Specyfikacja wymagań

Software Requirements Specification

# Wersja

| Wersja | Data | Autor | Zakres zmian |
| --- | --- | --- | --- |
| 0.1.0 | 2020-10-07 | Łukasz Łudwiczyński, Valeriia Balanivska, Jakub Kościołowski | Inicjalizacja szablonu dokumentu |

# Spis treści

[Wersja 1](#_gjdgxs)

[Spis treści 1](#_30j0zll)

[1. Wprowadzenie 2](#_1fob9te)

[1.1 Cel 2](#_3znysh7)

[1.2 Przyjęte konwencje 2](#_2et92p0)

[1.3 Zakres projektu 2](#_tyjcwt)

[1.4 Odwołania 2](#_3dy6vkm)

[2. Opis ogólny 2](#_1t3h5sf)

[2.1 Perspektywa produktu 2](#_4d34og8)

[2.2 Klasy i charakterystyki użytkowników 2](#_2s8eyo1)

[2.3 Środowisko działania 3](#_17dp8vu)

[2.4 Ograniczenia projektowe i wykonawcze 3](#_3rdcrjn)

[2.5 Założenia i zależności 3](#_26in1rg)

[3. Funkcjonalności systemu 3](#_lnxbz9)

[3.x Funkcjonalność XYZ 3](#_35nkun2)

[3.x.1 Opis 3](#_1ksv4uv)

[3.x.2 Wymagania funkcjonalne 3](#_44sinio)

[4. Wymagania dotyczące danych 3](#_2jxsxqh)

[4.1 Logiczny model danych 3](#_z337ya)

[4.2 Raporty 3](#_3j2qqm3)

[4.3 Pozyskiwanie, integralność przechowywanie I usuwanie danych 4](#_1y810tw)

[5. Wymagania dotyczące interfejsu zewnętrznego 4](#_4i7ojhp)

[5.1 Interfejsy użytkownika 4](#_2xcytpi)

[5.2 Interfejsy programowe 4](#_1ci93xb)

[5.3 Interfejsy sprzętowe 4](#_3whwml4)

[5.4 Interfejsy komunikacyjne 4](#_2bn6wsx)

[6. Cechy jakości 5](#_qsh70q)

[6.1 Użyteczność 5](#_3as4poj)

[6.2 Wydajność 5](#_1pxezwc)

[6.3 Zabezpieczenia 5](#_49x2ik5)

[6.4 Bezpieczeństwo użytkowania 5](#_2p2csry)

[6.x Inne 5](#_147n2zr)

[7. Wymagania internacjonalizacji i lokalizacji 5](#_3o7alnk)

[8. Inne wymagania 5](#_23ckvvd)

[Suplement A: Słownictwo 5](#_ihv636)

[Suplement B: Modele analityczne 5](#_32hioqz)

# 1. Wprowadzenie

Specyfikacja jest zbiorem wymagań, czyli zakresem funkcjonalności systemu. Wymagania funkcjonalne opisują funkcje użytkowników. Wymagania niefunkcjonalne cechy i ograniczenia systemu, wymagania procesowe. W dokumencie można poznać zakres projektu, opis ogólny, obszar, do którego system będzie dedykowany, cele systemu, funkcjonalności, wymagania dotyczące danych, wymagania dotyczące interfejsu zewnętrznego, cechy jakości.

Jednoznaczne wskazanie wszystkich użytkowników korzystających z systemu, jednoznaczne wskazanie wszystkich użytkowników, niezbędnych do działania systemu (obsługa, wprowadzanie danych, administracja). Określenie funkcji systemu udostępnianych użytkownikowi oraz sposobów korzystania z planowanego systemu. Określenie systemów zewnętrznych (obcych baz danych, sieci, Internetu), wykorzystywanych w czasie pracy systemu. Ustalenie struktur organizacyjnych, przepisów prawnych, statutów, zarządzeń, instrukcji, itd., które pośrednio lub bezpośrednio określają funkcje wykonywane przez planowany system.

## 1.1 Cel

Produkt jest prostą w obsłudzę aplikacją przeznaczoną dla pracowników, służącą do śledzenia czasu pracy, wyliczania należnego wynagrodzenia.

Dokument przeznaczony jest dla wszystkich osób zaangażowanych w realizację projektu i dla użytkowników.

## 1.2 Przyjęte konwencje

# FURPS: U - użyteczność, oznacza łatwość użytkowania systemu. Szkolenie pracownika nie jest potrzebne; korzystanie jest bardzo intuicyjne, liczba kontaktów ze wsparciem klienta jest prognozuj, liczbą sytuacji, w których konieczne jest skorzystanie z systemu pomocy;

# R - niezawodność, może być mierzona poprzez: średnią liczbę godzin pracy bez awarii (ang. MTBF - Mean Time Between Failure), maksymalną liczbą godzin w miesiącu w ciągu których system może być wyłączony w celach pielęgnacyjnych (ang. maintainance) - ma to znaczenie szczególnie w przypadku systemów, które muszą pracować na okrągło - np. systemy bankowości elektronicznej

# P - wydajność - mierzona np. liczbą transakcji, którą system jest w stanie obsłużyć w ciągu godziny, liczbą użytkowników, którzy mogą być zalogowani jednocześnie do portalu

# S - bezpieczeństwo, to wymagania związane z szyfrowaniem, polityką praw.

System musi spełniać wymagania rozporządzenia RODO.

# 1.3 Zakres projektu

Produkt jest prostą w obsłudzę aplikacją przeznaczoną dla pracowników, służącą do śledzenia czasu pracy. Podstawową funkcją aplikacji jest monitorowanie czasu zadań w projektach, dodanie klienta do projektu. Można mierzyć czas za pomocą stopera. Zgromadzone dane można filtrować po kliencie i dacie.

# 2. Opis ogólny

Oprogramowanie jest stroną w przeglądarce dla wyliczania czasu, logowania, historii, klientami, dodawanych do projektów, filtrowaniem dla pracowników. Ograniczone funkcje wychodzą z ograniczenia czasu, wyliczanego na zrealizowanie projektu.

## 2.1 Perspektywa produktu

Aplikacja nie jest produktem nowym na rynku i nie stanowi fragmentu większego systemu. W przyszłości istnieje możliwość rozbudowania aplikacji o kolejne segmenty zapewniające nowe funkcjonalności.

## 2.2 Klasy i charakterystyki użytkowników

1. Niezalogowany użytkownik - użytkownik jest niezarejestrowany w aplikacji.
2. Zalogowany użytkownik - użytkownik jest zarejestrowany w aplikacji.

## 2.3 Środowisko działania

Środowisko, w którym działać będzie oprogramowanie jest przeglądarka internetowa, do której dostęp będzie opierał się przez dowolny system operacyjny. Inne składniki oprogramowania lub aplikacje, z którymi system musi współpracować: MySQL, serwer, aplikacja WEB, HTML 5, JavaScript (ES5), PHP 8, CSS 2.1.

## 2.4 Ograniczenia projektowe i wykonawcze

Do realizacji projektu zostały wyznaczone odpowiednie technologie, w tym PHP, JS, HTML, CSS. Podstawą wyboru języka PHP jest - dynamiczność stron, integracja z bazą danych; HTML, CSS, JS - standardowe narzędzia do tworzenia stron.

Techniczne - system powinien być powiązany z bazą danych MySQL, w której będzie gromadzona informacja dotycząca użytkowników i adminów.

Protokoły komunikacyjne - HTTP, TCP.

Podstawowe wymogi bezpieczeństwa. System powinien być zabezpieczony przed niepowołanym dostępem.

Prawne - ustawa o ochronie danych osobowych, ustawa o prawach konsumenta. System musi spełniać wymagania rozporządzenia RODO.

Ograniczenia czasowe.

## 2.5 Założenia i zależności

Wymagania funkcjonalne opisują funkcje użytkowników: logowanie się; śledzenie czasu pracy i historia. Zarządzanie współbieżnym dostępem - obsługa wielu użytkowników w taki sposób aby każdy z nich był niewidoczny dla innych użytkowników. Dodanie klienta do projektu. Zgromadzone dane można filtrować po kliencie i dacie.

Zależności budowanego produktu od czynników zewnętrznych lub komponentów pozostających poza jego kontrolą: czynniki ekonomiczne, prawne, społeczne, technologiczne, organizacyjne, związane z procesem decyzyjnym. Naturalne (katastrofy).

# 3. Funkcjonalności systemu

## 3.1 Funkcjonalność A

1. Uruchomienie programu, rejestracja użytkownika

### 3.1.1 Opis

Warunek początkowy: uruchomienie programu, powoduje to zaistnienie obiektu System. Funkcjonalność służy do dodania do bazy danych i korzystania z systemu. Częstotliwość – Bardzo Wysoka, istotność – Bardzo Wysoka.

### 3.1.2 Wymagania funkcjonalne

Potrzebne są baza danych i interfejs aplikacji (HTML, CSS).

1. Wybór przycisku “Rejestracja”.

2. Wyświetlenie formularzu rejestracji zawierający pola z danymi do wypełnienia przez użytkownika.

3. Wypełnienie formularzu i zatwierdzenie użytkownikiem.

3.1. Komunikat o potwierdzeniu. Użytkownik zostaje zarejestrowany w aplikacji. Wyświetlenie strony logowania się.

3.2. Użytkownik nie zostaje zarejestrowany w systemie.

## 3.2 Funkcjonalność B

1. Logowanie się.

### 3.2.1 Opis

Niezbędna funkcja do korzystania z systemu. Warunek początkowy: uruchomiony program, użytkownik zarejestrowany. Częstotliwość – Bardzo Wysoka, istotność – Bardzo Wysoka.

### 3.2.2 Wymagania funkcjonalne

Potrzebne są baza danych i interfejs aplikacji (HTML, CSS).

1. Wyświetlone okno logowania.

2. Wprowadzenie danych.

3.1. Logowanie poprawne - wyświetla się Główne okno programu.

3.2. Logowanie niepoprawne - wyświetla się komunikat o błędzie logowania.

3.2.1. Przycisk OK - powrót do okna logowania.

## 3.3 Funkcjonalność C

1. Reset hasła

### 3.3.1 Opis

Zmiana hasła w przypadku zapomnienia. Warunek początkowy: uruchomiony program, użytkownik zarejestrowany. Częstotliwość – Średnia, istotność – Bardzo Wysoka.

### 3.3.2 Wymagania funkcjonalne

1. Wybranie opcji resetowania hasła przez użytkownika przy próbie logowania.

2. System wyświetla formularz resetowania hasła zawierający pole: E-mail (\*pole obowiązkowe)

3. Użytkownik wypełnia formularz i zatwierdza.

4. System sprawdza, czy adres email wprowadzony przez użytkownika jest poprawny oraz czy istnieje w bazie danych.

4.1. System wysyła nowe, wygenerowane hasło na wskazany przez użytkownika adres e-mail. System wyświetla komunikat powodzenia. Użytkownik zostaje przekierowany na stronę logowania.

4.2. Użytkownik nie zostanie zalogowany do systemu z powodu braku zapisu.

## 3.4 Funkcjonalność D

1. Śledzenie czasu pracy

### 3.4.1 Opis

Warunek początkowy: uruchomienie programu, użytkownik zarejestrowany i zalogowany. Częstotliwość – Bardzo Wysoka, istotność – Bardzo Wysoka.

### 3.4.2 Wymagania funkcjonalne

Funkcjonalność w JavaScript, HTML, CSS.

1. Naciśnięcie przycisku “Start”.

2. Wybór “Początek pracy” i wyznaczenie daty i godziny początku pracy.

3. Wybór “Koniec pracy” i wyznaczenie daty i godziny końca pracy.

4. Wybór przycisku “Zapisz”.

4.1. Zapisanie poprawne.

4.2. Zapisanie niepoprawne.

## 3.5 Funkcjonalność E

1. Dodanie klienta

### 3.5.1 Opis

Warunek początkowy: uruchomienie programu, użytkownik zarejestrowany i zalogowany. Częstotliwość – BW, istotność – BW.

### 3.5.2 Wymagania funkcjonalne

Funkcjonalność w JavaScript, HTML, CSS.

1. Naciśnij przycisk “+ Klient”

2. Wpisz nazwę/imię.

3. Naciśnij OK.

4. Zapisanie się powiodło.

## 3.6 Funkcjonalność F

F. Historia śledzeń czasu pracy

### 3.6.1 Opis

Warunek początkowy: uruchomienie programu, użytkownik zalogowany, wpisy. Częstotliwość – BW, istotność – BW.

### 3.6.2 Wymagania funkcjonalne

Funkcjonalność w JavaScript, HTML, CSS.

1. Zapisanie czasu poprawne.

2. Wyświetlanie zapisu czasu (data, czas, Klient).

## 3.5 Funkcjonalność G

G. Filtrowanie po dacie i kliencie

### 3.5.1 Opis

Warunek początkowy: uruchomienie programu, użytkownik zalogowany, wpisy z dodanym Klientem. Częstotliwość – BW, istotność – BW.

### 3.5.2 Wymagania funkcjonalne

Funkcjonalność w JavaScript, HTML, CSS.

1. Naciśnij przycisk “Filtruj”.

2. Wybierz “Klient”.

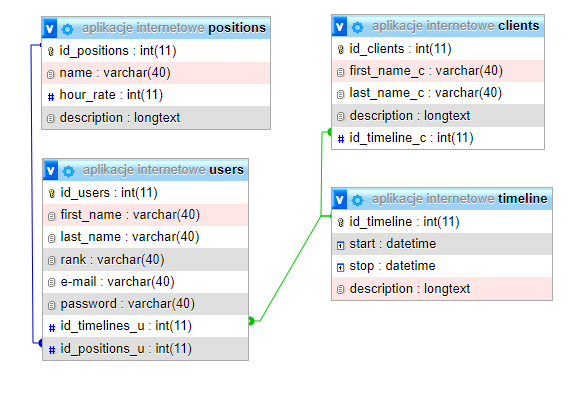
3. Wpisz nazwę (wybierz z wypadającej listy).

4. Wyświetlanie zapisów po Kliencie.

# 4. Wymagania dotyczące danych. Logiczny model danych

Diagram ERD.

## 4.1 Logiczny model danych



## 4.3 Pozyskiwanie, integralność przechowywanie i usuwanie danych

Wszystkie dane będą znajdować się w bazie danych MYSQL. W przypadku zarządzania kopiami danych, będą one znajdować się na dyskach zewnętrznych, a proces samego zabezpieczania tych danych będzie wykonywany według harmonogramu odgórnie ustalonego. Zaś każda walidacja w przypadku nadpisania poprzedniego stanu rozwiąże integralność.

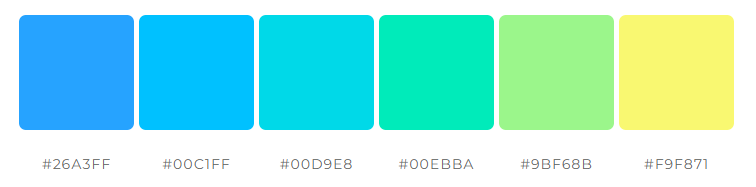
# 5. Wymagania dotyczące interfejsu zewnętrznego

System powinien być responsywny i przygotowany do wyświetlania zarówno na dużych monitorach jak i smartfonach.

## 5.1 Interfejsy użytkownika

Interfejs użytkownika będzie przedstawiony w sposób minimalistyczny. Responsywności wobec urządzeń o rozbieżnej rozdzielczości będzie zachowana. Sam styl będzie w formie Atomic Design.

Interfejs będzie składał się z następujących palet kolorów:



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

Opisz połączenia między tym produktem a innymi składnikami oprogramowania (zidentyfikowanymi według nazwy i wersji), w tym innymi aplikacjami, bazami danych, systemami operacyjnymi, narzędziami, bibliotekami, witrynami internetowymi i zintegrowanymi komponentami komercyjnymi.

Określić cel, formaty i treść komunikatów, danych i wartości kontrolnych wymienianych między komponentami oprogramowania.

Określ odwzorowania danych wejściowych i wyjściowych między systemami oraz wszelkie tłumaczenia, które muszą zostać wykonane, aby dane mogły zostać przesłane z jednego systemu do drugiego.

## 5.4 Interfejsy komunikacyjne

Określić wymagania dotyczące wszelkich funkcji komunikacyjnych, których będzie używać produkt, w tym poczty e-mail, przeglądarki internetowej, protokołów sieciowych i formularzy elektronicznych.

Zdefiniować odpowiednie formaty komunikatów. Określ zasady bezpieczeństwa i szyfrowania komunikacji, szybkości przesyłania danych, mechanizmy synchronizacji.

# 6. Cechy jakości

W tej sekcji określa się wymagania niefunkcjonalne inne niż ograniczenia, które są zapisane w sekcji 2.4 oraz 5.1.

Te wymagania jakościowe powinny być szczegółowe, ilościowe i weryfikowalne.

Należy wskazać względne priorytety różnych atrybutów, np. bezpieczeństwo ponad wydajnością.

## 6.1 Użyteczność

System jest prosty w obsłudze dla użytkowników. Aplikacja posiada bardzo czytelny interfejs oraz sposób nawigacji w systemie. Każda z funkcjonalności występuje w osobnej zakładce. Użyte czcionki oraz kolory są adekwatne do tematyki aplikacji i wpływają na komfort jej użytkowania.

## 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.

## 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

Utwórz osobną sekcję w SRS dla każdego dodatkowego atrybutu jakości produktu, aby opisać cechy, które będą ważne zarówno dla klientów, jak i dla programistów.

# 7. Wymagania internacjonalizacji i lokalizacji

Oprogramowanie nie wspiera innych regionów niż Polska.

# Suplement A: Słownictwo

**Pracownik**  - osoba, która świadczy swoje usługi osobiście i realizuje projekty na zlecenie.

**Klient** - osoba lub organizacja, która jest przypisana do pracownika.