**Specyfikacja Systemu Zarządzania Zadaniami**

**Zespół Projektowy:**

* **Piotr Szpila**
* **Patryk Sadok**
* **Sebastian Powroźnik**
* **Krystian Pinderski**
* **Maciej Piotrowski**

**SPIS TREŚCI**

[1. Moduły w Aplikacji 2](#_Toc516108831)

[2. Grupy użytkowników 2](#_Toc516108832)

[3. Problematyka projektu 3](#_Toc516108833)

[4. Opis Interesariuszy Projektu: 3](#_Toc516108834)

[5. Szczegółowy opis i ocena wpływu siły: 4](#_Toc516108835)

[6. Przetrzymywanie oraz wprowadzanie danych do systemu 6](#_Toc516108837)

[7. Raporty, typy oraz prawa dla grup do ich generowania 7](#_Toc516108838)

[8. Typy wymaganych dokumentów w projekcie oraz dostęp do nich 8](#_Toc516108839)

[9. Ustalenie przepływu informacji w środowisku systemu 8](#_Toc516108840)

[10. Ustalenie uprawnień poszczególnych grup użytkowników 9](#_Toc516108841)

[11. Wymagania funkcjonalne 10](#_Toc516108842)

[12. Wykres Gantta 12](#_Toc516108843)

[13. Diagram przypadków użycia 13](#_Toc516108844)

[14. Diagram aktywności 14](#_Toc516108845)

[15. Diagram sekwencji 15](#_Toc516108846)

[16. Diagram ERD 19](#_Toc516108847)

[17. Analiza kodu zespołu pomarańczowych 20](#_Toc516108848)

[18. Skrypt tworzący bazę danych 25](#_Toc516108849)

1. Moduły w Aplikacji

* Moduł rejestracji nowych użytkowników - odpowiadać będzie za pobieranie informacji o nowych użytkownikach systemu, walidacji ich oraz po poprawnej rejestracji zapisaniu danych w bazie danych
* Moduł główny zawierać będzie w sobie widoki planów projektowych, użytkowników pracujących nad danym projektem, chat, oraz zadania we wszystkich fazach ich wykonywania.
* Moduł generowania oraz wysyłania maili do użytkowników systemu wykorzystywany do autentykowania użytkowników.
* Moduł generowania raportów - moduł zawierający narzędzia do generowania raportów z osiągnięć zespołu takich jak:
  + wykonane zadania
  + postęp projektu
  + zadania których nie udało się wykonać w danym czasie
  + dlaczego zadanie zostało niewykonane

1. Grupy użytkowników

W systemie istnieją następujące grupy użytkowników:

* administrator systemu,
* pracownik firmy,
* kierownik zespołu.

1. Problematyka projektu

Utworzenie aplikacji do zarządzania zadaniami. Pracownicy będą podzieleni na zespoły, będą posiadać swojego kierownika. Pracownicy będą mieli możliwość podglądu przypisanych do niego zadań oraz generowanie raportów (w formacie .PDF) z postępu prac oraz komunikacji z innymi użytkownikami serwisu.

Kierownik natomiast będzie miał uprawnienia do podglądu wszystkich zadań w projekcie oraz komunikacji z innymi użytkownikami serwisu. Ponadto będzie mógł generować raporty (w formacie .PDF), obrazujące członków projektu oraz zadania realizowane w projekcie.

Administrator będzie odpowiadał: za zarządzanie projektami ich dodawanie, edycja, usuwanie, przydzielanie zadań, za zarządzanie użytkownikami tj. ich blokowanie lub odblokowywanie, ich usuwanie, wymuszenie na nich zamiany hasła. Ponadto administrator będzie mógł się komunikować z innymi użytkownikami serwisu oraz generować raporty (w formacie .PDF), obrazujące członków projektu oraz zadania realizowane w projekcie.

1. Opis Interesariuszy Projektu:

**Interesariusze wewnętrzni:**

* Zarząd: Piotr Szpila
* Kadra Kierownicza, Scrum Master: Sebastian Powroźnik
* Grupa Programistów: Piotr Szpila, Sadok Patryk, Sebastian Powroźnik, Maciej Piotrowski
* Grupa FrontEnd-owców: Krystian Pinderski
* Grupa Dokumentacyjna: Maciej Piotrowski

**Interesariusze zewnętrzni:**

* Konkurenci: Inne grupy realizujące podobną tematykę projektu
* Klienci: mgr Adam Szczur
* Opinia publiczna: 2 Grupa Laboratoryjna - Aplikacje Internetowe
* Inwestorzy: Brak

1. Szczegółowy opis i ocena wpływu siły:



**Stanowisko Interesariuszy:**



## **Interesy szkodzące (blokujące) przeciwdziałanie:**

Blokujące działania:

* możliwa zmiana założeń projektu zmieniająca znaczącą część logiki aplikacji,
* ograniczenie możliwości commitowania osób spoza zespołu,
* zamknięty kanał konwersacji w serwisie #Slack w celu wymiany poglądów na temat projektu,
* możliwe usterki serwisowe github.com uniemożliwiające zaktualizowanie lokalnych repozytoriów członków zespołu,
* negatywna opinia klienta nt. logiki istniejących lub przyszłych funkcjonalności aplikacji,
* autentykacja użytkowników w procesie rejestracji poprzez adres email ,
* autoryzacja w czasie logowania za pomocą loginu i hasła.

1. Przetrzymywanie oraz wprowadzanie danych do systemu

Dane w systemie przetrzymywane będą w bazie danych PostgreSQL, z scentralizowanym połączeniem za pomocą frameworka Hibernate. Dane wprowadzane do systemu będą przez różne grupy użytkowników, zależnie od typu danych:

* Rejestrowanie nowych użytkowników - dostęp dla każdego nie zarejestrowanego użytkownika
* Tworzenie nowych projektów – administrator,
* Wprowadzanie zadań projektowych - administrator,
* Konwersacja wewnątrz projektowa – administrator, kierownik projektu, pracownicy,
* Szczegółowe informacje o koncie – użytkownik, kierownik, administrator.

1. Raporty, typy oraz prawa dla grup do ich generowania

Raporty są generowane w formacie .PDF i zawierają informacje takie jak:

* Nazwę Projektu
* Datę wygenerowania poprzedniego raportu
* Datę wygenerowania raportu,
* Kierownik zespołu,
* Skład zespołu, w tym :
  + Imie,
  + Nazwisko
* Zadania do wykonania w projekcie, w tym:
  + Nazwa zadania,
  + Tag,
  + Priorytet
  + Czas jaki pozostał na wykonanie zadania,
  + Status zadania

Raporty może generować:

* administrator sytemu,
* kierownik projektu,

niezależnie od roli raport zawiera te same dane.

1. Typy wymaganych dokumentów w projekcie oraz dostęp do nich

* administrator w każdym momencie może wygenerować raport ukazujący informacje o dowolnym projekcie tj. zespół projektowy oraz zadania.
* kierownik projektu będzie miał możliwość wygenerowania w każdej chwili raportu, w którym będą informacje o ukazujący informacje o konkretnym projekcie tj. zespół projektowy oraz zadania,
* w każdej chwili, każdy z użytkowników systemu może uzyskać dostęp do dokumentacji oraz podręcznika użytkownika Systemu Zarządzania Zadaniami.

***\*wszystkie raporty będą generowane w plikach PDF***

1. Ustalenie przepływu informacji w środowisku systemu

Informacje w systemie będą przepływały w sposób scentralizowany, to znaczy wszystkie moduły, oraz funkcjonalności będą łączyły się z bazą danych i to z niej będą pobierały dane o projekcie.

Moduły będą łączyły się z bazą poprzez interfejs DAO (Data Access Object), który będzie pełnił rolę scentralizowanego dostępu do bazy danych. Wpłynie to znacząco na minimalizację redundancji kodu walidującego wprowadzane dane do bazy danych, oraz zmniejszy ilość wątków jednocześnie utrzymujących z nią połączenie.

1. Ustalenie uprawnień poszczególnych grup użytkowników

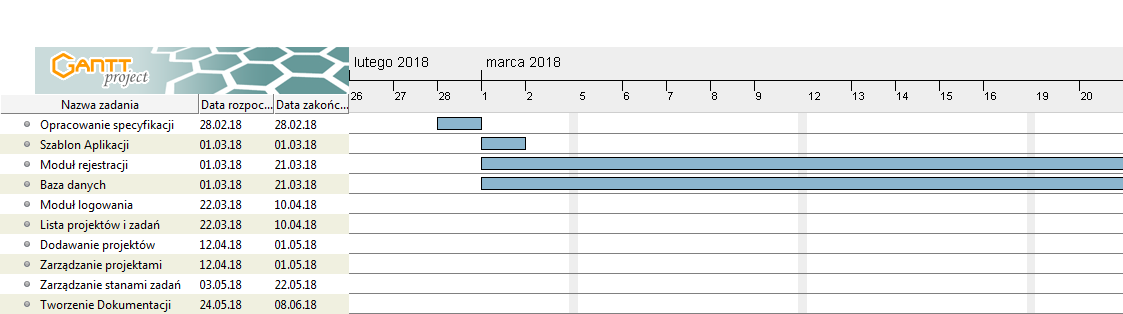
Poszczególnym użytkownikom przysługują następujące uprawnienia:

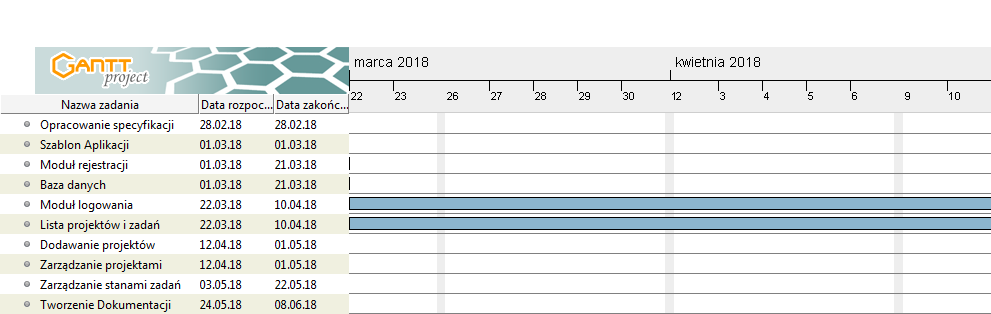
* administrator systemu:
  + tworzenie nowych projektów, przypisywanie członków oraz kierownika
  + edycja właściwości projektu,
  + usuwanie projektów,
  + wyświetlanie widoku szczegółowego zadań w projekcie,
  + tworzenie zadań oraz przydzielanie zadań członkom zespołu,
  + edycja zadań, modyfikacja statusu zadań
  + blokowanie/odblokowywanie użytkowników,
  + wymuszenie zmiany hasła przez użytkownika,
  + usuwanie użytkowników.
  + wyświetlanie oraz zmiana informacji o swoim koncie,
  + możliwość prowadzenia konwersacji z pozostałymi użytkownikami systemu za pomocą czatu oraz przeglądanie historii rozmów,
  + usuwanie wiadomości.
* kierownik zespołu:
  + podgląd zadań zespołu,
  + generowanie raportów z postępu prac,
  + wyświetlanie oraz zmiana informacji o swoim koncie,
  + możliwość prowadzenia konwersacji z pozostałymi użytkownikami systemu za pomocą czatu oraz przeglądanie historii rozmów,
  + usuwanie wiadomości.
* pracownik:
  + podgląd przypisanych do niego zadań,
  + generowanie raportów z postępu prac,
  + wyświetlanie oraz zmiana informacji o swoim koncie,
  + możliwość prowadzenia konwersacji z pozostałymi użytkownikami systemu za pomocą czatu oraz przeglądanie historii rozmów,
  + usuwanie wiadomości.

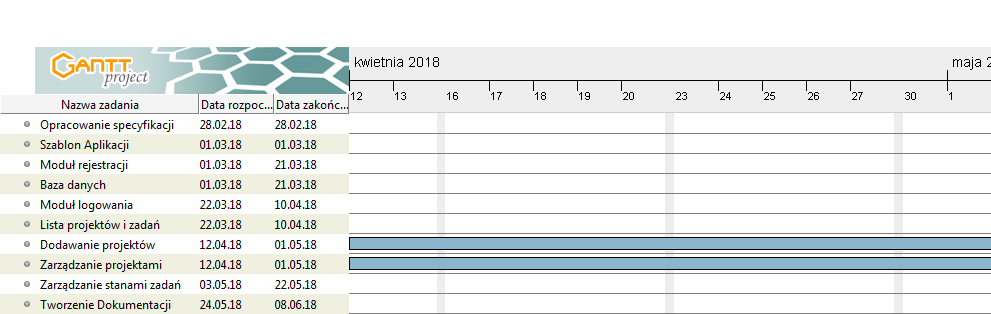
1. Wymagania funkcjonalne

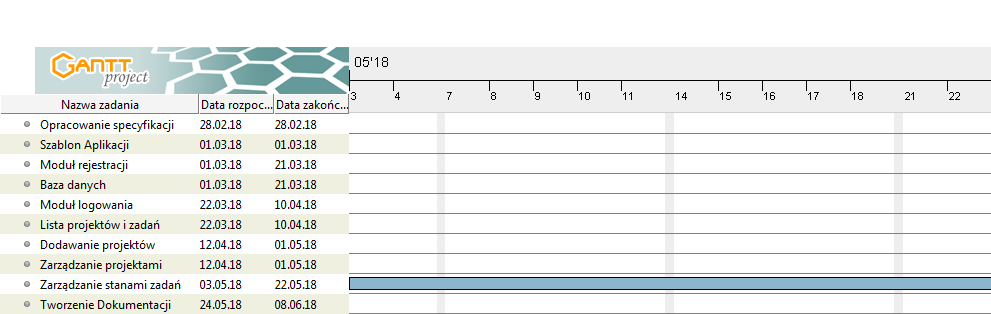
* Aplikacja ma pozwalać na możliwość rejestracji nowych użytkowników systemu, łącznie z walidacją oraz autentykacją osobową,
* Aplikacja ma mieć możliwość zalogowania się do systemu, oraz przypomnienia hasła na email w przypadku jego utraty,
* Wszyscy korzystający z aplikacji powinni mieć możliwość zmiany podstawowych informacji o swoim koncie,
* Administrator powinien mieć możliwość zobaczenia listy użytkowników wraz z możliwością ich blokady,
* Administrator powinien mieć możliwość zobaczenia podstawowych informacji o użytkowniku,
* Administrator powinien mieć możliwość usunięcia użytkownika, jego zablokowania lub odblokowania oraz wymuszenia na nim zmiany hasła,
* Administrator serwisu powinien mieć możliwość przydzielania zadań członkom zespołu,
* Administrator serwisu powinien mieć możliwość tworzenia, edycji oraz usuwania zadań zespołu w każdym stadium,
* Administrator serwisu powinien mieć możliwość generowania raportów zawierających informacje o projekcie tj. jego członkach oraz taskach,
* Administrator powinien mieć możliwość zobaczenia podstawowych informacji o projekcie,
* Administrator serwisu powinien mieć możliwość dodawania nowego projektu wraz z jego kierownikiem oraz członkami,
* Administrator serwisu powinien mieć możliwość edycji istniejącego projektu wraz ze zmianą kierownika oraz członków projektu,
* Administrator serwisu powinien mieć możliwość usunięcia istniejącego projektu,
* Administrator powinien mieć możliwość komunikacji z pozostałymi użytkownikami serwisu oraz zarządzać wiadomościami w swojej skrzynce,
* Kierownik zespołu powinien mieć możliwość podglądu zadań w całym projekcie,
* Kierownik zespołu powinien mieć możliwość komunikacji z administratorem, członkami jego zespołu na specjalnych kanałach czatowych oraz zarządzać wiadomościami w swojej skrzynce,
* Kierownik zespołu powinien mieć możliwość generowania raportów zawierających informacje o projekcie tj. jego członkach oraz zadaniach,
* Użytkownik powinien mieć możliwość zobaczenia tablicy projektów do jakich jest przydzielony,
* Użytkownik powinien mieć możliwość zobaczenia przypisanych do niego zadań w przypisanym do niego projekcie
* Użytkownik powinien mieć możliwość konwersacji z innymi członkami zespołu oraz wszystkimi użytkownikami na chacie z wydzielonymi kanałami oraz zarządzać wiadomościami w swojej skrzynce.

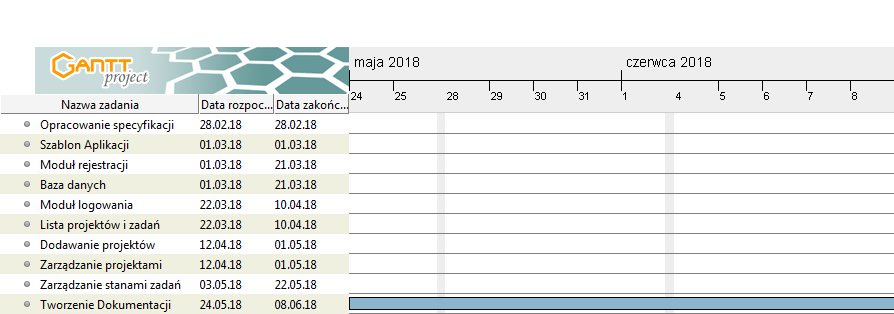
1. Wykres Gantta

****

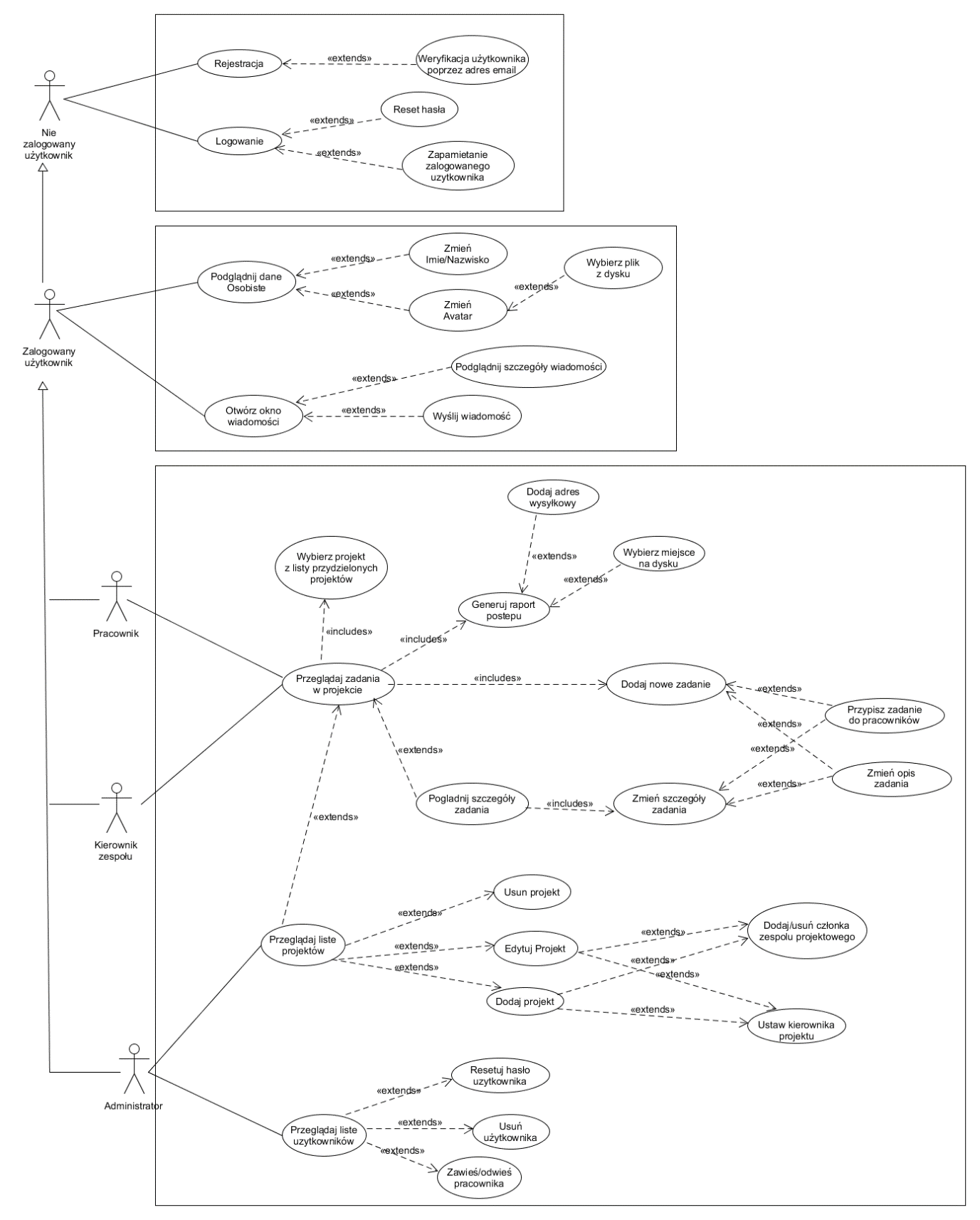
****

****

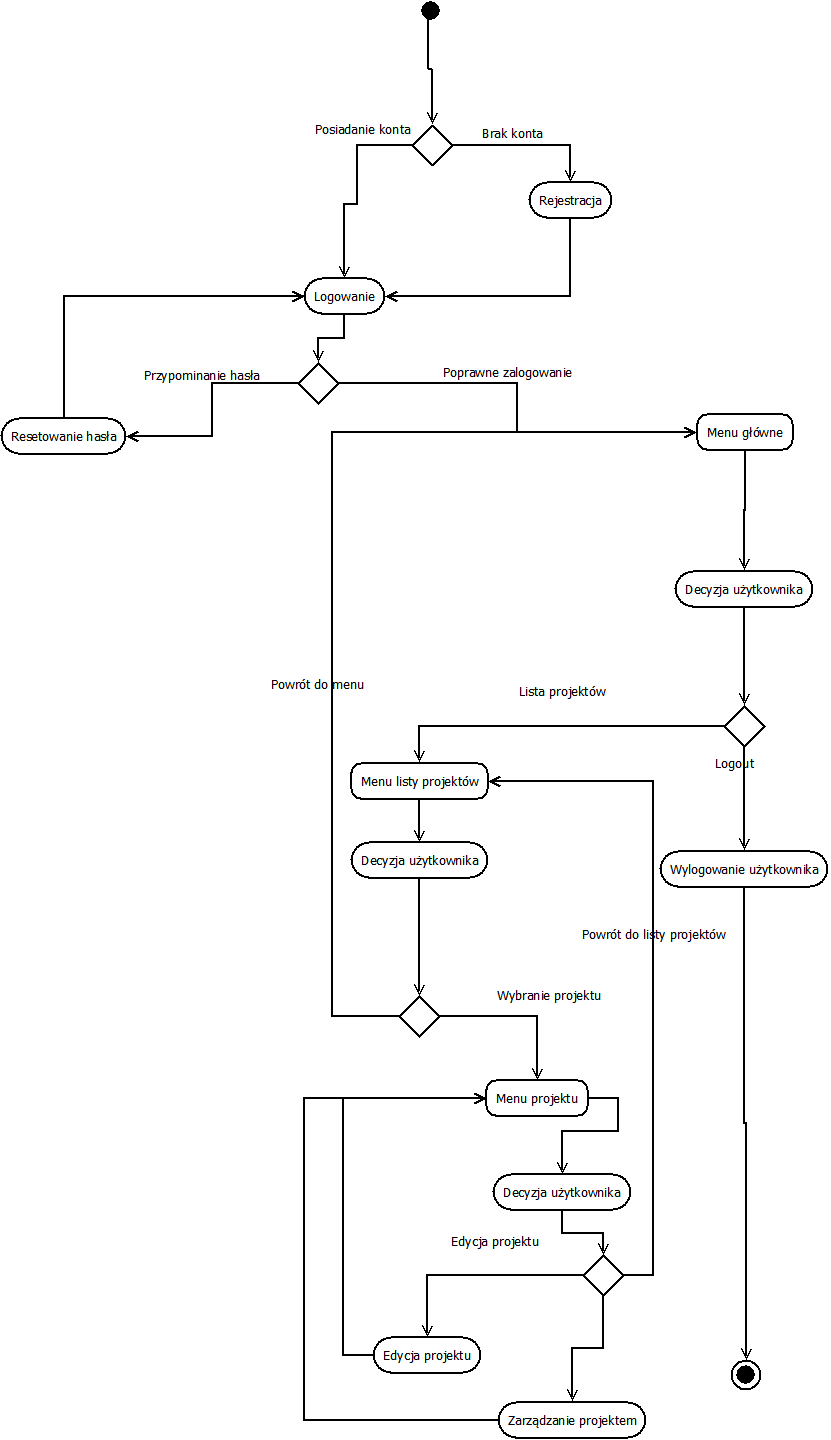
****

****

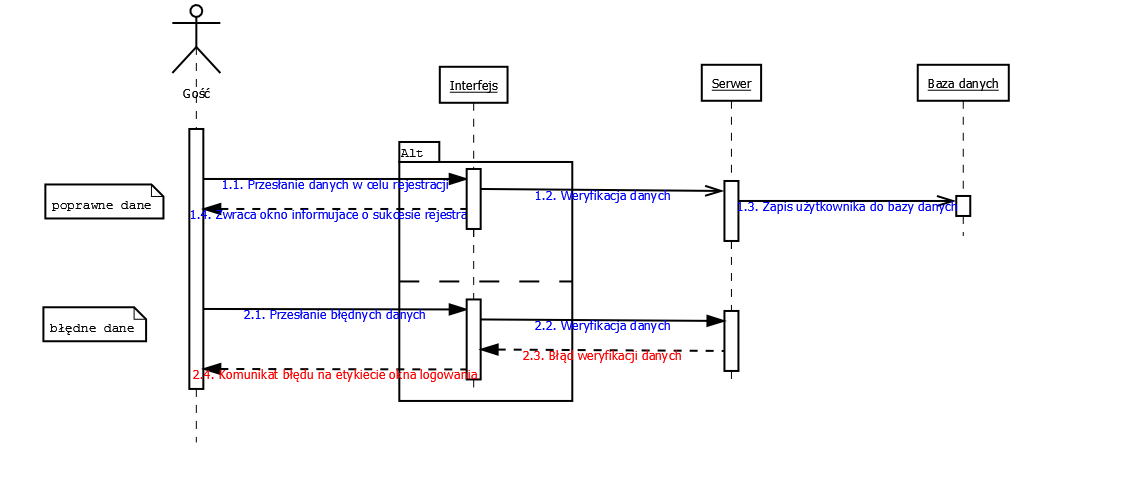
1. Diagram przypadków użycia

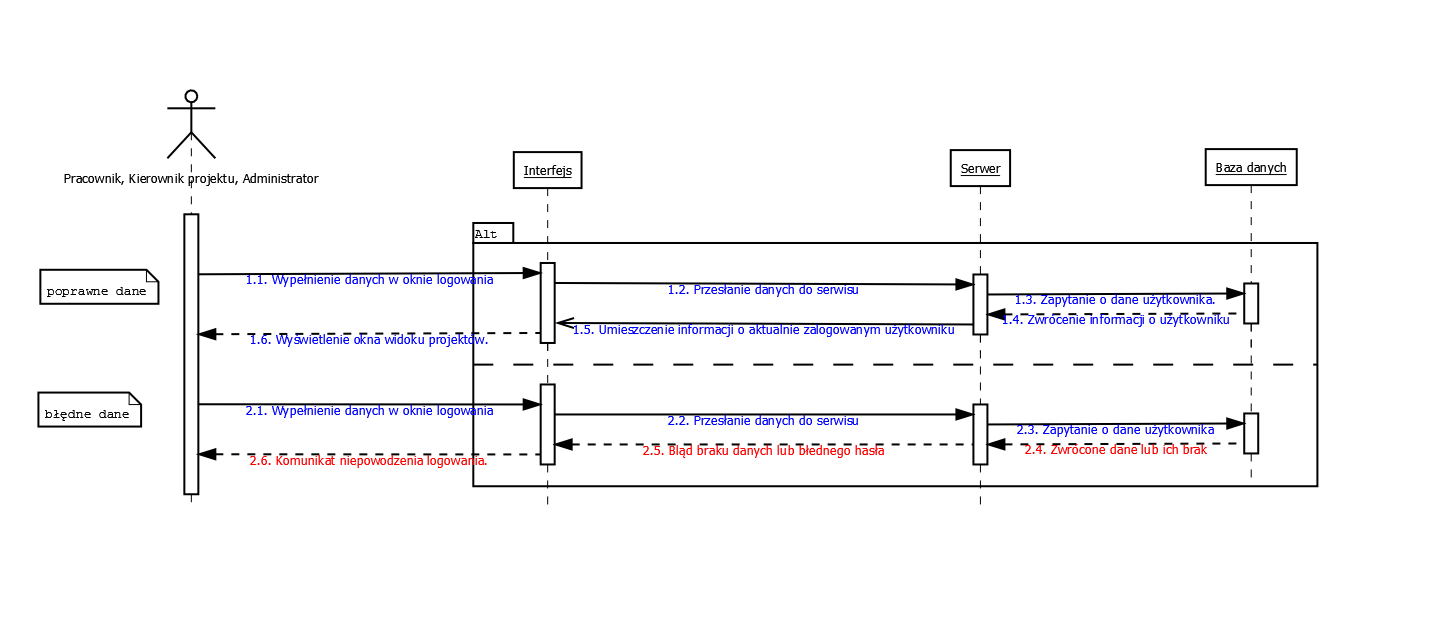


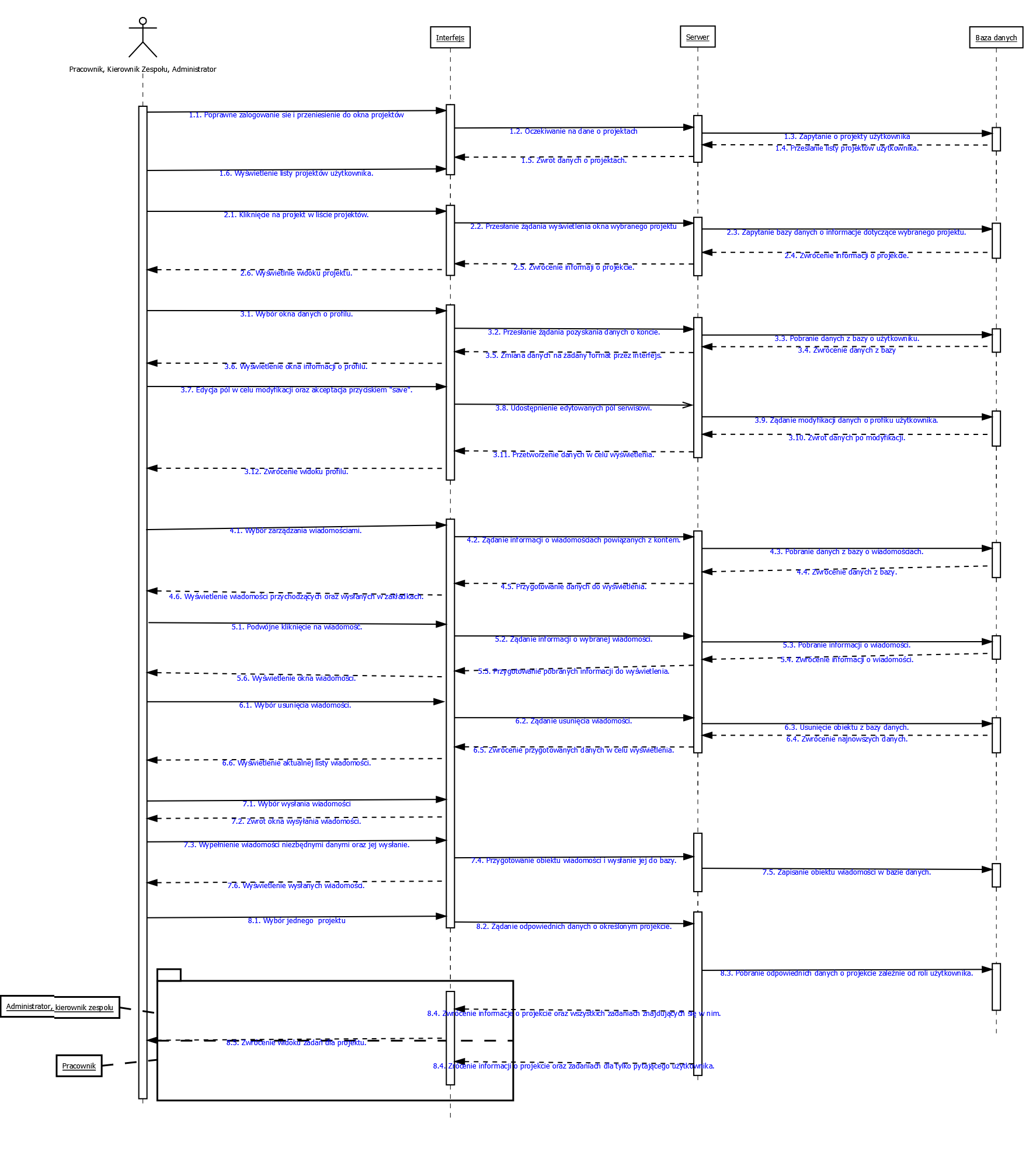
1. Diagram aktywności

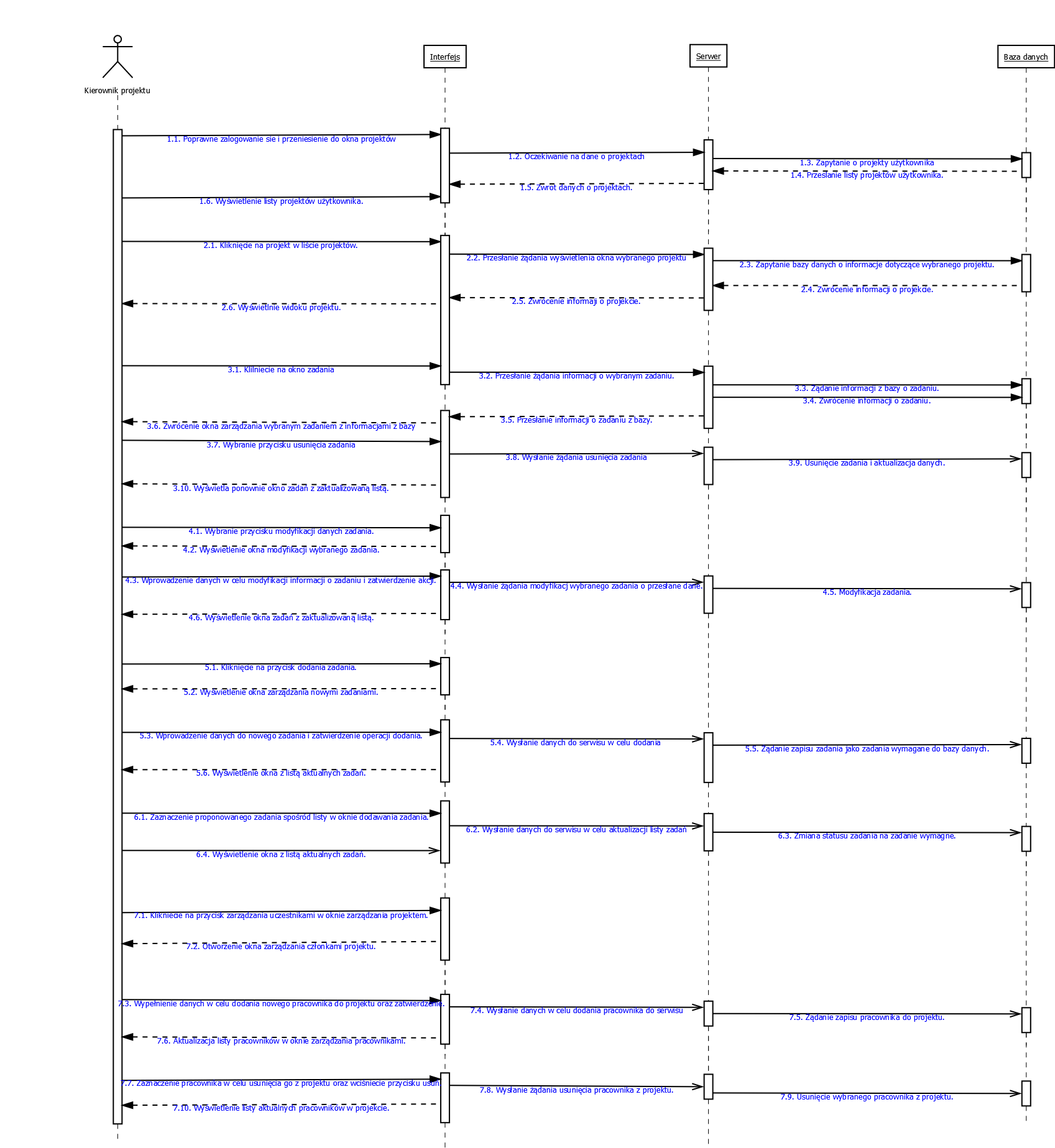
****

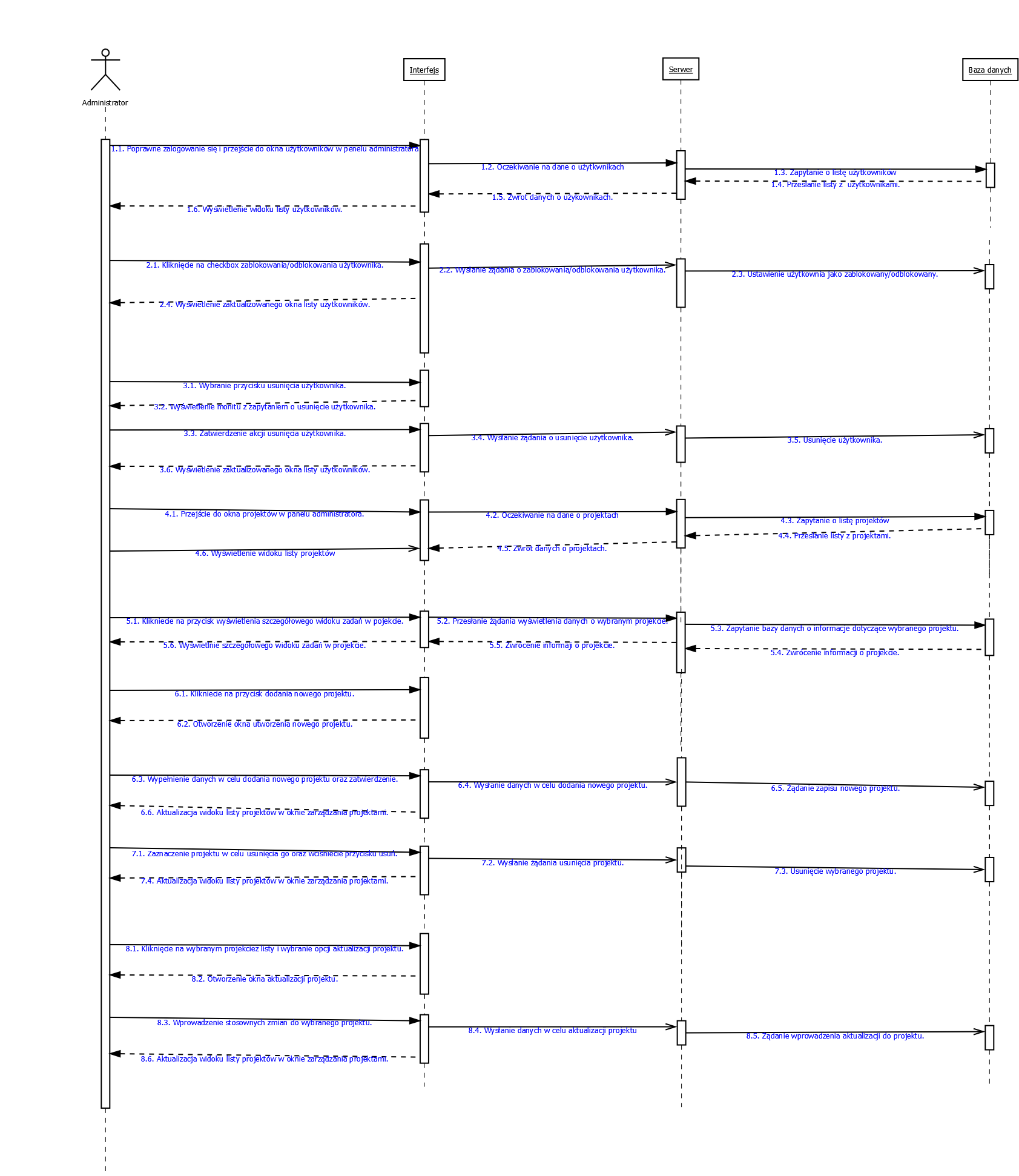
1. Diagram sekwencji

****

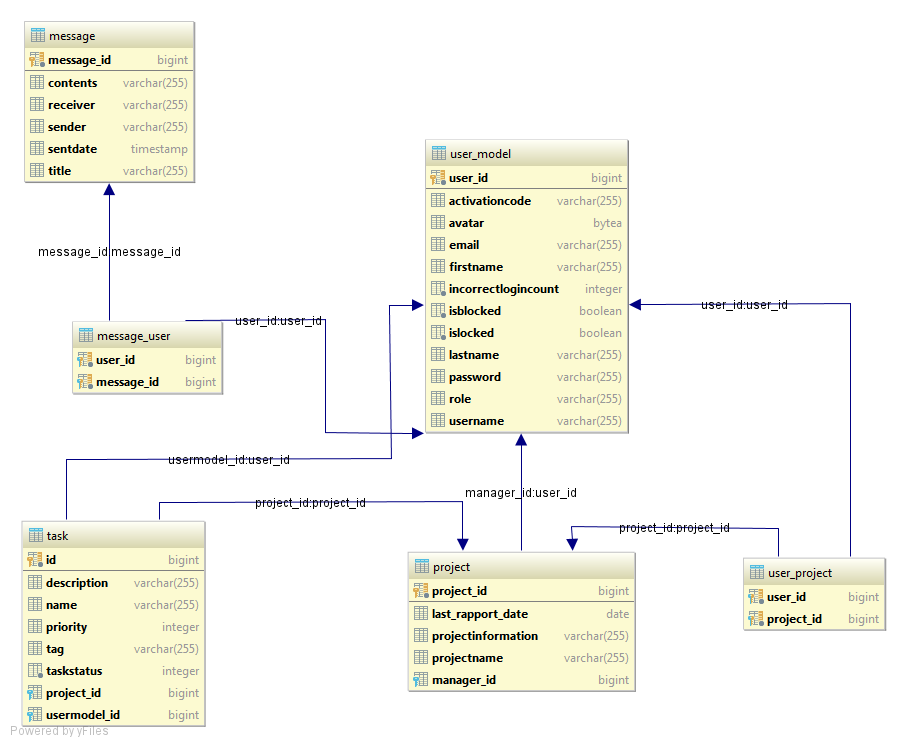
****

****





1. Diagram ERD



1. Analiza kodu zespołu pomarańczowych

Raport z kodu programu zespołu pomarańczowych - 12.04.2018

Programowanie Zespołowe:

Controllers:  
*Admin:*

* AdminPanelController.java
  + Linie 52, 53, zbędne
  + Linia 62 do usunięcia
  + Linia 21 do usunięcia
  + Linie 45, 46, brak komentarzy do parametrów
  + Linia 17, brak opisu klasy w komentarzu dokumentacyjnym
* TabGroupsTasksController.java
  + Linia 37, brak opisu klasy w komentarzu dokumentacyjnym
  + Linia 67, nazwa skrótowa
  + Linia 76, brak komentarzy do parametrów
  + Linia 94, zbędna (pusta)
  + Linia 177 system.out.println
* TabGroupsUsersController.java
  + Linia 34, brak opisu klasy w komentarzu dokumentacyjnym
  + Linia 77, brak komentarzy do parametrów
  + Linia 96, zbędna (pusta)
  + Linia 136, system.out.println
  + Linia 189, złapanie wyjątku w konsoli
  + Linia 211, złapanie wyjątku w konsoli
* TabUsersController.java
  + Linia 37, brak opisu klasy w komentarzu dokumentacyjnym
  + Linia 90, brak komentarzy do parametrów
  + Linia 145, złapanie wyjątku w konsoli
  + Linia 197, złapanie wyjątku w konsoli

*Login:*

* LoginFormController.java
  + 1-5 License basic header
  + Linia 24, brak opisu klasy w komentarzu dokumentacyjnym
  + Linia 43, brak komentarzy do parametrów

*Manager:*

* ManagerPanelController.java
  + Linia 1-5, license basic header
  + Linia 23, brak opisu klasy w komentarzu dokumentacyjnym
  + Linie 48-49, brak komentarzy do parametrów
  + Linia 55-56, zakomentowany kod
  + Linia 65 pusta
* TabGroupsTasksController.java
  + Linie 1-5, basic license header
  + Linia 20, brak opisu klasy
  + Linia 52, skrótowa nazwa
  + Linia 55, brak opisu parametrów
  + Linia 73 zbędna (pusta)
* TabGroupsUsersController.java
  + Linie 1-5, basic license header
  + Linia 20, brak opisu klasy
  + Linia 57, brak opisu parametrów
  + Linia 79 pusta

*Register*

* RegisterFormController.java
  + Linie 1-5, basic license header
  + Linia 27, brak opisu klasy
  + Linia 51, brak opisu parametrów
  + Linia 61, brak opisu metody
  + Linia 71, brak opisu metody

*Shared*

* EditMyDataController.java
  + Linia 13, brak opisu klasy
  + Linia 34, 35 brak opisu parametrów
  + Linia 46, brak opisu metody
  + Linia 45, metoda bindings() zbyt długie linie!!!
  + Linia 49, 144 znaki w linii!!!!!!!!!!!!!!!!!!!!
  + Linia 53, brak opisu metody
  + Linia 65, zakomentowany kod
  + Linia 67, brak opisu metody
* GroupDetailsController.java
  + Brak komentarza z opisem klasy
  + Linie 37, 38, brak opisu parametrów
  + Linia 48, brak opisu metody
  + Linia 60, brak opisu metody
  + Linie 64, 65, 66, puste linie
* GroupFormController.java
  + Linia 18, brak opisu klasy
  + Linie 37, 38, brak opisu parametrów
  + Linia 54, brak opisu metody
  + Linia 64, brak opisu metody
  + Linia 68 pusta
* TabReportsController.java
  + Brak opisu klasy
  + Linie 47, 48, brak opisu parametrów
  + Linia 58, brak opisu metody
  + Linia 60 pusta
  + Linia 84, system.out.println
  + Linia 105, system.out.println
  + Linia 111, pusta
* TaskDetailsController.java
  + Brak opisu klasy
  + Linie 49, 50, brak opisu parametrów
  + Linia 69, brak opisu metody
  + Linia 84, brak opisu metody
  + Linia 88 pusta
* TaskFormController.java
  + Brak opisu klasy
  + Linie 52, 53, brak opisu parametrów
  + Linia 80, brak opisu metody
  + Linia 90, brak opisu metody
  + Linia 97 pusta
* UserDetailsController.java
  + Brak opisu klasy
  + Linie 44, 45, brak komentarzy parametrów
  + Linia 54, brak opisu metody
  + Linia 69, brak opisu metody
* UserFormController.java
  + Brak opisu klasy
  + Linie 42, 43, brak opisu parametrów
  + Linia 54, brak opisu metody
  + Linia 63, brak opisu metody
  + Linia 76, zakomentowany kod
  + Linia 78, brak opisu metody
  + Linia 87, brak opisu metody
  + Linia 91, brak opisu metody

*User:*

* TabMyTasksController.java
  + Brak opisu klasy
  + Linia 45, brak opisu parametrów
  + Linia 59 pusta
* UserPanelController.java
  + Brak opisu klasy
  + Linia 35, 36, brak opisu parametrów
  + Linie 44, 45, zakomentowany kod
  + Linia 54 pusta

Pakiet database.dao

Klasa GroupDao.java

* Linia 9-11 zbędne komentarze,
* 30,36 nazwy parametrów,
* Brak javadoc,

Klasa ReportDao.java

* 8-10 Zbędne komentarze,
* Brak javadoc,

Klasa TaskDao.java

* 8-10 Zbędne komentarze,
* Brak javadoc,

Klasa UserDao.java

* Brak javadoc,
* 61 linia nazwy parametrów,

Pakiet database.model

Klasa Group, Report, Task, User

* Brak javadoc,

Pakiet database

Klasa DbSeeder

* Brak javadoc,

Klasa HibernateUtil,

* Brak javadoc,

Pakiet enums

Klasa ReportVisibility, TaskProgress, UserRole, UserState

* Brak javadoc,

Pakiet modelFx – nazwa pakietu !

Klasa GroupFX

* Brak javadoc,
* 151 usunąć

Klasa GroupModel

* Brak javadoc
* Linia 23 do usunięcia,
* Linia 29, 33, 40, 44, 54, 55, 60 nazwy zmiennych,

Klasa ReportFX

* Brak javadoc,

Klasa ReportModel

* Linia 20 do usunięcia
* Linia 24, 28, 35 nazwy zmiennych,
* Brak javadoc,

Klasa TaskFX

* Brak javadoc,

Klasa TaskModel

* Brak javadoc,
* Linia 20 usunąc,
* Linia 26, 30, 37 nazwy zmiennych,

Klasa UserFX

* Brak javadoc
* Linia 180-181 do usunięcia
* Linia 186-187 do usunięcia

Klasa UserModel

* Brak javadoc,
* Linia 34, 86, 92, 91, 92, 97, 124, 128, 132, 137, 145 nazwy zmiennych i parametrów,

Pakiet utils.converters, enums

Klasa TaskProgressStringConverter, UserRoleStringConverter, UserStateStringConverter

* Linia 13 nazwa zmiennej,
* Brak javadoc

Pakiet utils.converters

Klasa GroupConverter

* Brak javadoc,
* Linia 17, 18, 31, 37, 43, 44, 57, 63 nazwa zmiennej

Klasa ReportConverter

* Brak javadoc,
* Linia 11,12 nazwa zmiennej,

Klasa TaskConverter

* Brak javadoc,
* Linia 15, 16, 31, 37, 53 nazwa zmienej,
* Linia 58 usunąć

Klasa UserConverter

* Brak javadoc,
* Linia 12, 13, 25, 26 nazwa zmiennej,

Pakiet utils

Klasa Alerts

* Brak javadoc,
* Linia 31, 39, 47, 48 nazwy metod i zmiennych,

Klasa Loaders

* Brak javadoc,
* Linia 22, 24, 63 nazwa zmiennej,

Klasa PasswordDialog

* Brak javadoc,
* Linia 18, 25, 28, 31 nazwa zmiennej

Pakiet wms

Klasa Main

* Brak javadoc
* Linia 29 nazwa zmiennej

Klasa MainWindowController

* Brak javadoc
* Linia 49 nazwa zmiennej

1. Skrypt tworzący bazę danych

**SET statement\_timeout = 0;**

**SET lock\_timeout = 0;**

**SET idle\_in\_transaction\_session\_timeout = 0;**

**SET client\_encoding = 'UTF8';**

**SET standard\_conforming\_strings = on;**

**SELECT pg\_catalog.set\_config('search\_path', '', false);**

**SET check\_function\_bodies = false;**

**SET client\_min\_messages = warning;**

**SET row\_security = off;**

**--**

**-- Name: plpgsql; Type: EXTENSION; Schema: -; Owner: -**

**--**

**CREATE EXTENSION IF NOT EXISTS plpgsql WITH SCHEMA pg\_catalog;**

**--**

**-- Name: EXTENSION plpgsql; Type: COMMENT; Schema: -; Owner: -**

**--**

**COMMENT ON EXTENSION plpgsql IS 'PL/pgSQL procedural language';**

**SET default\_tablespace = '';**

**SET default\_with\_oids = false;**

**--**

**-- Name: message; Type: TABLE; Schema: public; Owner: -**

**--**

**CREATE TABLE public.message (**

**id bigint NOT NULL,**

**contents character varying(255),**

**receiver character varying(255),**

**sender character varying(255),**

**sentdate timestamp without time zone,**

**title character varying(255)**

**);**

**--**

**-- Name: message\_id\_seq; Type: SEQUENCE; Schema: public; Owner: -**

**--**

**CREATE SEQUENCE public.message\_id\_seq**

**START WITH 1**

**INCREMENT BY 1**

**NO MINVALUE**

**NO MAXVALUE**

**CACHE 1;**

**--**

**-- Name: message\_id\_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: -**

**--**

**ALTER SEQUENCE public.message\_id\_seq OWNED BY public.message.id;**

**--**

**-- Name: message\_user; Type: TABLE; Schema: public; Owner: -**

**--**

**CREATE TABLE public.message\_user (**

**message\_id bigint NOT NULL,**

**user\_id bigint NOT NULL**

**);**

**--**

**-- Name: project; Type: TABLE; Schema: public; Owner: -**

**--**

**CREATE TABLE public.project (**

**project\_id bigint NOT NULL,**

**projectinformation character varying(255),**

**projectname character varying(255),**

**manager\_id bigint**

**);**

**--**

**-- Name: project\_project\_id\_seq; Type: SEQUENCE; Schema: public; Owner: -**

**--**

**CREATE SEQUENCE public.project\_project\_id\_seq**

**START WITH 1**

**INCREMENT BY 1**

**NO MINVALUE**

**NO MAXVALUE**

**CACHE 1;**

**--**

**-- Name: project\_project\_id\_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: -**

**--**

**ALTER SEQUENCE public.project\_project\_id\_seq OWNED BY public.project.project\_id;**

**--**

**-- Name: task; Type: TABLE; Schema: public; Owner: -**

**--**

**CREATE TABLE public.task (**

**id bigint NOT NULL,**

**description character varying(255),**

**name character varying(255),**

**priority integer,**

**tag character varying(255),**

**taskstatus integer NOT NULL,**

**project\_id bigint**

**);**

**--**

**-- Name: task\_id\_seq; Type: SEQUENCE; Schema: public; Owner: -**

**--**

**CREATE SEQUENCE public.task\_id\_seq**

**START WITH 1**

**INCREMENT BY 1**

**NO MINVALUE**

**NO MAXVALUE**

**CACHE 1;**

**--**

**-- Name: task\_id\_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: -**

**--**

**ALTER SEQUENCE public.task\_id\_seq OWNED BY public.task.id;**

**--**

**-- Name: user\_model; Type: TABLE; Schema: public; Owner: -**

**--**

**CREATE TABLE public.user\_model (**

**user\_id bigint NOT NULL,**

**activationcode character varying(255),**

**email character varying(255),**

**firstname character varying(255),**

**isblocked boolean NOT NULL,**

**islocked boolean NOT NULL,**

**lastname character varying(255),**

**password character varying(255),**

**role character varying(255),**

**username character varying(255)**

**);**

**--**

**-- Name: user\_model\_user\_id\_seq; Type: SEQUENCE; Schema: public; Owner: -**

**--**

**CREATE SEQUENCE public.user\_model\_user\_id\_seq**

**START WITH 1**

**INCREMENT BY 1**

**NO MINVALUE**

**NO MAXVALUE**

**CACHE 1;**

**--**

**-- Name: user\_model\_user\_id\_seq; Type: SEQUENCE OWNED BY; Schema: public; Owner: -**

**--**

**ALTER SEQUENCE public.user\_model\_user\_id\_seq OWNED BY public.user\_model.user\_id;**

**--**

**-- Name: user\_project; Type: TABLE; Schema: public; Owner: -**

**--**

**CREATE TABLE public.user\_project (**

**user\_id bigint NOT NULL,**

**project\_id bigint NOT NULL**

**);**

**--**

**-- Name: message id; Type: DEFAULT; Schema: public; Owner: -**

**--**

**ALTER TABLE ONLY public.message ALTER COLUMN id SET DEFAULT nextval('public.message\_id\_seq'::regclass);**

**--**

**-- Name: project project\_id; Type: DEFAULT; Schema: public; Owner: -**

**--**

**ALTER TABLE ONLY public.project ALTER COLUMN project\_id SET DEFAULT nextval('public.project\_project\_id\_seq'::regclass);**

**--**

**-- Name: task id; Type: DEFAULT; Schema: public; Owner: -**

**--**

**ALTER TABLE ONLY public.task ALTER COLUMN id SET DEFAULT nextval('public.task\_id\_seq'::regclass);**

**--**

**-- Name: user\_model user\_id; Type: DEFAULT; Schema: public; Owner: -**

**--**

**ALTER TABLE ONLY public.user\_model ALTER COLUMN user\_id SET DEFAULT nextval('public.user\_model\_user\_id\_seq'::regclass);**