentation 6](#_Toc11183698)

[2.7 Assumptions and Dependencies 6](#_Toc11183699)

[3. External Interface Requirements 6](#_Toc11183700)

[3.1 User Interfaces 6](#_Toc11183701)

[3.2 Hardware Interfaces 8](#_Toc11183702)

[3.3 Software Interfaces 9](#_Toc11183703)

[3.4 Communications Interfaces 9](#_Toc11183704)

[4. System Features 10](#_Toc11183705)

[4.1 Wyszukiwanie przepisów z wyszukaniem filtrów 10](#_Toc11183706)

[4.2 Wyszukiwanie ostatnio odwiedzonych przepisów 11](#_Toc11183707)

[4.3 Ocena przepisu 12](#_Toc11183708)

[4.4 Dodanie przepisu do bazy danych 13](#_Toc11183709)

[4.5 Skalowanie ilości składników do liczby osób 14](#_Toc11183710)

[5. Other Nonfunctional Requirements 14](#_Toc11183711)

[5.1 Performance Requirements 14](#_Toc11183712)

[5.2 Safety Requirements 14](#_Toc11183713)

[5.3 Security Requirements 14](#_Toc11183714)

[5.4 Software Quality Attributes 14](#_Toc11183715)

[5.5 Business Rules 15](#_Toc11183716)

[6. Other Requirements 15](#_Toc11183717)

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Ewa Swatowska | 8.06.2019 | Uaktualniona wersja po 2 miesiącach pracy (m. in. dodano diagramy czynności). | 1.1 |
| Paweł Goliszewski | 11.06.2019 | Dodano informacje o powodach wyboru narzędzi zgodnie z wymaganiami prowadzącej przedmiot. | 1.2 |

**Członkowie zespołu:**

* **KIEROWNIK / PROGRAMISTA BACKEND:** Paweł Goliszewski
* **INŻYNIER SYSTEMOWY:** Ewa Swatowska
* **PROGRAMISTKA / TESTER FRONTEND:** Alicja Gorzkowicz
* **TESTER SYSTEMOWY:** Justyna Apryasz
* **TESTER JEDNOSTKOWY / INTEGRACYJNY:** Mateusz Nowak

# Introduction

## Purpose

Przenaczeniem tego dokumentu jest przedstawienie w ustandaryzowany sposób wymagań, co do aplikacji, będącej częścią projektu zaliczeniowego “Aplikacja webowa do wyszukiwania przepisu” na przedmiot Inżynieria Oprogramowania, realizowany w semestrze letnim roku akademickiego 2018/2019 na Uniwersytecie Jagiellońskim.

## Document Conventions

W tym dokumencie nie zastosowano dodatkowych konwencji, oznaczeń.

## Intended Audience and Reading Suggestions

Dokument składa się z 6. części. Pierwsza zawiera wstęp. Druga mówi o ogólnych cechach produktu. Trzecia dotyczy wymagań systemowych. Czwarta mówi o wymaganiach funkcjonalnych i jest szczególnie istotna dla programistów i testerów. Piąta dotyczy wybranych wymagań niefunkcjonalnych tj. wymagań wydajnościowych, bezpieczeństwa, jakościowych czy biznesowych - ze względu na to, że nie jest planowane jej udostępnienie na szeroką skalę, ta sekcja nie jest rozbudowana. Ostatnia część - szósta - dotyczy pozostałych wymagań i zawiera elementy takie jak: glosariusz, diagramy przypadków użycia, diagramy czynności oraz instrukcje obsługi dla administratora oraz użytkownika, a także wskazówki co do instalacji.

## Product Scope

Głównym celem stworzonej przez nas aplikacji jest pomoc w wyszukiwaniu przepisów na podstawie wprowadzonych oczekiwanych i niechcianych składników. Aplikacja zawiera własną bazę, w której przetrzymywane są przepisy. Przykładowe zastosowanie naszego produktu: szukanie potrawy, którą użytkownik może przyrządzić na podstawie produktów znajdujących się w lodówce oraz dostosowanie swoich składników do ilości osób (przeskalowanie).

Produkt ma formę aplikacji webowej.

## References

Lista wymagań wraz z dodanymi wymaganiami zaproponowanymi przez inną grupę dodana w osobnym pliku (plik o nazwie Lista wymagań).

# Overall Description

## Product Perspective

Aplikacja składa się z 2 elementów współpracujących ze sobą: frontendu oraz backendu. Frontend odpowiada za obsługę aplikacji po stronie klienta w przeglądarce internetowej i interfejs wyświetlany użytkownikowi. Backend odpowiada za obsługę aplikacji po stronie serwera oraz za łączność z bazą danych.

## Product Functions

Użytkownik ma możliwość wyszukiwania przepisów z wykorzystaniem różnorodnych filtrów - w tym filtrów pozytywnych (dany składnik występuje), czy filtrów negatywnych (dany składnik nie może się znaleźć, ich poziomie trudności, czy czasie przygotowania.

Przepisy mogą być oceniane.

Może także mieć dostęp w odpowiedniej sekcji strony do ostatnio odwiedzonych przepisów.

## User Classes and Characteristics

Będą 2 typy użytkowników: zwykli użytkownicy oraz administrator:

* zwykli użytkownicy: dostęp do strony głównej; wyszukiwanie, skalowanie i ocena przepisów
* administrator: możliwość logowania do panelu administracyjnego i zarządzania przepisami (dodawanie, modyfikowanie, usuwanie)

## Operating Environment

Apikacja korzysta z narzędzia Docker, które umożliwia łatwe jej uruchomienie wraz z bazą danych na większości powszechnie stosowanych systemów operacyjnych (z rodziny MacOS, Unix, czy Windows).

## Design and Implementation Constraints

Wykorzystywane narzędzia i języki zostały opisane w punkcie 3.3.

## User Documentation

Instrukcja obsługi dla użytkownika jest załączona w osobnym pliku (razem z instrukcją obsługi dla administratora i wskazówkami instalacji).

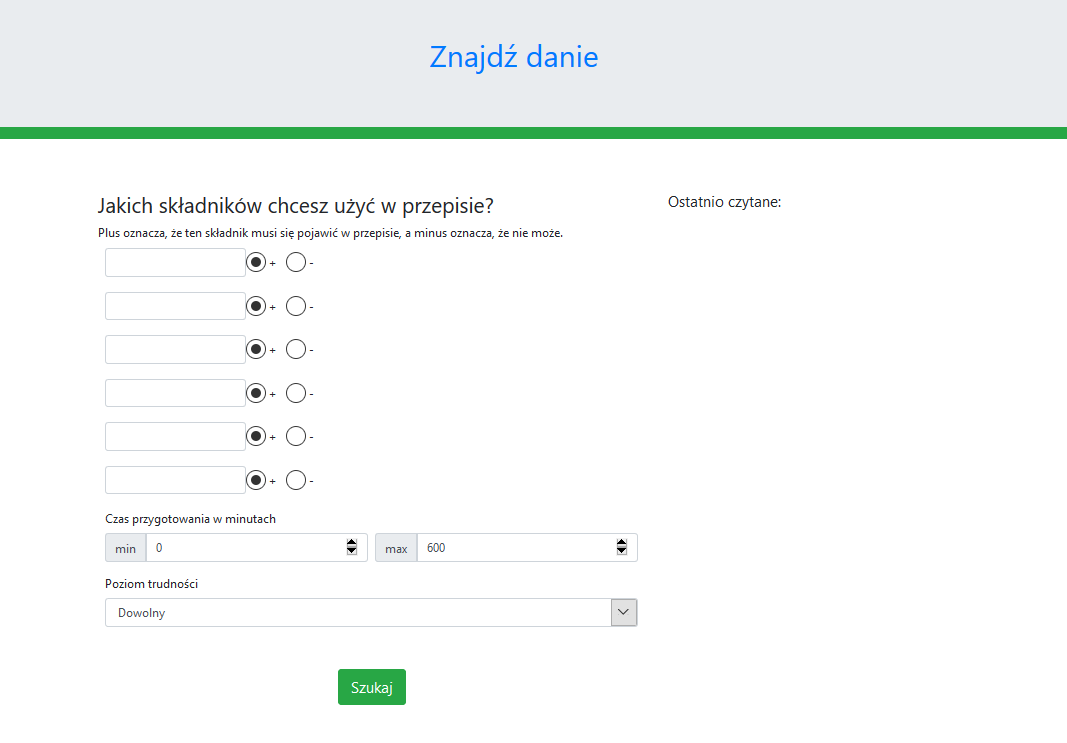
## Assumptions and Dependencies

Cała aplikacja jest tworzona przez nasz zespół, korzystaliśmy z narzędzi open source, więc nie musimy spełniać żadnych istotnych wymagań prawnych, czy oczekiwać od innego zespołu wykonania elementu aplikacji.

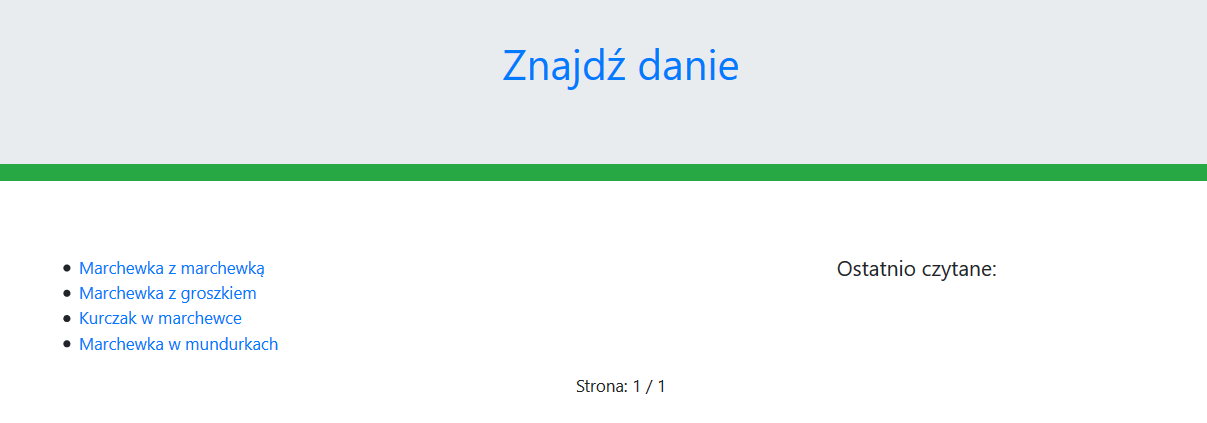
# External Interface Requirements

## User Interfaces

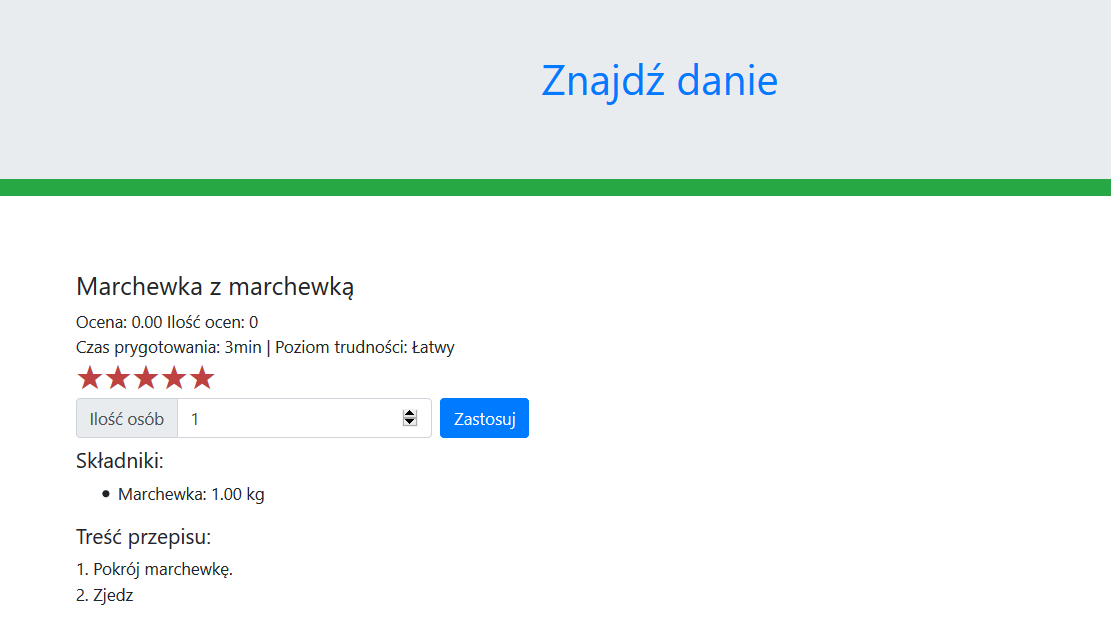
Interfejs użytkownika po wejściu na stronę główną



Interfejs użytkownika po wprowadzeniu składników (tu Marchewka)



Interfejs użytkownika po wybraniu przepisu (tu Marchewka z marchewką)



## Hardware Interfaces

Aplikacja jest responsywna - tj. dostosowuje umieszczenie elementów interfejsu, czy ich rozmiar do rozmiaru ekranu, czy okna przeglądarki. W szczególności aplikacja udostępnia wszystkie elementy interfejsu na ekranach telefonów komórkowych, tabletów, czy monitorach komputerów osobistych.

## Software Interfaces

Backend aplikacji został napisany w języku Python 3.7 wykorzystując framework Django w wersji 2.2. Został użyty ten framework ze względu na doświadczenie programistki (niestety nieobecnej już w zespole) w jej stosowaniu oraz zaimplementowane w nim wzorce projektowe i architektoniczne. W szczególności:

* MVC – wzorzec architektoniczny model-widok-kontroler zapewniający podział aplikacji na 3 główne części: modele – będące reprezentacją logiki aplikacji (w szczególności w Django instancje modeli mają swoje tabele w bazie danych, gdzie mogą być zapisane), widoki – opisują jak wyświetlić logikę aplikacji w ramach interfejsu użytkownika oraz kontrolery – przyjmują one dane od użytkownika i przetwarzają ją poprzez aktualizacje modeli i odświeżanie widoków.
* Metoda szablonowa – czynnościowy wzorzec architektoniczny używany w Django przy generowaniu widoków za pomocą szablonów. Każdy widok jest generowany w ten sam sposób – wykorzystujemy metodę przyjmującą argument request (i opcjonalne argumenty dodatkowe), który zawiera informacje wysłane przez przeglądarkę użytkownika, które przetwarzamy w tej metodzie. Na tej podstawie generujemy odpowiedź najczęściej wykorzystując szablon napisany w technologiach front-endowych.
* ActiveRecord – wzorzec architektoniczny zapewniający komunikację z bazą danych za pomocą obiektów będących reprezentacją danych w tabeli relacyjnej bazy danych. W szczególności programista zamiast wywoływać bezpośrednio zapytania SQL dodający wiersz do tabeli może stworzyć obiekt odpowiedniej klasy (będącej potomkiem klasy modelu) i wywołać na nim operację save().

Frontend aplikacji został napisany z wykorzystaniem HTML, CSS, JavaScript oraz frameworku Bootstrap w wersji 4. Został on użyty ze względu na to, że zapewnia responsywność aplikacji.

Wykorzystywana jest baza danych PostgreSQL.

Sama aplikacja oraz baza danych są uruchamiane w kontenerach dockerowych.

## Communications Interfaces

Komunikacja z użytkownikiem odbywa się przez przeglądarkę internetową z wykorzystaniem protokołu HTTP. Część interakcji z użytkownikiem wykorzystuje technologię JavaScriptową AJAX. Technologia AJAX została użyta w celu zapewnienia większej wygody użytkownika I zmniejszenia ruchu w sieci – strona nie musi być przeładowywana całkowicie w celu interakcji z użytkownikiem.

# System Features

## Wyszukiwanie przepisów z wyszukaniem filtrów

4.1.1 Description and Priority

Jest to podstawowa funkcjonalność systemu. Założyliśmy, że użytkownik może wyszukać odpowiadające mu przepisy na podstawie wybranych filtrów.

4.1.2 Stimulus/Response Sequences

1. Użytkownik odwiedza stronę, wprowadza pożądane i niepożądane składniki.
2. Serwer wysyła zapytanie do bazy danych.
3. Na podstawie odpowiedzi jest generowana strona z wynikami zapytania jako lista przepisów.
4. Użytkownik wybiera preferowany przepis poprzez kliknięcie.
5. Serwis wyświetla użytkownikowi pełny przepis.

Scenariusze alternatywne:

* Użytkownik wybiera składnik nieobecny w bazie danych.

W odpowiedzi użytkownik dostaje komunikat "brak takiego składnika w bazie".

* Użytkownik wybiera niedostępną kombinację składników (np. "majonez i truskawki", co jest bardzo nietypowym połączeniem).

W odpowiedzi w pkt. 3 użytkownik dostaje komunikat "brak przepisów spełniających wymagania".

4.1.3 Functional Requirements

REQ-2: Filtrowanie przepisów po wystąpieniu składników, możliwość używania wielu filtrów – pozytywne (dany składnik musi wystąpić) i negatywne (dany składnik nie może wystąpić).

REQ-3: Suma ilości filtrów wystąpienia składników (REQ-2) przy danym wyszukiwaniu nie może przekroczyć 6.

REQ-7: Każdy przepis ma przypisany poziom trudności i czas przygotowania, a użytkownik ma możliwość filtrowania po nich.

REQ-9: 10 wyników na stronie, kolejne wyniki można przeglądać na następnych podstronach (paginacja).

REQ-10: Tekstowy (po nazwie) wybór składników z bazy przy wyszukiwaniu

REQ-11: Auto-uzupełnianie nazw składników przy wpisywaniu (wybór z listy)

## Wyszukiwanie ostatnio odwiedzonych przepisów

4.2.1 Description and Priority

Użytkownik może odwiedzić kilka podstron z przepisami w serwisie. Ostatnie kilka przepisów jest mu wyświetlane w postaci linków w odpowiedniej sekcji strony.

4.2.2 Stimulus/Response Sequences

1. Użytkownik odwiedza stronę, gdzie w części "ostatnio odwiedzone przepisy" ma wyszczególnione kilka ostatnio odwiedzonych przepisów.
2. Użytkownik wybiera jeden z przepisów.
3. Serwer wysyła zapytanie do bazy danych o treść danego przepisu.
4. Na podstawie odpowiedzi jest generowana strona z danym przepisem.

Scenariusze alternatywne:

* Użytkownik nie odwiedził żadnego przepisu, usunął ciasteczka, ma wyłączone ciasteczka lub skorzystał z innej przeglądarki/komputera.

Lista ostatnich przepisów jest pusta.

4.2.3 Functional Requirements

REQ-8: Zapamiętywanie 5 ostatnio wyświetlonych przepisów za pomocą ciasteczek.

## Ocena przepisu

4.3.1 Description and Priority

Użytkownik odwiedzając strony z przepisami ma możliwość jednokrotnej ich oceny.

4.3.2 Stimulus/Response Sequences

1. Użytkownik wyszukuje przepis.
2. Użytkownik widzi średnią ocen wyszukanych przepisów z dokładnością do 2 miejsc po przecinku i wybiera interesujący go przepis.
3. Użytkownik widzi średnią ocen przepisu z dokładnością do 2 miejsc po przecinku i wyświetlone obok 5 gwiazdek, które są zamalowane w liczbie równej średniej ocen.
4. Po najechaniu na gwiazdki, stają się one białe w środku i zamalowują się w zależności od pozycji myszki.
5. Użytkownik wybiera ocenę przepisu od 1(zły) do 5(dobry) poprzez kliknięcie na jedną z 5 gwiazdek, gdzie pierwsza gwiazdka odpowiada ocenie 1 a każda następna o 1 większej.
6. Użytkownik dostaje informację zwrotną, że wystawił ocenę poprzez trwałe zamalowanie ilości gwiazdek i nie może już jej zmienić. Najechanie myszką na gwiazdki nic nie zmienia.
7. Ocena użytkownika wliczana jest do średniej.

Scenariusze alternatywne:

* Ponowna próba oceny przepisu.

Użytkownik dostaje informację, że ten przepis już ocenił.

* Ponowna próba oceny przepisu na innej przeglądarce lub po wyczyszczeniu ciasteczek

Użytkownik może ocenić przepis.

* Dodawanie oceny przepisu w trybie prywatnym przeglądarki.

Po wyjściu z trybu prywatnego użytkownik może jeszcze raz ocenić przepis.

* Przeglądarka nie obsługuje ciasteczek albo są wyłączone.

Po kliknięciu na gwiazdkę, użytkownik dostaje informację, że nie może wystawić oceny i powinien włączyć ciasteczka jeżeli chce to zrobić.

4.3.3 Functional Requirements

REQ-12: system oceniania w postaci gwiazdek (od 1 do 5) potraw:

- odporny na możliwość złożenia wielu głosów przez jedną osobę

- pokazuję średnią arytmetyczną głosów

## Dodanie przepisu do bazy danych

4.4.1 Description and Priority

Administrator ma możliwość dodania (analogicznie modyfikacji i usunięcia) przepisu do bazy danych (listy składników, kroków, które należy przeprowadzić, aby przygotować potrawę) wraz z możliwością dodania zdjęcia.

4.4.2 Stimulus/Response Sequences

1. Administrator loguje się do panelu administratora.
2. Administrator wprowadza nazwę dodawanego przepisu.
3. Administrator określa liczbę składników potrzebnych do wykonania przepisu dla jednej osoby. W razie potrzeby dodaje składniki do bazy.
4. Administrator opisuje kolejne kroki przepisu.
5. Administrator dodaje zdjęcie (opcjonalne).
6. Administrator zatwierdza wpisany przepis.
7. Przepis jest dostępny w bazie.

Scenariusze alternatywne:

* Użytkownik niepoprawnie loguje się do bazy (błędny login lub hasło).

Dostaje komunikat o błędnym logowaniu.

* Administrator próbuje dodać przepis o nazwie już istniejącej w bazie.

Dostaje informację że dany przepis już istnieje.

4.4.3 Functional Requirements

REQ-5: Dodawanie przepisów do bazy przez administratora.

REQ-13: Zdjęcia potraw ilustrujące przykładowe podanie potrawy.

## Skalowanie ilości składników do liczby osób

4.4.1 Description and Priority

Użytkownik po wyświetleniu przepisu ma możliwość przeskalowania składników w stosunku do liczby osób (domyślnie wyświetla się ilość składników dla 1 osoby).

4.4.2 Stimulus/Response Sequences

1. Użytkownik wyświetla przepis.
2. Użytkownik wybiera z rozwijanej listy ilość osób, dla której ma być przeznaczony przepis.
3. Wartości ilości składników aktualizują się w przeglądarce użytkownika.

4.4.3 Functional Requirements

REQ-6: Skalowanie składników do liczby porcji.

# Other Nonfunctional Requirements

## Performance Requirements

Nie stwierdzono istotnych wymagań wydajnościowych.

## Safety Requirements

Nie stwierdzono istotnych wymagań ze względu na odporność aplikacji.

## Security Requirements

Dostęp do panelu administratora wykorzystuje system użytkowników frameworku Django i jest zabezpieczony hasłem (REQ-6).

## Software Quality Attributes

Aplikacja jest responsywna - tj. dostosowuje umieszczenie elementów interfejsu, czy ich rozmiar do rozmiaru ekranu, czy okna przeglądarki. W szczególności aplikacja udostępnia wszystkie elementy interfejsu na ekranach telefonów komórkowych, tabletów, czy monitorach komputerów osobistych (REQ-1).

Użytkownik ma możliwość jednokrotnego ocenienia każdego przepisu. Zabezpieczenie przed wielokrotnym ocenieniem jest realizowane za pomocą ciasteczek (REQ-12).

## Business Rules

Nie dotyczy.

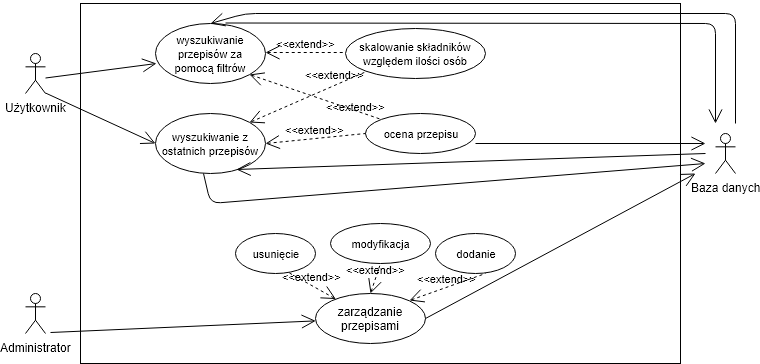
# Other Requirements

**Appendix A: Glossary**

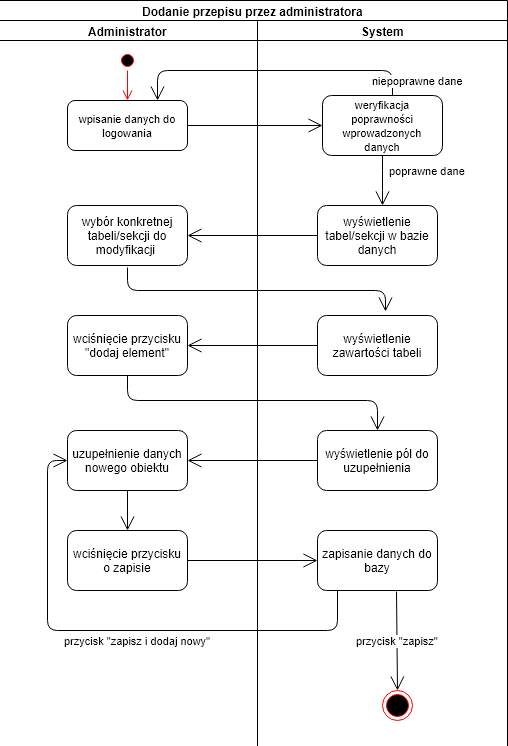
1. REQ-# (gdzie # jest pewną liczbą) oznacza wymaganie nr # z dokumentu z wymaganiami

**Appendix B: Analysis Models**

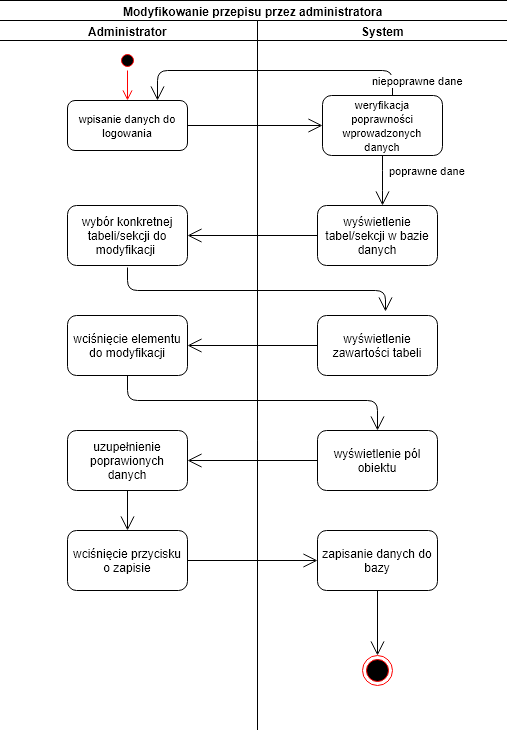
**Diagram przypadków użycia**:



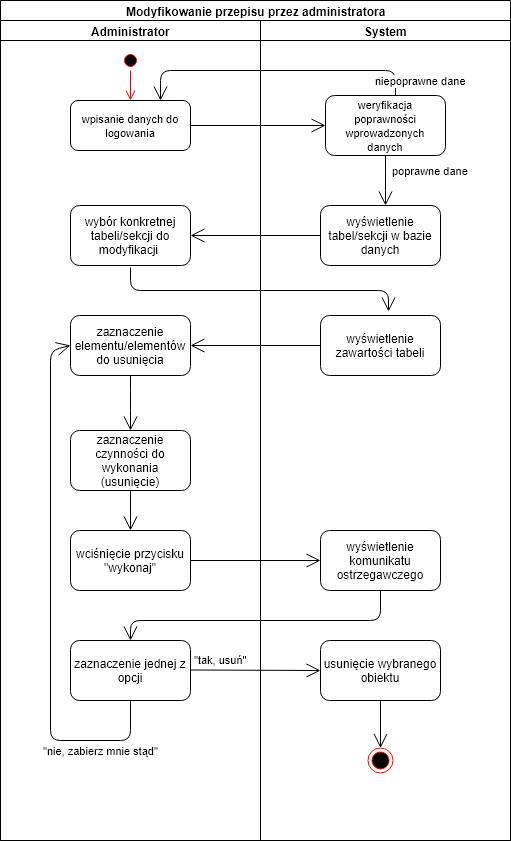
**Diagram czynności dla przypadku użycia “dodanie przepisu przez administratora”:**

****

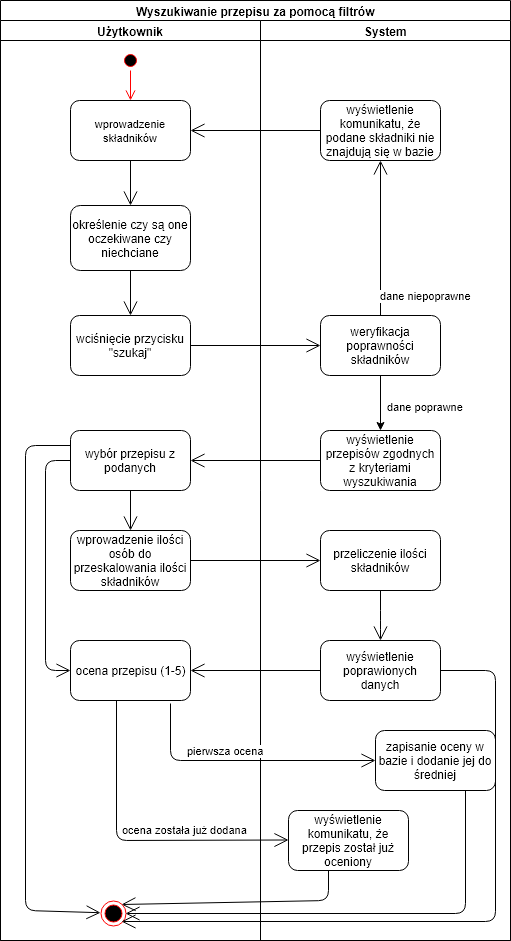
**Diagram czynności dla przypadku użycia “modyfikacja przepisu przez administratora”:**

****

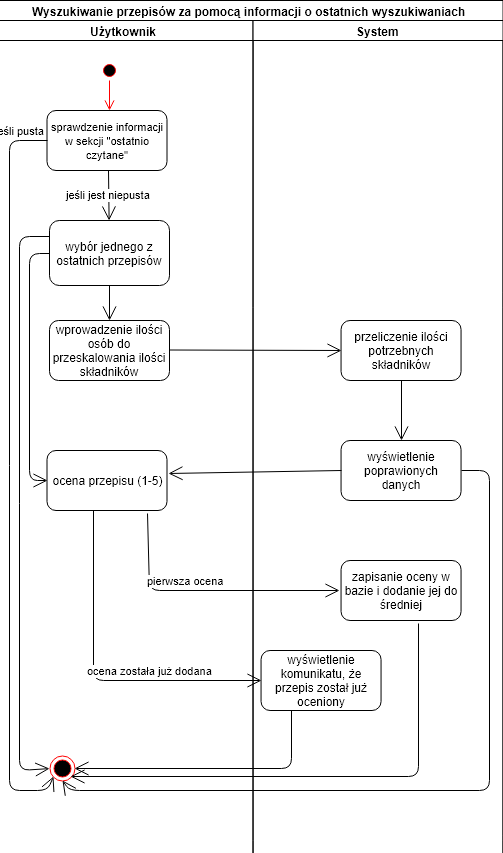
**Diagram czynności dla przypadku użycia “usunięcie przepisu przez administratora”:**

****

**Diagram czynności dla przypadku użycia “wyszukiwanie przepisu za pomocą filtrów”:**

****

**Diagram czynności dla przypadku użycia “wyszukiwanie ostatnio oglądanych przepisów”:**

****