Specyfikacja wymagań

Software Requirements Specification

# Wersja

|  |  |  |  |
| --- | --- | --- | --- |
| Wersja | Data | Autor | Zakres zmian |
| 0.1.0 | 2020-10-07 | Łukasz Obłąk  Wojciech Barłowski  Nikola Kowalik  Kamil Pawłowski  Michał Szczepaniak  Kacper Wielągowski |  |

# Spis treści

[Wersja 1](#_Toc52991313)

[Spis treści 1](#_Toc52991314)

[1. Wprowadzenie 2](#_Toc52991315)

[1.1 Cel 2](#_Toc52991316)

[1.2 Przyjęte konwencje 2](#_Toc52991317)

[1.3 Zakres projektu 2](#_Toc52991318)

[1.4 Odwołania 2](#_Toc52991319)

[2. Opis ogólny 2](#_Toc52991320)

[2.1 Perspektywa produktu 2](#_Toc52991321)

[2.2 Klasy i charakterystyki użytkowników 2](#_Toc52991322)

[2.3 Środowisko działania 3](#_Toc52991323)

[2.4 Ograniczenia projektowe i wykonawcze 3](#_Toc52991324)

[2.5 Założenia i zależności 3](#_Toc52991325)

[3. Funkcjonalności systemu 3](#_Toc52991326)

[3.x Funkcjonalność XYZ 3](#_Toc52991327)

[3.x.1 Opis 3](#_Toc52991328)

[3.x.2 Wymagania funkcjonalne 3](#_Toc52991329)

[4. Wymagania dotyczące danych 3](#_Toc52991330)

[4.1 Logiczny model danych 3](#_Toc52991331)

[4.2 Raporty 3](#_Toc52991332)

[4.3 Pozyskiwanie, integralność przechowywanie I usuwanie danych 4](#_Toc52991333)

[5. Wymagania dotyczące interfejsu zewnętrznego 4](#_Toc52991334)

[5.1 Interfejsy użytkownika 4](#_Toc52991335)

[5.2 Interfejsy programowe 4](#_Toc52991336)

[5.3 Interfejsy sprzętowe 4](#_Toc52991337)

[5.4 Interfejsy komunikacyjne 4](#_Toc52991338)

[6. Cechy jakości 5](#_Toc52991339)

[6.1 Użyteczność 5](#_Toc52991340)

[6.2 Wydajność 5](#_Toc52991341)

[6.3 Zabezpieczenia 5](#_Toc52991342)

[6.4 Bezpieczeństwo użytkowania 5](#_Toc52991343)

[6.x Inne 5](#_Toc52991344)

[7. Wymagania internacjonalizacji i lokalizacji 5](#_Toc52991345)

[8. Inne wymagania 5](#_Toc52991346)

[Suplement A: Słownictwo 5](#_Toc52991347)

[Suplement B: Modele analityczne 5](#_Toc52991348)

# 1. Wprowadzenie

## 1.1 Cel

Celem projektu jest stworzenie aplikacji webowej służącej do inwentaryzacji i ewidencji uposażenia domowego z podziałem na kategorie.

Dokument jest przeznaczony dla programistów i testerów.

## 1.2 Przyjęte konwencje

Opisz tu przyjęte standardy I konwencje, znaczenie formatowania.

# 1.3 Zakres projektu

Produkt ma na celu obsługę wszystkich aspektów inwentaryzacyjnych, z czego najważniejsze aspekty to:

* Modyfikacja, wyświetlanie, dodawanie i usuwanie przedmiotów inwentarza
* Zmiana możliwych typów i lokalizacji produktów inwentaryzowanych
* Wyszukiwanie
* Wyświetlanie powiadomień o brakujących i przeterminowanych produktach

## 1.4 Odwołania

# 2. Opis ogólny

## 2.1 Perspektywa produktu

Projekt zakłada utworzenie całkowicie nowego produktu.

## 2.2 Klasy i charakterystyki użytkowników

Użytkownik – korzysta z aplikacji.

## 2.3 Środowisko działania

Przeglądarki internetowe:

* Google Chrome
* Opera
* Edge
* Mozilla Firefox

## 2.4 Ograniczenia projektowe i wykonawcze

## 2.5 Założenia i zależności

Java + Spring

MySQL + MySQL Workbench 8.0

Git + Github

Docker

Vue + Vite

# 3. Funkcjonalności systemu

## 3.1: Konto użytkownika

### 3.1.1 Opis

Priorytet: wysoki

### 3.1.2 Wymagania funkcjonalne DO UZUPELNIENIA

1. Użytkownik musi mieć możliwość rejestracji w systemie.
2. Użytkownik musi mieć możliwość zalogowania się w systemie.
3. Użytkownik musi mieć wylogowania się z systemu.
4. Użytkownik musi mieć możliwość odzyskania konta po utracie hasła metodą potwierdzenia poprzez email.
5. Użytkownik musi mieć możliwość usunięcia konta.
6. Użytkownik musi mieć możliwość zmiany hasła.

## 3.2 Inwentaryzacja

### 3.2.1 Opis

Priorytet: wysoki

Ogół wymagań umożliwiających użytkownikowi zarządzanie inwentarzem.

### 3.2.2 Wymagania funkcjonalne

1. System musi umożliwić użytkownikowi wyświetlanie wszystkich przedmiotów.
2. System musi umożliwić użytkownikowi dodawanie przedmiotów.
3. System musi umożliwić użytkownikowi dodawanie i usuwanie kontenerów.
4. System musi umożliwić sortowanie przedmiotów po nazwie, kategorii, dacie ważności.
5. Użytkownik musi mieć możliwość utworzenia nowej kategorii przedmiotu.
6. Użytkownik musi mieć możliwość usuwania przedmiotu z inwentarza.
7. Użytkownik musi mieć możliwość wyszukiwania przedmiotów po atrybutach:
   1. Nazwa
   2. Kategoria
   3. Ilość
   4. Data ważności
   5. Kontener
   6. Ulubione
8. Użytkownik musi mieć możliwość wyświetlenia powiadomień o brakujących i przeterminowanych produktach.
9. Użytkownik musi mieć możliwość wprowadzenia limitu ilości danego przedmiotu od którego wyskakuje powiadomienie o niskiej ilości.
10. Użytkownik musi mieć możliwość dodać przedmiot do swoich ulubionych przedmiotów.

### 3.3.1 Kontenery

Priorytet: wysoki

Ogół wymagań umożliwiających użytkownikowi zarządzanie kontenerami.

### 3.3.2 Wymagania funkcjonalne

1. Kontener musi mieć możliwość być dodanym do innego kontenera.
2. Kontener można zamienić w przedmiot i odwrotnie.
3. Użytkownik musi mieć możliwość zmiany położenia przedmiotu między dwoma kontenerami.
4. Użytkownik musi mieć możliwość stworzenia kontenera.
5. Użytkownik musi mieć możliwość usunięcia kontenera z całą zawartością albo z automatycznym przeniesieniem przedmiotów do nadrzędnego kontenera.
6. Użytkownik musi mieć możliwość edytowania właściwości kontenera.

### 3.3.1 Role

Priorytet: wysoki

Ogół wymagań umożliwiających użytkownikowi zarządzanie rolami użytkownika.

Uprawnienia:

View: Można zobaczyć przedmioty, kontenery potomne, ilości i właściwości przedmiotów,

Edit: Można dodawać, usuwać i edytować zawartość w kontenerze, ich ilości, właściwości.

Full: Można dodawać, usuwać, edytować zawartość oraz przyznawać uprawnienia innym użytkownikom.

Owner: Można dodawać, usuwać, edytować zawartość oraz przyznawać i odbierać uprawnienia innym użytkownikom. Uprawnienie maksymalne, może wystąpić tylko jeden owner na kontener.

### 3.3.2 Wymagania funkcjonalne

1. System umożliwia nadanie użytkownikowi jednej roli przyznającej prawa dostępu do kontenera i jego zawartości.
2. System umożliwia nadanie użytkownikowi jednej roli przyznającej prawa dostępu do kontenera i jego zawartości.
3. System umożliwia odjęcie użytkownikowi roli przyznającej prawa dostępu do kontenera.

# 4. Wymagania dotyczące danych

## 4.1 Logiczny model danych

<https://online.visual-paradigm.com/share.jsp?id=323231313636372d31> link do ERD w visual paradigm online

## 4.2 Raporty

Można wydrukować listę wszystkich przedmiotów w kontenerze niebędących kontenerami.

## 4.3 Pozyskiwanie, integralność przechowywanie I usuwanie danych

Użytkownik może pobrać wszystkie swoje przedmioty lub ich część do pliku json.

Użytkownik może zaimportować swoje dane z pliku json, co skutkuje dodaniem przedmiotów i kontenerów do bazy danych.

# 5. Wymagania dotyczące interfejsu zewnętrznego

## 5.1 Interfejsy użytkownika

Opisz logiczną charakterystykę każdego interfejsu użytkownika, którego potrzebuje system. Niektóre specyficzne cechy interfejsów użytkownika mogą pojawić się w 6.1 Użyteczność. Należy zwrócić uwagę na:

* Odniesienia do standardów interfejsu użytkownika, których należy przestrzegać
* Standardy dotyczące czcionek, ikon, etykiet przycisków, obrazów, schematów kolorów, sekwencji zakładek w polach, powszechnie używanych elementów sterujących, grafiki marki, informacji o prawach autorskich i prywatności itp.
* Ograniczenia rozmiaru ekranu, układu lub rozdzielczości
* Standardowe przyciski, funkcje lub łącza nawigacyjne, które pojawią się na każdym ekranie, na przykład przycisk pomocy
* Skróty klawiszowe
* Wyświetlanie wiadomości i konwencje budowy / składni komunikatów
* Wytyczne dotyczące walidacji danych (takie jak ograniczenia wartości wejściowych i kiedy sprawdzać zawartość pól)
* Standardy układu ułatwiające lokalizację (tłumaczenie) oprogramowania
* Udogodnienia dla użytkowników niedowidzących, daltonistów lub mających inne ograniczenia

## 5.2 Interfejsy programowe

Aplikacja podzielona na backend i frontend komunikuje się pomiędzy tymi warstwami za pomocą fetch + json. Aplikacja łączy się z bazą danych mysql w celu przechowywania danych.

## 5.3 Interfejsy sprzętowe

Użytkownik porusza się po aplikacji używając klawiatury i myszki.

## 5.4 Interfejsy komunikacyjne

Gdy użytkownik zaloguje się ze swojego konta na inne urządzenie, nadal będzie widział te same dane w aplikacji.

# 6. Cechy jakości

## 6.1 Użyteczność

* Strona responsywna dostosowująca się do wymiarów wyświetlacza
* Współdzielenie kontenerów za pomocą uprawnień – jeden użytkownik może nadać uprawnienia drugiemu
* Możliwość wpisywania danych w formie szybkiej (w formie arkusza)

## 6.2 Wydajność

Określić wymagania dotyczące wydajności dla różnych operacji systemowych. Jeśli różne wymagania funkcjonalne lub cechy mają różne wymagania dotyczące wydajności, lepiej jest określić te cele wydajnościowe razem z odpowiednimi wymaganiami funkcjonalnymi, zamiast zbierać je w tej sekcji.

## 6.3 Zabezpieczenia

Określ wszelkie wymagania dotyczące kwestii bezpieczeństwa lub prywatności. Mogą one odnosić się do bezpieczeństwa fizycznego, danych lub oprogramowania.

* Weryfikacja użytkownika poprzez wysłanie czasowego tokena drogą mailową.
* Podczas logowania używa się jwt
* Użycie bezpiecznych szyfrów do szyfrowania danych wrażliwych
* Użycie architektury REST

## 6.4 Bezpieczeństwo użytkowania

Określ wymagania dotyczące możliwych szkód, które mogą wynikać z użytkowania produktu. Zdefiniuj wszelkie zabezpieczenia lub działania, które należy podjąć, a także potencjalnie niebezpieczne działania, którym należy zapobiec.

## 6.x Inne

* Po usunięciu kontenera uprawnienia wszystkich użytkowników do niego są kasowane.
* Po utworzeniu konta użytkownika należy utworzyć kontener zbiorczy na wszystkie kontenery tego użytkownika. Należy dodać dla tego użytkownika uprawnienie Owner do tego katalogu.

# 7. Wymagania internacjonalizacji i lokalizacji

Wymagania dotyczące umiędzynarodowienia i lokalizacji zapewniają, że produkt będzie odpowiedni do użytku w innych krajach, kulturach i lokalizacjach geograficznych niż te, w których został stworzony. Przykładowo różnice walutowe; formatowanie dat, numerów, adresów czy numer++ów telefonów. Wymagania dotyczące umiędzynarodowienia i lokalizacji mogą być ponownie wykorzystywane w różnych projektach.

Angielski, Polski

# 8. Inne wymagania

Opcjonalnie można zdefiniować dowolne inne wymagania nie zdefiniowane wcześniej w tym SRS. Przykładowo wymagania prawne, finansowe; wymagania instalacji produktu, jego konfiguracji, uruchomienia czy wyłączenia.

# Suplement A: Słownictwo

Należy zdefiniować tutaj wszelkie skróty i specjalistyczne słownictwo używane w tym dokumencie.

# Suplement B: Modele analityczne

W tej opcjonalnej sekcji należy umieścić wszelkie modele analityczne które pomogą czytelnikowi zrozumieć system. Przykładowo drzewa funkcjonalności, diagramy ERD, procesy BPMN. Czasem czytelniejsze może być umieszczenie tych diagramów bezpośrednio w treści SRSa.