Specyfikacja wymagań

Software Requirements Specification

# Wersja

|  |  |  |  |
| --- | --- | --- | --- |
| Wersja | Data | Autor | Zakres zmian |
| 1.0 | 2022-01-27 | Grupa projektowa Marek | Inicjalizacja szablonu dokumentu |

# Spis treści

[Wersja 1](#_Toc94026353)

[Spis treści 1](#_Toc94026354)

[1. Wprowadzenie 2](#_Toc94026355)

[1.1 Cel 2](#_Toc94026356)

[1.2 Przyjęte konwencje 2](#_Toc94026357)

[1.3 Zakres projektu 3](#_Toc94026358)

[2. Opis ogólny 3](#_Toc94026359)

[2.1 Perspektywa produktu 3](#_Toc94026360)

[2.2 Klasy i charakterystyki użytkowników 3](#_Toc94026361)

[2.3 Środowisko działania 4](#_Toc94026362)

[2.4 Ograniczenia projektowe i wykonawcze 4](#_Toc94026363)

[2.5 Założenia i zależności 4](#_Toc94026364)

[3. Funkcjonalności systemu 4](#_Toc94026365)

[3.1 Strona one-page z opisem systemu dostępna dla wszystkich użytkowników. 5](#_Toc94026366)

[3.1.1 Opis 5](#_Toc94026367)

[3.1.2 Wymagania funkcjonalne 5](#_Toc94026368)

[3.2 Tworzenie użytkowników 5](#_Toc94026369)

[3.2.1 Opis 5](#_Toc94026370)

[3.2.2 Wymagania funkcjonalne 5](#_Toc94026371)

[3.3 Śledzenie czasu 5](#_Toc94026372)

[3.3.1 Opis 5](#_Toc94026373)

[3.3.2 Wymagania funkcjonalne 5](#_Toc94026374)

[3.4 Tworzenie wpisów 5](#_Toc94026375)

[3.4.1 Opis 5](#_Toc94026376)

[3.4.2 Wymagania funkcjonalne 5](#_Toc94026377)

[3.5 Tworzenie projektów 6](#_Toc94026378)

[3.5.1 Opis 6](#_Toc94026379)

[3.5.2 Wymagania funkcjonalne 6](#_Toc94026380)

[3.6 Tworzenie klientów 6](#_Toc94026381)

[3.6.1 Opis 6](#_Toc94026382)

[3.6.2 Wymagania funkcjonalne 6](#_Toc94026383)

[3.7 Utworzenie grupy i zarządzanie nią 6](#_Toc94026384)

[3.7.1 Opis 6](#_Toc94026385)

[3.7.2 Wymagania funkcjonalne 6](#_Toc94026386)

[3.8 Generowanie raportów 6](#_Toc94026387)

[3.8.1 Opis 6](#_Toc94026388)

[3.8.2 Wymagania funkcjonalne 7](#_Toc94026389)

[4. Wymagania dotyczące danych 7](#_Toc94026390)

[4.1 Logiczny model danych 7](#_Toc94026391)

[4.2 Raporty 7](#_Toc94026392)

[4.3 Pozyskiwanie, integralność przechowywanie I usuwanie danych 8](#_Toc94026393)

[5. Wymagania dotyczące interfejsu zewnętrznego 8](#_Toc94026394)

[5.1 Interfejsy użytkownika 8](#_Toc94026395)

[5.2 Interfejsy programowe 8](#_Toc94026396)

[5.3 Interfejsy sprzętowe 9](#_Toc94026397)

[5.4 Interfejsy komunikacyjne 9](#_Toc94026398)

[6. Cechy jakości 9](#_Toc94026399)

[6.1 Użyteczność 9](#_Toc94026400)

[6.2 Wydajność 9](#_Toc94026401)

[6.3 Zabezpieczenia 9](#_Toc94026402)

[6.4 Bezpieczeństwo użytkowania 9](#_Toc94026403)

[7. Wymagania internacjonalizacji i lokalizacji 10](#_Toc94026404)

[Suplement A: Słownictwo 10](#_Toc94026405)

# 1. Wprowadzenie

## 1.1 Cel

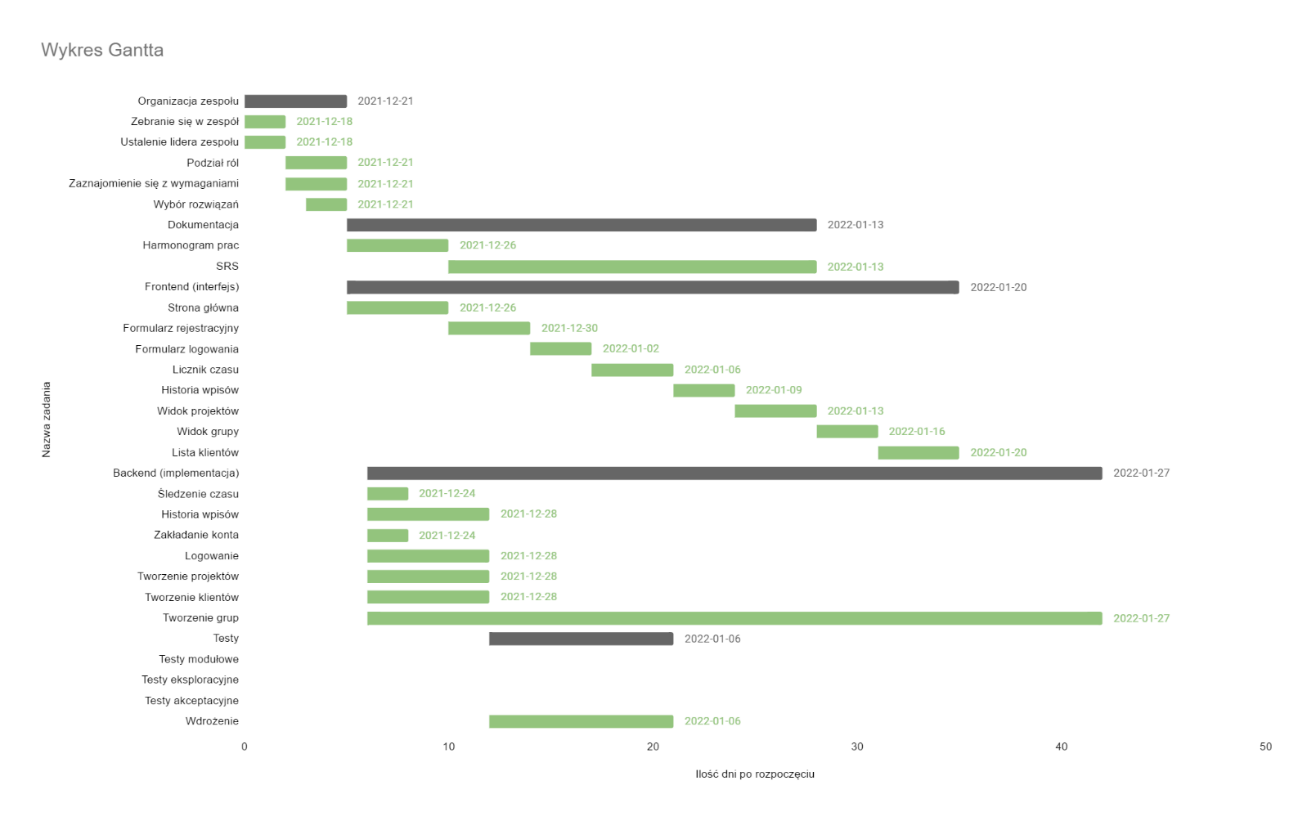
ClockerTracker jest to aplikacja internetowa, która ma służyć do zarządzania czasem poświęcanym podczas tworzenia projektów. Udostępnione statystki oraz grupy użytkowników mają pomóc również w kontrolowaniu przebiegu czynności, które wykonuje użytkownik. Program jest przeznaczony dla osób, które chcą lepiej rozporządzać swoim czasem od kierowników projektów do studentów wykonujących zadnia w ramach uczelni.

## 1.2 Przyjęte konwencje

Według przyjętych założeń, projekt powinien posiadać poniższe zastosowania:

* Paradygmatu obiektowego
* Mapowania obiektowego-relacyjnego
* Zastosowanie front-controllera
* Model MVC
* Responsywny wygląd systemu

# 1.3 Zakres projektu



Na powyższym wykresie, zaprezentowany jest zakres projektu i jego funkcjonalności w porównaniu do przypuszczalnego czasu poświęconego na projekt.

# 2. Opis ogólny

## 2.1 Perspektywa produktu

ClockerTracker to nowy produkt konkurencyjny na rynku. Został zaprojektowany z myślą o studentach, którzy mają kłopoty z zarządzaniem czasem. Aplikacja wymaga rejestracji w celu pełnego korzystania z funkcjonalności.

## 2.2 Klasy i charakterystyki użytkowników

|  |  |
| --- | --- |
| **Tabela klas użytkowników** | |
| **Klasa użytkownika** | **Opis** |
| Użytkownik | Osoba, która posiada funkcje tworzenia postu, mierzenia czasu, tworzenie grupy, tworzenia projektu i dodanie klienta. |
| Admin | Posiada wszystkie funkcje użytkownika, ale może również zarządzać utworzoną grupą. |

## 2.3 Środowisko działania

Głównym środowiskiem w jakim działa użytkownik jest przeglądarka internetowa, która zawiera dostęp do Internetu. Za jej pomocą aplikacja jest w stanie się uruchomić, połączyć z serwerem oraz bazą danych. Strona nie posiada ograniczeń dotyczących rodzaju systemu czy też kategorii przeglądarki, jest dopasowana do każdego rodzaju tego rozwiązania. Najlepsze połączenie osiągnie użytkownik, który znajduje się w UE ze względu na znajdowanie się serwerów domyślnych w Polsce, jednakże nie powoduje to braku dostępu do strony dla osób z innych kontynentów czy też regionów świata.

## 2.4 Ograniczenia projektowe i wykonawcze

Ze względu na to, że projekt jest tworzony w ramach kursu Aplikacje Internetowe, zostały nałożone ograniczenia projektowe w postaci języka programowania za pomocą, którego mają zostać zaimplementowane funkcjonalności. Jest to język PHP. W ramach tego założenia została podjęta decyzja o używaniu bazy danych phpmyadmin. W celu stworzenia graficznego interfejsu użytkownika został użyty język CSS w technologii REACT.

## 2.5 Założenia i zależności

Projekt jest tak skonstruowany, że nie potrzebna jest dodatkowa konfiguracja.

# 3. Funkcjonalności systemu

**Część publiczna**

* Dostęp bez logowania.
* Strona one-page z opisem systemu dostępna dla wszystkich użytkowników.
* Bieżące statystyki liczby użytkowników, suma zaraportowanych godzin w tygodniu, miesiącu, roku, w całości

**Część zamknięta**

* Dostęp tylko dla użytkowników zalogowanych.
* Role w systemie:
  + §  administrator
  + §  użytkownik
* Możliwość utworzenia wielu użytkowników w każdej roli
* Możliwość śledzenia czasu: start, stop, opis, powiązany projekt, powiązany z projektem klient.
* Możliwość edycji wpisów.
* Historia wpisów.
* Możliwość zarządzania użytkownikami (ADMIN).
* Możliwość tworzenia grup użytkowników. Jeden użytkownik może należeć do wielu grup.
* Możliwość tworzenia projektów i przypisywania ich do rekordów historii czasu.
* Możliwość tworzenia klientów i przypisywanie projektów do nich.
* Generowanie raportów CSV za zadany okres / użytkownika / projekt / klienta: szczegółowy: lista wpisów

## 3.1 Strona one-page z opisem systemu dostępna dla wszystkich użytkowników.

### 3.1.1 Opis

Funkcjonalność polega na dostępnie do wglądu strony z opisem aplikacji dla użytkowników niezarejestrowanych. Priorytet niski.

### 3.1.2 Wymagania funkcjonalne

* Strona umieszczona na domenie.

## 3.2 Tworzenie użytkowników

### 3.2.1 Opis

System pozwala na utworzenie konta za pomocą formularza rejestracji.

Priorytet funkcjonalności: Wysoki

### 3.2.2 Wymagania funkcjonalne

* Strona umieszczona na domenie.
* Aplikacja połączona z bazą
* Formularz

## 3.3 Śledzenie czasu

### 3.3.1 Opis

Funkcjonalność umożliwia pomiar czasu. Dostępne są takie możliwości jak:

Start pomiaru, Koniec pomiaru, Opis mierzonego procesu, powiązany z procesem projekt i klient.

Priorytet funkcjonalności: Wysoki

### 3.3.2 Wymagania funkcjonalne

* Użytkownik musi posiadać konto i być zalogowany.

## 3.4 Tworzenie wpisów

### 3.4.1 Opis

Użytkownik posiada możliwość dodania wpisów na swój profil. Polegają one zamieszczenie opisu mierzonego zdarzenia.

Priorytet funkcjonalności: Średni

### 3.4.2 Wymagania funkcjonalne

* Użytkownik musi posiadać konto i być zalogowany.

## 3.5 Tworzenie projektów

### 3.5.1 Opis

Funkcjonalność polega na stworzeniu projektu do którego można przypisać mierzone działania.

Priorytet funkcjonalności: Średni

### 3.5.2 Wymagania funkcjonalne

* Użytkownik musi posiadać konto i być zalogowany.

## 3.6 Tworzenie klientów

### 3.6.1 Opis

Funkcjonalność polega na utworzeniu klienta dla, którego tworzony jest dany projekt, czy też dla którego są tworzone dane czynności wcześniej zmierzone poprzez system.

Priorytet funkcjonalności: Średni

### 3.6.2 Wymagania funkcjonalne

* Użytkownik musi posiadać konto i być zalogowany.

## 3.7 Utworzenie grupy i zarządzanie nią

### 3.7.1 Opis

Funkcjonalność ta opiera się na utworzeniu grup użytkowników w celu lepszego zarządzania projektem. Każdy użytkownik może utworzyć grupę składającą się z danych użytkowników oraz zostać adminem danej grupy w celu zarządzania nią.

Priorytet funkcjonalności: Średni

### 3.7.2 Wymagania funkcjonalne

* Użytkownik musi posiadać konto i być zalogowany.

## 3.8 Generowanie raportów

### 3.8.1 Opis

Funkcjonalność ta opiera się na możliwości generowania raportów CSV na podstawie zmierzonych czasów.

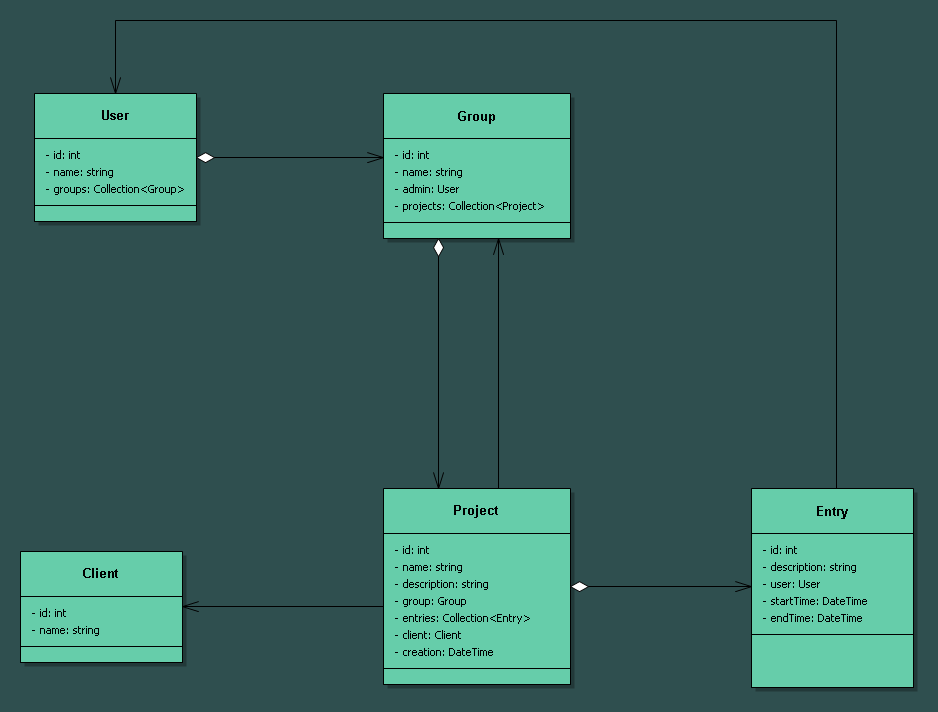
Priorytet funkcjonalności: Niski

### 3.8.2 Wymagania funkcjonalne

* Użytkownik musi posiadać konto i być zalogowany.
* Użytkownik musi posiadać zmierzone wartości

# 4. Wymagania dotyczące danych

## 4.1 Logiczny model danych



Powyższy diagram ukazuje model bazy danych, który został zaimplementowany w ramach projektu. Na jego podstawie można stwierdzić, że dostępność do danej grupy danych jest ustalona i nie przypadkowa.

## 4.2 Raporty

Generowane raporty będą posiadały dane takie jak:

* Nazwa użytkownika
* Ilość przeprowadzonych pomiarów
* Utworzone projekty
* Czas pomiarów
* Podsumowanie

Raporty będą generowane w formacie CSV.

## 4.3 Pozyskiwanie, integralność przechowywanie I usuwanie danych

Zebrane dane takie jak adres email użytkownika czy hasło zostaje przekazane do bazy danych, która jest utworzona na serwerze zut.edu.pl. Całość przekazanych danych jest przetrzymywana przez deweloperów zgodnie z określonym regulaminem, na okres dwóch lat od utworzenia konta. Korzystanie z serwerów uczelnianych pozwala na bezpieczne przechowywanie każdego rodzaju informacji ze względu na możliwość stworzenia kopii zapasowej danych oraz na rodzaj kontrolowania klastrów pamięci.

# 5. Wymagania dotyczące interfejsu zewnętrznego

## 5.1 Interfejsy użytkownika

|  |  |
| --- | --- |
| **Wymagania odnośnie interfejsu** | **Stan aplikacji** |
| Standard interfejsu użytkownika, których należy przestrzegać | Interfejs musi spełniać funkcjonalności zawarte w aplikacji oraz wpisywać się w zasadę maksymalnie trzech kolorów projektu strony. |
| Standardy dotyczące ikon | Wszystkie ikony muszą posiadać licencję dzięki, której będzie można ich używać w projekcie. <https://materialdesignicons.com/> |
| Ograniczenia rozmiaru ekranu, układu lub rozdzielczości | Projekt musi być dopasowany do ekranów monitorów komputera, nie przewiduje dopasowania do telefonów. |
| Standardowe przyciski, funkcje lub łącza nawigacyjne, które pojawią się na każdym ekranie, na przykład przycisk pomocy | Wszystkie przyciski musza być zaimplementowane na podstawie funkcjonalności. |
| Skróty klawiszowe | Aplikacja nie przewiduje konkretnych użyć skrótów klawiszowych |
| Wyświetlanie wiadomości i konwencje budowy / składni komunikatów | Komunikaty są oparte na czerwonych podświetleniach tekstu lub na powiadomieniach okienkowych. |
| Wytyczne dotyczące walidacji danych (takie jak ograniczenia wartości wejściowych i kiedy sprawdzać zawartość pól) | W przypadku rejestracji czy logowania przewidziana jest walidacja wpisywanych haseł. |
| Standardy układu ułatwiające lokalizację (tłumaczenie) oprogramowania | Brak |
| Udogodnienia dla użytkowników niedowidzących, daltonistów lub mających inne ograniczenia | Strona jest możliwa do odczytania screen readerem. |

## 5.2 Interfejsy programowe

Do komunikacja z bazą danych używa się interfejsu mapowania obiektów do encji. Pozwala to na łatwe zarzadzanie niezależne od systemu bazy danych. Tłumaczenie jest wykonywane przez Doctrine Entity Manager, dzięki czemu programista nie musi się przejmować tym procesem, ponieważ jest on wykonywany za niego.

## 5.3 Interfejsy sprzętowe

Interfejs sprzętowy jaki jest potrzebny to serwer. Po stronie użytkownika jest to monitor wraz z komputerem z dostępem do Internetu.

## 5.4 Interfejsy komunikacyjne

Projekt korzysta z poczty, która jest w domenie uczelni Wydziału Informatyki. Powoduje to, że nie jest konieczne przypisanie specjalnie dedykowanego emailu do serwera.

# 6. Cechy jakości

## 6.1 Użyteczność

Interfejs użytkowniaka jest łatwy i przystępny. Dobrana paleta kolorów ułatwia odczytanie wartości, a sposób nawigacji jest określony w taki sposób, aby użytkownik w prosty sposób potrafił odnaleźć wszystkie możliwe funkcjonalności. Strona jest przejrzysta.

## 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.

|  |  |
| --- | --- |
| **Operacja** | **Czas odpowiedzi na uruchomienie** |
| Logowanie | 4s |
| Utworzenie projektu | 2s |
| Kliknięcie przycisku Start do mierzenia czasu | 1s |
| Przejście między zakładką Grupy->Klienci | 2s |

## 6.3 Zabezpieczenia

Pierwszym elementem działającym na zabezpieczenie projektu jest umieszczenie aplikacji na zabezpieczonym serwerze Zachodniopomorskiego Uniwersytetu Technologicznego na Wydziale Informatyki. Są to specjalne dedykowane serwery dla studentów, które są zarządzane przez specjalistów odnośnie administrowania sieciami.

W przypadku aplikacji dodatkowe zabezpieczenia są zaimplementowane za pomocą używania haseł z domieszką soli i ich odszyfrowywania ich.

## 6.4 Bezpieczeństwo użytkowania

Użytkownik jest narażony na niebezpieczeństwo poprzez zmęczenie wzroku w przypadku długiego używania aplikacji (8h).

# 7. Wymagania internacjonalizacji i lokalizacji

Strona nie jest stworzona dla użytkowników z poza Polski. W planach jest stworzenie tłumaczenia do interfejsu graficznego.

# Suplement A: Słownictwo

**Encja** – reprezentacja wyobrażonego lub rzeczywistego obiektu (grupy obiektów) stosowana przy modelowaniu danych podczas analizy informatycznej.

**Baza** **danych** – zbiór danych zapisanych zgodnie z określonymi regułami.

**Php** – interpretowany, skryptowy język programowania zaprojektowany do generowania stron internetowych i budowania aplikacji webowych w czasie rzeczywistym.

**React.js** - biblioteka języka programowania JavaScript, która wykorzystywana jest do tworzenia interfejsów graficznych aplikacji internetowych.