**Rzeszów, 07.06.2018 r.**

**Wydział: Matematyczno-Przyrodniczy  
Studia: Stacjonarne  
Kierunek: Informatyka  
Rok: III**

**Członkowie zespołu:**

**- Piotr Markowski (Team Lider)  
- Adrian Wilk  
- Tomasz Wysocki  
- Michał Śliwa**

**Przedmiot:   
Programowanie Zespołowe**

Dokumentacja Projektu

**„Aplikacja do zarządzania zadaniami**”

Prowadzący:

Mgr. Inż. Adam Szczur

# Spis treści

[Spis treści 2](#_Toc510689798)

[Cel Projektu 3](#_Toc510689799)

[Zakres projektu 3](#_Toc510689800)

[Przedsięwzięcie 3](#_Toc510689801)

[Użyte technologie 4](#_Toc510689802)

[Diagram Gantta 5](#_Toc510689803)

[Diagram Klas 7](#_Toc510689804)

[Baza Danych 7](#_Toc510689805)

[Rodzaje dostępów i funkcjonalności 7](#_Toc510689806)

[Konta w Aplikacji: 7](#_Toc510689807)

[Panele w Aplikacji: 8](#_Toc510689808)

[Możliwości w aplikacji: 8](#_Toc510689809)

[Opis Użycia 8](#_Toc510689810)

# Cel Projektu

Celem jest stworzenie uniwersalnego systemu do zarządzania zadaniami w przedsiębiorstwie. System posiadać będzie wiele funkcjonalności dając różne uprawnienia użytkownikom na podstawie posiadanego stanowiska i uprawnień w firmie. Aplikacja pozwoli zarządzać czasem, zadaniami, zasobami finansowymi i ludzkimi. Umożliwi również szybką komunikację pomiędzy użytkownikami oraz ułatwi prowadzenie obserwacji nad postępami w poszczególnych zadaniach. Zrealizowanie założeń projektu wymaga stworzenia środowiska i mechanizmów, które pozwolą na szybkie i niezawodne działanie programu. Aplikacja ma być intuicyjna i łatwa obsłudze, o prostym interfejsie, który będzie odpowiadał wymaganiom wielu użytkowników. Baza danych posłuży do przechowywania i modyfikacji wszystkich potrzebnych danych do zapewnienia pełnej funkcjonalności aplikacji.

W dokumentacji zawarte zostaną informacje dotyczące metod realizacji projektu, implementacji oraz użytych technologii.

# Zakres projektu

- Projekt będzie realizowany w 4 osobowej grupie z wykorzystaniem systemów GitHub i Jira

- Opracowanie wizji projektu i przygotowanie środowiska pracy

- Opracowanie diagramów technicznych (przypadków użycia, aktywności, sekwencji)

- Wstępne zaprojektowanie interfejsu użytkownika w Java FX Builder umożliwiające łatwe dodawanie dalszych modułów uwzględniając jednocześnie różne prawa dostępu. Logowanie z ustawionymi na „sztywno” użytkownikami

- Zaprojektowanie schematu bazy danych. Utworzenie diagramu ERD

- Budowanie złożonych kwerend SQL (budowa kwerend, które wybierają z kilku tabel)

- Stworzenie biblioteki generującej raporty w PDF i uruchomienie brakujących modułów

- Wykonanie brakujących elementów (testów, javadoc) oraz instalatora

- Tworzenie pakietów instalacyjnych

- Finalizacja i prezentacja projektu

# Przedsięwzięcie

Projekt będzie realizowany na zasadach metody SCRUM:  
- Samoorganizacja wewnątrz zespołu  
- Odpowiedzialność zbiorowa w ramach grupy  
- 2 tygodniowe Sprinty  
- Utrzymywanie aktualnych diagramów w Jira  
- Realizacja trzech Daily Sprintów na każdy Sprint

# Użyte technologie

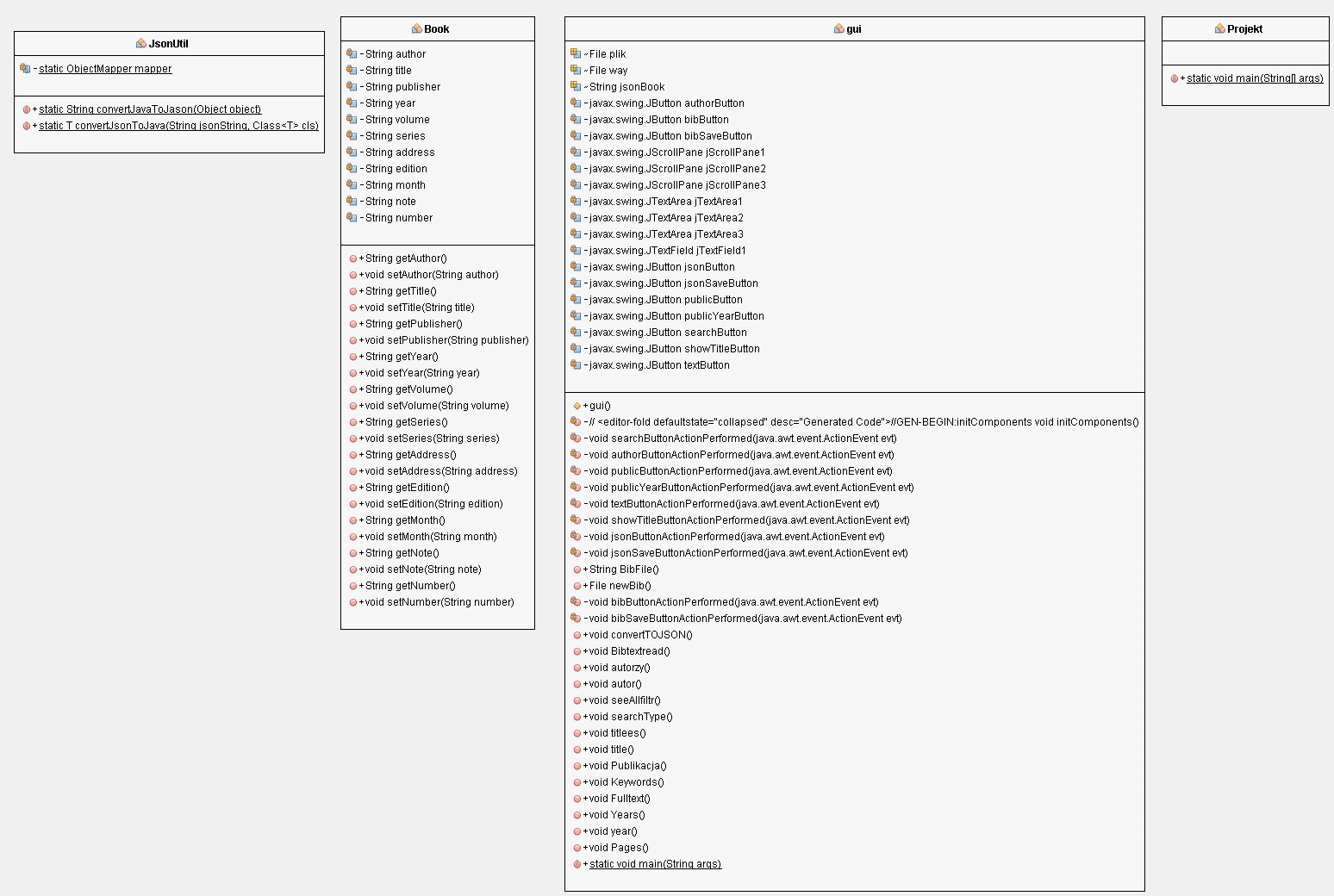
* **Java**  
  Obiektowy język programowania stworzony przez grupę roboczą pod kierunkiem Jamesa Goslinga z firmy Sun Microsystems. Java jest językiem tworzenia programów źródłowych kompilowanych do kodu bajtowego, czyli postaci wykonywanej przez maszynę wirtualną. Język cechuje się silnym typowaniem. Jego podstawowe koncepcje zostały przejęte z języka [Smalltalk](https://pl.wikipedia.org/wiki/Smalltalk" \o "Smalltalk) (maszyna wirtualna, zarządzanie pamięcią) oraz z języka C++ (duża część składni i słów kluczowych).
* **JavaFX Builder**  
  JavaFX rozszerza potencjał platformy Java, pozwalając programistom na używanie bibliotek środowiska Java w aplikacjach JavaFX. W ten sposób programiści mogą poszerzyć swoje możliwości względem platformy Java i skorzystać z technologii prezentacji, jaką oferuje JavaFX, umożliwiając konstruowanie środowisk o atrakcyjnej szacie graficznej. Użytkownicy mogą uruchamiać aplikacje JavaFX w przeglądarce lub przeciągać je na pulpit.
* **MySQL (MariaDB)**  
  MariaDB to [baza danych](https://pl.wikipedia.org/wiki/Baza_danych) stworzona przez grupę byłych pracowników MySQL AB, pod przewodnictwem Michaela Wideniusa, współtwórcy [MySQL](https://pl.wikipedia.org/wiki/MySQL). Celem głównym projektu jest współpraca ze społecznością [wolnego oprogramowania](https://pl.wikipedia.org/wiki/Wolne_oprogramowanie) i udostępnianie jej na licencji [GPL](https://pl.wikipedia.org/wiki/GNU_General_Public_License), w przeciwieństwie do niepewnego statusu licencji MySQL, która zależy teraz od firmy [Oracle](https://pl.wikipedia.org/wiki/Oracle_Corporation). Autorzy MariaDB stawiają sobie również za cel utrzymanie kompatybilności z wcześniejszymi wersjami MySQL. [Kod źródłowy](https://pl.wikipedia.org/wiki/Kod_%C5%BAr%C3%B3d%C5%82owy) MariaDB bazuje na wcześniejszej wersji MySQL
* **GitHub**Git jest rozproszonym systemem kontroli wersji. Stanowi wolne oprogramowanie i został opublikowany na licencji GNU GPL. Pierwsza wersja narzędzia Git została wydana 7 kwietnia 2005 roku. Cechuje się dobrym wsparciem dla rozgałęzionego procesu tworzenia oprogramowania, algorytmów łączenia zmian z dwóch gałęzi, dodawaniem własnych algorytmów, pracą off-line tzn. każdy programista posiada własną kopię repozytorium lokalnie, gdzie może zapisywać zamiany na bieżąco. GitHub również wspiera istniejące protokoły sieciowe:  HTTP(S), FTP, rsync, SSH. Umożliwia efektywną pracę z dużymi projektami.
* **Jira**Jira to [zamknięte oprogramowanie](https://pl.wikipedia.org/wiki/Zamkni%C4%99te_oprogramowanie) firmy [Atlassian](https://pl.wikipedia.org/w/index.php?title=Atlassian&action=edit&redlink=1" \o "Atlassian (strona nie istnieje)) służące do [śledzenia błędów](https://pl.wikipedia.org/wiki/Bugtracker) oraz [zarządzania projektami](https://pl.wikipedia.org/wiki/Zarz%C4%85dzanie_projektami). Umożliwia zarządzanie dokumentacją, zasobami firmy, ułatwia wymianę informacji wewnątrz firmy, daje możliwość tworzenia kartotek pracowników i ewidencji zgłoszeń oraz paneli. W głównej mierze pozwala na zarządzanie projektami i zadaniami.

# Diagram Gantta

Przedstawia podział projektu na zadania oraz czas, w jakim zostały wykonane.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr | Zadanie | Początek | Trwanie | Zakończenie |
| 1 | Opracowanie wizji projektu i przygotowanie środowiska pracy | 1 mar | 14 | 15 mar |
| 2 | Działające okna (zakładki) min. 2 modułów (bez bazy danych) | 15 mar | 21 | 5 kwi |
| 3 | działająca i wypełniana automatycznie baza danych | 5 kwi | 14 | 19 kwi |
| 4 | podłączenie bazy danych do istniejących okienek | 19 kwi | 21 | 10 maj |
| 5 | stworzenie biblioteki generującej raporty w PDF i uruchomienie brakujących modułów | 10 maj | 14 | 24 maj |
| 6 | wykonanie brakujących elementów (testów, javadoc, itd.) oraz instalatora | 24 maj | 14 | 7 cze |

# Diagram Klas

Został on wygenerowany w środowisku NetBeans 8.2 DO ZROBIENIA NA KONIEC ****

# Baza Danych

Baza danych została oparta na technologii

# Rodzaje dostępów i funkcjonalności

## Konta w Aplikacji:

- Konto Właściciela Zadania / Projektu -> czyli przedstawiciela klienta, który zleca firmie wykonanie zadania/projektu. Dzięki posiadanemu dostępowi będzie mógł kontrolować i śledzić ogólny proces wykonania projektu + ewentualnie dodawać informacja o poprawkach jakie by chciał aby zostały wprowadzone w zadaniu / projekcie

- Konto Szefa (właściciel firmy wykonującej zadanie/projekt) -> w sumie to podobnie jak wyżej z tym, że to on będzie tworzył nowe zadania/projektu w aplikacji i przypisywał je do odpowiednich kierowników i liderów.

- Konto Kierownika Zadania / Projektu -> osoba, która zarządza grupą lub grupami osób pracujących nad zadaniem/projektem.

- Konto Lidera Grupy np. programistów

- Konto Pracownika -> otrzyma wgląd do otrzymanych zadań i będzie mógł „informować” o postępach w zadaniach do niego przypisanych.

## Panele w Aplikacji:

- Panelu Właściciela Zadania / Projektu

- Panelu Szefa (Właściciela firmy)

- Panel tworzenia nowych zadań/projektów

- Panel z listą zadań/projektów aktualnie wykonywanych

- Panel konkretnego Zadania/Projektu z informacjami ,kto nad czym pracuje + statystyki postępu itd.

- Panelu Kierownika Zadania / Projektu

- Panel Zarządzania Grupami

- Panelu Lidera Grupy

- Panelu Pracownika

- Panelu Logowania

## Możliwości w aplikacji:

- Tworzenie Zadań / Projektów, nadawanie im nazwy roboczej, informacje, dla kogo jest wykonywany dany projekt, określony budżet i termin zakończenia projektu, przypisywanie do nich kierowników

- Tworzenie grup i przypisywanie do nich pracowników przez Liderów oraz określanie zakresu zadań, jakie mają wykonać + terminy itd.

- Tworzenie konkretnych zadań i przypisywanie do nich pracowników przez Liderów

- Prowadzenie rozmów na chatcie w obrębie grupy

## Opis Użycia