Software Requirements Specification

for

Aplikacja webowa do wyszukiwania przepisów

**Version 1.0 approved**

**Prepared by Paweł Goliszewski**

**Uniwersytet Jagielloński**

**12.04.2019**

**Table of Contents**

[**Introduction**](#_3znysh7) **4**

[**Purpose**](#_2et92p0) **4**

[**Document Conventions**](#_tyjcwt) **4**

[**Intended Audience and Reading Suggestions**](#_3dy6vkm) **4**

[**Product Scope**](#_1t3h5sf) **4**

[**References**](#_4d34og8) **4**

[**Overall Description**](#_2s8eyo1) **4**

[**Product Perspective**](#_17dp8vu) **4**

[**Product Functions**](#_3rdcrjn) **5**

[**User Classes and Characteristics**](#_26in1rg) **5**

[**Operating Environment**](#_lnxbz9) **5**

[**Design and Implementation Constraints**](#_35nkun2) **5**

[**User Documentation**](#_1ksv4uv) **5**

[**Assumptions and Dependencies**](#_44sinio) **5**

[**External Interface Requirements**](#_2jxsxqh) **5**

[**User Interfaces**](#_z337ya) **5**

[**Hardware Interfaces**](#_3j2qqm3) **6**

[**Software Interfaces**](#_1y810tw) **6**

[**Communications Interfaces**](#_4i7ojhp) **6**

[**System Features**](#_2xcytpi) **6**

[**Wyszukiwanie przepisów z wyszukaniem filtrów**](#_1ci93xb) **6**

[**Wyszukiwanie ostatnio odwiedzonych przepisów**](#_3whwml4) **7**

[**Ocena przepisu**](#_3oyd76ela77i) **7**

[**Dodanie przepisu do bazy danych**](#_td28oeq8l4h0) **8**

[**Skalowanie ilości składników do liczby osób**](#_110cs89t5pmo) **9**

[**Other Nonfunctional Requirements**](#_2bn6wsx) **9**

[**Performance Requirements**](#_qsh70q) **9**

[**Safety Requirements**](#_3as4poj) **9**

[**Security Requirements**](#_1pxezwc) **9**

[**Software Quality Attributes**](#_49x2ik5) **10**

[**Business Rules**](#_2p2csry) **10**

[**Other Requirements**](#_147n2zr) **10**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

Celem 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 itp.

## 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, listę elementów, które należy uzupełnić w tym dokumencie.

## Product Scope

Aplikacja ma za zadanie wyświetlać dostępne przepisy z własnej bazy na podstawie wprowadzonych produktów. Wyobrażamy sobie następujące przykładowe zastosowanie: szukanie potrawy, którą użytkownik może przyrządzić na podstawie produktów znajdujących się w lodówce. Produkt ma mieć formę aplikacji webowej.

## References

[1] Lista wymagań wraz z dodanymi wymaganiami zaproponowanymi przez inną grupę: <https://drive.google.com/open?id=1Wf9nirlyWLraiWfClngcY1Tvn4i3-kAao9glasud9r8>

# 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 mieć możliwość wyszukiwania przepisów z wykorzystaniem różnorodnych filtrów - w tym filtrów pozytywnych (dany składnik występuje), czy filtrów negatywnych (dany składnik nie może się znaleźć).

Przepisy mogą być oceniane, czy zawierać zdjęcia. Użytkownik może też filtrować po ocenach przepisów, ich poziomie trudności, czy czasie przygotowania przy wyszukiwaniu. 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 mają dostęp do strony, mogą wyszukiwać i oceniać przepisy.

Administrator ma możliwość logowania do panelu administracyjnego i dodawania przepisów.

## Operating Environment

Nie postawiono szczegółowych wymagań, co do środowiska, na którym operuje aplikacja.

## Design and Implementation Constraints

Wykorzystywane narzędzia i języki zostały opisane w punkcie 3.3.

## User Documentation

TBD

## Assumptions and Dependencies

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

# External Interface Requirements

## User Interfaces

TBD

## Hardware Interfaces

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

## Software Interfaces

Backend aplikacji zostanie napisana w języku Python 3.7 wykorzystując framework Django w wersji 2.2.

Frontend aplikacji zostanie napisany z wykorzystaniem frameworku Bootstrap w wersji 4.

Wykorzystywana będzie baza danych PostgreSQL.

Sama aplikacja oraz baza danych będą uruchamiane w kontenerach dockerowych.

## Communications Interfaces

Komunikacja z użytkownikiem będzie się odbywała z użytkownikiem przez przeglądarkę internetową z wykorzystaniem protokołu HTTP. Część interakcji z użytkownikiem będzie wykorzystywać technologię JavaScriptową AJAX.

# System Features

## Wyszukiwanie przepisów z wyszukaniem filtrów

4.1.1 Description and Priority

Jest to podstawowa funkcjonalność systemu. Zakładamy, 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, użytkownik 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 może odwiedzić kilka podstron z przepisami w serwisie. Ostatnie kilka przepisów jest mu wyświetlane w postaci linków w odpowiedniej sekcji strony.

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 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 będzie wykorzystywał system użytkowników frameworku Django i będzie zabezpieczony hasłem (REQ-6).

## Software Quality Attributes

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

Użytkownik będzie miał możliwość jednokrotnego ocenienia każdego przepisu. Zabezpieczenie przed wielokrotnym ocenieniem będzie 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 [1]
2. TBD - akronim od “to be determined”, oznacza element dokumentu, który zostanie uzupełniony w przyszłości

**Appendix B: Analysis Models**

Diagram przypadków użycia:



**Appendix C: To Be Determined List**

1. Część 2.6 tego dokumentu - należy przemyśleć, jaką dokumentację dostarczymy dla przyszłego użytkownika.
2. Część 3.1 tego dokumentu - należy uzupełnić o grafiki obrazujące ułożenie elementów interfejsu użytkownika.