# ${\bf Projekt\ zaliczeniowy-Time sheet}$

Dokumentacja aplikacji do zarządzania zadaniami

 ${\rm J.W.,\ O.Z.,\ M.B.,\ R.B.,\ M.S.}$ 

Uniwersytet Śląski 2017

SPIS TREŚCI Timesheet

# Spis treści

1	Zał	ożenia projektowe	3
	1.1	Przeznaczenie aplikacji	3
	1.2	Elementy składowe	3
<b>2</b>	$\mathbf{Spe}$	ecyfikacja techniczna	5
	2.1	Język programowania i frameworki	5
		2.1.1 PHP oraz Laravel	5
		2.1.2 Bootstrap	5
	2.2	Baza danych	5
	2.3	Repozytorium kodu	5
	2.4	Zarządzanie pracą w zespole	6
	2.5	Uruchomienie aplikacji	6
3	Rea	alizacja	6
	3.1	Przykładowe widoki	7
		3.1.1 Główny ekran aplikacji	7
		3.1.2 Tworzenie nowego projektu	7
		3.1.3 Coś jeszcze (?)	7
	3.2	Mechanizm logowania użytkownika	7
	3.3	Mechanizm rejestracji użytkownika	7
4	Tes	towanie	7
	4.1	Metody testowania	7
	4.2	Jakieś wnioski?	7

# 1 Założenia projektowe

### 1.1 Przeznaczenie aplikacji

Celem niniejszego projektu jest stworzenie aplikacji webowej, pozwalającej na zarządzanie zadaniami podczas pracy w zespole. Umożliwiać ona będzie tworzenie wielu projektów, a w ich ramach zadań, do realizacji których przydzielać będzie można konkretnych użytkowników. Każdy projekt posiadać będzie menadżera, który ma większe uprawnienia, niż pozostali członkowie zespołu. Administrator aplikacji posiada maksymalne uprawnienia i może zarządzać również menadżerami.

### 1.2 Elementy składowe

Zgodnie z przedstawionymi wymaganiami, tworzona aplikacja składać się będzie z następujących elementów (cech):

#### • Ogólne:

- Witryna internetowa,
- Należy zastosować ORM lub Micro-ORM,
- Dane powinny być walidowane po stronie front i back endu,
- Mechanizm logowania błędów,
- Projektu musi się znajdować na repozytorium,
- Dane muszą być zapisywane i odczytywane z bazy danych,
- Projekt będzie oceniany również ze względu na estetykę,
- Projekt musi posiadać czytelną dokumentację,

### • Aplikacja:

- Rejestracje nowego użytkownika wraz z podaniem danych osobowych,
- Mechanizm "Przypominania hasła",

- W systemie muszą występować 3 role Administrator, Manager i Pracownik,
- Moduł przesyłania wiadomości pomiędzy użytkownikami systemu,
- We wszystkich tabelach w systemie powinna być możliwość filtracji danych,
- Należy zaimplementować mechanizm aprobowania wpisów.

#### • Pracownik:

- Możliwość dodawania i edycji wpisów za danych dzień pracy. Wpis musi zawierać: datę, czas pracy, projekt, zadanie, komentarz,
- Wyświetlanie listy wszystkich wpisów (wraz z możliwością ich filtracji).

#### • Manager:

- Wyświetlanie, dodawanie, edycja i usuwanie zadań (zadania powinny posiadać przewidywaną estymację),
- Wyświetlanie wszystkich projektów, które są prowadzone przez danego managera,
- Wyświetlanie listy pracowników w danym projekcie,
- Generowanie raportu z przebiegu prac w projekcie,
- Aprobowanie wpisów pracowników.

#### • Administrator:

- Wyświetlanie listy użytkowników, zadań i projektów,
- Możliwość zablokowania użytkownika lub wymuszenie zmiany hasła,
- Dodawanie, edycja i usuwanie projektów,
- Przypisywanie managerów i pracowników do projektów.

# 2 Specyfikacja techniczna

W tym rozdziale opisane zostaną techniczne aspekty wykonania projektu, w tym zastosowane języki programowania, biblioteki oraz technologie. Zostanie również wyjaśniona kwestia zarządzania pracą w zespole, opisane repozytorium kodu oraz struktura bazy danych.

### 2.1 Język programowania i frameworki

#### 2.1.1 PHP oraz Laravel

Aplikacja stworzona została w oparciu o skryptowy, interpretowany o język PHP. Od początku stosowano także framework Laravel w wersji 5.4 rozszerzający możliwości języka między innymi o modułowość. Framework ten rozwijany jest w marach licencji MIT. Jego kod źródłowy pozostaje dostępny w serwisie GitHub, pod adresem: https://github.com/laravel. Laravel uznawany jest za jeden z najpopularniejszych frameworków języka PHP. Szacuje się, iż wykorzystuje go obecnie kilkadziesiąt milionów aplikacji webowych.

### 2.1.2 Bootstrap

Butsztrap

## 2.2 Baza danych

Jakaż to baza i jej struktura

# 2.3 Repozytorium kodu

Aby zapewnić najwyższą wygodę pracy w zespole oraz bezpieczeństwo tworzonego kodu aplikacji, pliki projektu umieszczone zostały w repozytorium w serwisie GitHub, pod adresem: https://github.com/Rys922/Timesheet. Znajduje się tam najnowsza wersja kodu, która na bieżąco aktualizowana jest przez członków zespołu, wraz z postępem w pracach.

3 REALIZACJA Timesheet

Wykorzystanie rozproszonego systemu kontroli wersji, jakim jest Git, gwarantuje dostępność kodu dla wszystkich członków zespołu i umożliwia jednoczesną pracę na plikach (aktualna wersja kodu zapisana jest zarówno lokalnie na komputerze programisty, jak i na głównym serwerze usługi).

### 2.4 Zarządzanie pracą w zespole

Do rozdzielania prac pomiędzy członków zespołu tworzącego niniejszy projekt, wykorzystano serwis *Trelloo* (https://trello.com). Jest to aplikacja pozwalająca dzielić zadania na grupy, przypisywać im osoby oraz wyznaczać terminy ich realizacji. Dzięki możliwości przenoszenia zrealizowanych zadań do innych grup, łatwo śledzić bieżący postęp w pracach. Pozwala to również w razie błędów cofnąć wybrany etap z fazy gotowości na przykład do fazy dalszych testów lub ponownej implementacji.

### 2.5 Uruchomienie aplikacji

Baza danych Baza chodzi sobie tu i tam (adres i w ogóle)

Witryna XAMMP-u XAMMP-u

# 3 Realizacja

Widoki Był sobie główny widok, który później się rozszerza

4 TESTOWANIE Timesheet

# 3.1 Przykładowe widoki

- 3.1.1 Główny ekran aplikacji
- 3.1.2 Tworzenie nowego projektu
- 3.1.3 Coś jeszcze (?)
- 3.2 Mechanizm logowania użytkownika
- 3.3 Mechanizm rejestracji użytkownika
- 4 Testowanie
- 4.1 Metody testowania
- 4.2 Jakieś wnioski?