

System zarządzania zasobami ludzkimi w firmie

Katarzyna Kucharczyk, Michał Mazek, Łukasz Żmuda, Michał Barański

24 maja 2014

Spis treści

1	Analiza wymagań	3
1.1	Wymagania funkcjonalne	3
1.2	Wymagania niefunkcjonalne	5
2	Aktorzy systemu	9
3	Przypadki użycia	10
3.1	Autoryzacja użytkownika	10
3.2	Dodanie/edycja profilu użytkownika	10
3.3	Usuwanie profilu użytkownika	11
3.4	Zmiana praw dostępu	12
3.5	Edycja zwierzchnictwa	12
3.6	Tworzenie projektu	13
3.7	Modyfikacja projektu	14
3.8	Usunięcie projektu	14
3.9	Tworzenie zadania	15
3.10	Modyfikacja zadania	16
3.11	Usunięcie zadania	16
3.12	Logowanie czasu pracy	17
3.13	Modyfikacja loga z czasem pracy	18
3.14	Usunięcie loga z czasem pracy	19
3.15	Przypisanie pracownika do projektu	19
3.16	Usunięcie pracownika z projektu	20
3.17	Zatwierdzenie raportu pracownika	21
3.18	Złożenie wniosku o urlop pracownika	21
3.19	Przyjęcie wniosku o urlop pracownika	22
3.20	Generowanie raportu dotyczącego pracownika	23
3.21	Generowanie raportu dotyczącego projektu	23
3.22	Ostrzeganie przed sytuacjami krytycznymi w projekcie	24
4	Architektura sytemu	26
4.1	Diagramy klas	26
4.2	Diagram komponentów	26
4.3	Diagramy sekwencji	27
5	Specyfikacja techniczna	34
5.1	Wstępna specyfikacja sprzętu i oprogramowania podstawowego	34
5.2	Specyfikacja technologii realizacji oprogramowania systemu	35
5.3	Specyfikacja parametrów sprzętu	36

1 Analiza wymagań

System skierowany jest do firm zajmujących się wytwarzaniem produktów IT. Skierowany jest do wszystkich zatrudnionych w firmie, z głównym wskazaniem na:

- menadżerów projektów,
- pracowników technicznych (programistów, testerów)
- analityków.

System ma za zadanie wspomaganie zarządzania projektami oraz zasobami niezbędnymi do ich wytwarzania. Ma umożliwiać stałe monitorowanie postępów projektu. Ma zapewniać możliwość szybkiej reakcji na zmiany (odejście/choroba pracownika, wcześniejsze/późniejsze zakończenie projektu).

1.1 Wymagania funkcjonalne

Głównym zadaniem systemu jest wsparcie zarządzania projektami pod kątem:

Lp.	Nazwa	Opis	Priorytet
Zarządzanie personelem			
F001	Uwierzytelnianie	Aplikacja umożliwia uwierzytelnianie osób korzystających z systemu	Wysoki
F002	Zarządzanie uprawnieniami	System umożliwia tworzenie kont z różnymi poziomami uprawnień	Wysoki
F003	Walidacja poziomu uprawnień	System dostosowuje widoczne dla użytkownika opcje w zależności od poziomu uprawnień użytkownika	Wysoki
F004	Zarządzanie pracownikami	System umożliwia dodawanie usuwanie i modyfikację kont pracowników. W ramach konta jednego pracownika system umożliwia wprowadzenie: <ul style="list-style-type: none">- danych personalnych- kwalifikacji- historii zatrudnienia- stawki- informacji o stanowisku	Wysoki
F005	Zarządzanie strukturą organizacyjną firmy	System umożliwia przeglądanie zależności stanowisk-owych pracowników firmy. System umożliwia tworzenie, usuwanie i modyfikacje elementów tak rozumianej struktury firmy.	Wysoki

F006	Zarządzanie czasem pracy pracowników	System umożliwia tworzenie raportów przez pracownika dotyczących przepracowanego czasu na poszczególnych zadaniach. W szczególności system umożliwia wprowadzenie informacji o stanie zaawansowania prac nad zadaniem którego dotyczy raport.	Wysoki
F007	Obsługa urlopów pracownika	System umożliwia składanie wniosków urlopowych przez pracownika i ich zatwierdzenie, bądź odrzucenie.	Średni
F008	Generowanie raportów dotyczących czynności wykonywanych przez pracownika	System umożliwia generowanie raportu na temat czasu poświęconego na poszczególne czynności wchodzące w skład zadań.	Średni
Zarządzanie projektami			
F101	Tworzenie projektu	System umożliwia tworzenie i usuwanie projektów. System umożliwia wprowadzenie oraz modyfikację danych projektu np. status projektu.	Wysoki
F102	Terminarz projektu	Dla każdego projektu system umożliwia tworzenie usuwanie i modyfikację terminarza projektu	Wysoki
F103	Definiowanie zadań	Dla każdego projektu system umożliwia dodanie i usunięcie zadania. System umożliwia także modyfikowanie danych zadania np. nazwa zadania, opis zadania	Wysoki
F104	Wymagania projektu	System umożliwia przypisanie do projektu kwalifikacji które musi posiadać zespół, aby można wykonać zaplanowaną pracę .	Wysoki
F105	Wyszukiwanie członków zespołów	System umożliwia wyszukiwanie pracowników o odpowiednich kwalifikacjach. Dla każdego znalezionego pracownika system pokazuje informacje o aktualnym obciążeniu pracownika	Wysoki
F106	Alokacja pracowników do projektów	System umożliwia przypisanie pracownika do projektu.	Wysoki

F107	Obciążenie pracowników	System udostępnia informacje o obciążeniu każdego z pracowników, czyli w nad jakimi projektami pracuje, oraz na jaki czas pracownik został 'zarezerwowany' w związku z danym projektem	Wysoki
F108	Zarządzanie kosztami projektów	System umożliwia: - definiowanie kosztów poszczególnych pracowników - zatwierdzanie raportów czasu pracy pracowników.	Wysoki
Monitorowanie statusu projektów			
F201	Monitorowanie kosztów	System automatycznie informuje o przekroczeniu planowanych kosztów zadań lub całego projektu	Wysoki
F202	Zaawansowanie pracy	System automatycznie określa stan prac na podstawie danych podanych w raportach pracowników	Wysoki
F203	Podgląd zaawansowania pracy	System pozwala na podgląd zaawansowania prac nad całym projektem oraz w rozbiciu na poszczególne zadania	Wysoki
F204	Generowanie raportów dotyczących projektu	Generowanie raportu do pliku, w raporcie informacje o zadaniach, kto się nimi zajmuje, ile czasu dotychczas to zajęło i ile procentowo jest zrobione.	Średni
F205	Ostrzeganie przed sytuacjami krytycznymi	Generowanie alertów w sytuacjach krytycznych, np. opóźnienia projektu, bądź przekroczenia kosztów.	Wysoki
Zarządzanie systemem			
F301	Generowanie profili	System pozwala generować profile umiejętności kandydatów.	Średni
F302	Archiwizacja danych	System umożliwia archiwizację danych związanych z projektami	Średni

1.2 Wymagania niefunkcjonalne

System ma zapewniać:

Lp.	Nazwa	Opis	Priorytet
Bezpieczeństwo			
NF001	Uwierzytelnienie użytkowników	W celach autoryzacji użytkownika jest niezbędne zastosowanie mechanizmów z wykorzystaniem prokołu zabezpieczającego połączenie (np. SSL).	Wysoki
NF002	Brak możliwości dostępu osób nie autoryzowanych	Podział ról (np. administrator) ma zapewniać zestaw odpowiednich uprawnień, dzięki którym użytkownik może dostać się do modułów systemu zgodnych z jego kompetencjami.	Wysoki
NF003	Możliwość skorzystania z aplikacji przy korzystaniu jedynie z sieci lokalnej firmy	Brak możliwości skorzystania z aplikacji, poza placówką firmy (nie dotyczy korzystania z usługi VPN).	Wysoki
NF004	Wymaganie od użytkownika stosowania złożonych haseł	Użytkownik w celu zachowania bezpieczeństwa swojego konta oraz danych systemu ma używać haseł złożonych z minimum 8 znaków, w których skład wchodzi duże i małe litery, oraz liczby.	Wysoki
NF005	Szyfrowanie danych w bazie	W razie sytuacji ataku i kradzieży danych należy zapewnić aby baza z poufnymi danymi użytkowników nie była rozszyfrowywalna przez osoby postronne.	Wysoki
NF006	Przechowywanie informacji (logów) dotyczących użytkownika systemu	Prowadzenie dzienników daje możliwość wglądu w opatrzone dokładnymi datami spis działań na systemie i jego środowisku (np. czas użytkownika, adresy ip). Dzięki temu w razie ataku możliwe jest odtworzenie towarzyszących zdarzeń.	Średni
Dostępność			

NF107	System ma być dostępny dla użytkowników w czasie godzin pracy zgodnych z ustalonymi normami	System powinien być dostępny w godzinach pracy, oraz poza nimi (w przypadku dodatkowych terminów, lub nienormowanego czasu pracy). System może być niedostępny 3 dni w skali roku - ze względu z zaplanowanymi pracami (utrzymanie, aktualizacja) oraz czasu reakcji na awarie, z preferencją na noc i dni wolne od pracy.	Wysoki
NF108	System powinien zapewniać szybką reakcję na działania użytkownika	System powinien reagować w ciągu 5 sekund na działania użytkownika (poza obciążającymi zadaniami jak np. generacja raportów).	Średni
Reakcja na awarie			
NF209	Tworzenie wersji zapasowych	System powinien zapewniać cotygodniowe, automatyczne tworzenie backupów, oraz możliwość przywrócenia wersji zapasowej	Wysoki
NF210	Krytyczne sytuacje awaryjne powinny być automatycznie przechwytywane	Wszelkie błędy działania systemu powinny być automatycznie wykrywane i przekazywane do administratora systemu	Wysoki
Skalowalność			
NF311	Możliwość obsłużenia wzrastającej ilości użytkowników i projektów	W ramach rozwoju formy systemu powinien być przygotowany na posiadanie zwiększonej ilości danych o pracownikach oraz nowych projektach	Wysoki
NF312	Możliwość obsłużenia dużej ilości zapytań	Przy rozwoju firmy ilość zapytań od użytkowników będzie wzrastać - responsywność systemu powinna zostać zachowana w granicach założonych limitów czasowych	Średni
NF313	Możliwość rozwoju systemu	System ma dawać możliwość dołączania nowych funkcjonalnych modułów	Wysoki
Ograniczenia systemu			

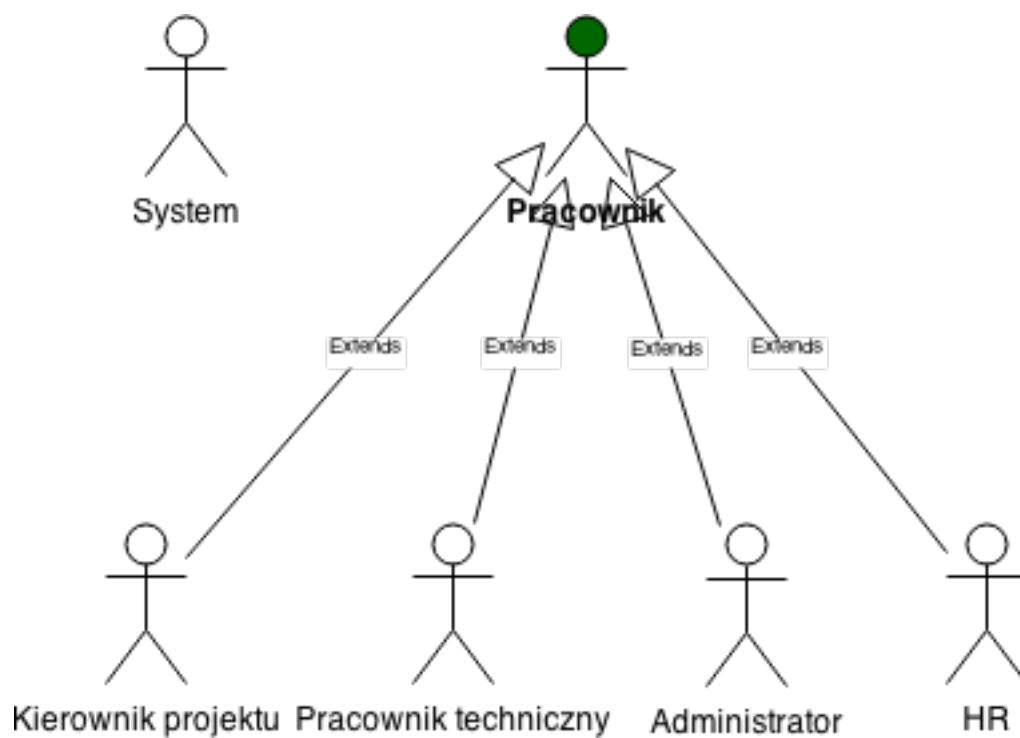
NF414	Udostępnienie usługi dla ilości użytkowników i projektów zgodnie z dostarczonymi zasobami	Zgodnie z ustaloną wielkością zasobów system powinien obsłużyć odpowiadającą ilość profili użytkowników, oraz projektów	Wysoki
Użytkowanie			
NF515	Dostarczenie intuicyjnego interfejsu użytkownika	System powinien być przejrzysty i intuicyjny, nawet dla nowych użytkowników	Średni
NF516	Łatwe poznanie dodatkowych funkcjonalności	Poznanie wszelkich bardziej skomplikowanych funkcjonalności, oraz przedstawienie systemu dla mniej obeznanych z podobnymi technologiami pracowników, powinno być możliwe dzięki około godzinnego kursu	Niski
NF516	Dostęp ze standardowej przeglądarki	System powinien być dostępny i prawidłowo działać przy użyciu jednej z trzech standardowych przeglądarek (Chrome, Firefox, Opera)	Średni

2 Aktorzy systemu

Aplikacja będzie używana przez różne osoby w firmie. W ramach aplikacji zdefiniowane są następujące role, jakie pełnione są przez użytkowników:

- System
- Administrator
- Pracownik
- Kierownik projektu
- HR

W ramach każdej roli zdefiniowany jest inny zakres możliwości i funkcji dostępnych w systemie.



Rysunek 1: Aktorzy systemu

3 Przypadki użycia

Wszelkie przypadki użycia posiadają warunek poprawnego przejścia scenariusza pierwszego (UC1 Autoryzacja użytkownika).

3.1 Autoryzacja użytkownika

Nazwa: UC1 Autoryzacja użytkownika
Opis: Proces logowania się pracownika
Aktorzy: Użytkownik
Warunki początkowe:

- Istniejące konto w systemie

Warunki końcowe:

- Dostęp do systemu

Scenariusz główny:

1. Aktor wprowadza swoje login i hasło
2. System sprawdza zapytanie
3. System wyświetla ekran powitalny

Scenariusz alternatywny - błędne dane:

1. Aktor wprowadza swoje login i hasło
2. System uznaje zapytanie za błędne
3. System wyświetla ekran informujący o błędnym loginie lub hasle

3.2 Dodanie/edycja profilu użytkownika

Nazwa: UC2 Dodanie/edycja profilu użytkownika
Opis: Proces dodania/edycji profilu dla pracownika
Aktorzy: HR
Warunki początkowe:

- Dokument z informacjami dotyczącymi pracownika

Warunki końcowe:

- Uaktualniony profil pracownika

Scenariusz główny:

1. Aktor otwiera formularz profilu pracownika
2. Aktor uzupełnia/zmienia wybrane pola

3. Aktor potwierdza zmiany odpowiednim przyciskiem
4. System zapisuje w bazie danych nowe dane
5. System wyświetla ekran poglądowy z nowym profilem użytkownika

Scenariusz alternatywny - walidacja danych:

1. Jak a-c w scenariuszu głównym
2. System wykrywa błędnie wpisane dane
3. System zwraca formularz z zaznaczonymi błędnymi polami

Scenariusz alternatywny - utrata łączności:

1. Jak a-c w scenariuszu głównym
2. System ma problem z łącznością
3. System wyświetla formularz, wraz z informacją o nie zapisaniu danych spowodowanych problemem łączności

Scenariusz alternatywny - kolizja transakcji:

1. Jak a-c w scenariuszu głównym
2. System wykrywa błąd transakcji (np. zakleszczenie)
3. System wyświetla formularz, wraz z prośbą o ponowienie zapytania

3.3 Usuwanie profilu użytkownika

Nazwa: UC3 Usuwanie profilu użytkownika

Opis: Proces usuwania profilu dla pracownika

Aktorzy: HR lub Administrator

Warunki początkowe:

- Informacja o koncie do usunięcia

Warunki końcowe:

- Zaktualizowana baza danych pozbawiona nieporządanego konta

Scenariusz główny:

1. Aktor otwiera listę użytkowników
2. Aktor wybiera opcje usunięcie konta przy nazwisku
3. System usuwa rekord z informacjami o użytkowniku
4. System wyświetla nową listę użytkowników

Scenariusz alternatywny - błąd usuwania:

1. Jak a-b w scenariuszu głównym
2. System wykrywa błąd podczas usuwania rekordu
3. System zwraca pierwotną listę z informacją z błędem usuwania

3.4 Zmiana praw dostępu

Nazwa: UC4 Zmiana praw dostępu

Opis: Proces zmiany uprawnień użytkownika do określonych modułów.

Aktorzy: Administrator

Warunki początkowe:

- Dotychczas nadane prawa

Warunki końcowe:

- Uaktualniony profil z nowymi rolami

Scenariusz główny:

1. Aktor wybiera z listy edytowane konto użytkownika
2. System wyświetla detale profilu użytkownika
3. Aktor wybiera odpowiednią opcję ról i zatwierdza
4. System zmienia rekord dotyczący ról dla danego konta
5. System zwraca widok profilu użytkownika z nowymi prawami dostępu

Scenariusz alternatywny - błąd zmiany :

1. Jak a-c w scenariuszu głównym
2. System wykrywa kolizję lub błąd połączenia
3. System wyświetla detale profilu użytkownika wraz z kodem błędu, który wystąpił

3.5 Edycja zwierzchnictwa

Nazwa: UC5 Edycja zwierzchnictwa

Opis: Proces zmiany hierarchii drzewa struktury firmy.

Aktorzy: Administrator lub HR

Warunki początkowe:

- Dokument dotyczący hierarchii firmy

Warunki końcowe:

- Uaktualnione drzewo hierarchii w systemie

Scenariusz główny:

1. Aktor wybiera z listy edytowane konto użytkownika
2. System wyświetla detale profilu użytkownika
3. Aktor wybiera użytkownika który ma być zwierzchnikiem dla edytowanego użytkownika

4. System zmienia rekord dotyczący zwierzchnika
5. System zwraca nowy widok profilu użytkownika

Scenariusz alternatywny - wykrycie pętli:

1. Jak a-c w scenariuszu głównym
2. System wykrywa pętle w drzewie hierarchii
3. System wyświetla informacje o błędzie

3.6 Tworzenie projektu

Nazwa: UC6 Tworzenie projektu

Opis: Proces odpowiedzialny za tworzenie projektu

Aktorzy: Kierownik projektu

Warunki początkowe:

- Brak projektu w systemie.

Warunki końcowe:

- Nowy projekt w systemie, bez stworzonych zadań.

Scenariusz główny:

1. Aktor wybiera utworzenie nowego projektu.
2. System wyświetla formularz z pustymi polami (Nazwa projektu, opis, data początkowa, data końcowa, status)
3. Aktor uzupełnia puste pola i zatwierdza.
4. System zwraca nowy widok z projektami, gdzie pojawia się nowo dodany projekt.

Scenariusz alternatywny - podanie błędnych danych projektu:

1. Jak a-c w scenariuszu głównym
2. System wyświetla informacje o błędzie
3. Aktor wprowadza prawidłowe dane
4. Jak d w scenariuszu głównym

3.7 Modyfikacja projektu

Nazwa: UC7 Modyfikacja projektu

Opis: Proces odpowiedzialny za modyfikację projektu

Aktorzy: Kierownik projektu

Warunki początkowe:

- W systemie istnieje projekt.

Warunki końcowe:

- Zmodyfikowany projekt w systemie.

Scenariusz główny:

1. Aktor wybiera modyfikację projektu.
2. System wyświetla formularz z danymi projektu
3. Aktor zmienia pola i zatwierdza.
4. System zwraca nowy widok z projektami.

Scenariusz alternatywny - podanie błędnych danych modyfikowanego projektu:

1. Jak a-c w scenariuszu głównym
2. System wyświetla informacje o błędzie
3. Aktor wprowadza prawidłowe dane
4. Jak d w scenariuszu głównym

3.8 Usunięcie projektu

Nazwa: UC8 Usunięcie projektu

Opis: Proces odpowiedzialny za usunięcie projektu

Aktorzy: Kierownik projektu

Warunki początkowe:

- W systemie istnieje projekt.

Warunki końcowe:

- W systemie nie ma projektu.

Scenariusz główny:

1. Aktor wybiera usunięcie projektu.
2. System wyświetla komunikat w celu potwierdzenia usunięcia.
3. Aktor zatwierdza wybór.

4. System zwraca widok z projektami.

Scenariusz alternatywny - brak potwierdzenia:

1. Jak a-b w scenariuszu głównym
2. Aktor nie potwierdza usunięcia.
3. Jak d w scenariuszu głównym

3.9 Tworzenie zadania

Nazwa: UC9 Tworzenie zadania

Opis: Proces odpowiedzialny za tworzenie zadania

Aktorzy: Kierownik projektu

Warunki początkowe:

- W systemie nie ma informacji o zadaniu, należącym do projektu.

Warunki końcowe:

- Nowy utworzone zadanie należące do projektu.

Scenariusz główny:

1. Aktor wybiera interesujący go projekt.
2. System wyświetla listę zadań należących do projektu, listę pracowników przypisanych do projektu oraz dane projektu.
3. Aktor wybiera utworzenie nowego zadania.
4. System wyświetla formularz z pustymi polami (Nazwa zadania, opis, data początkowa, data końcowa, szacowane koszty, status)
5. Aktor uzupełnia puste pola i zatwierdza.
6. System zwraca nowy widok z projektem w którym pojawia się nowo utworzone zadanie.

Scenariusz alternatywny - podanie błędnych danych zadania:

1. Jak a-e w scenariuszu głównym
2. System wyświetla informacje o błędzie
3. Aktor wprowadza prawidłowe dane
4. Jak f w scenariuszu głównym

3.10 Modyfikacja zadania

Nazwa: UC10 Modyfikacja zadania

Opis: Proces odpowiedzialny za modyfikację zadania

Aktorzy: Kierownik projektu

Warunki początkowe:

- W systemie istnieje zadanie należące do projektu.

Warunki końcowe:

- W systemie istnieje zmienione zadanie należące do projektu.

Scenariusz główny:

1. Aktor wybiera interesujący go projekt.
2. System wyświetla listę zadań należących do projektu, listę pracowników przypisanych do projektu oraz dane projektu.
3. Aktor wybiera modyfikację zadania.
4. System wyświetla formularz z danymi zadania.
5. Aktor zmienia dane i zatwierdza.
6. System zwraca nowy widok z projektem w którym pojawia się zmodyfikowane zadanie.

Scenariusz alternatywny - podanie błędnych danych zadania modyfikowanego:

1. Jak a-e w scenariuszu głównym
2. System wyświetla informacje o błędzie
3. Aktor wprowadza prawidłowe dane
4. Jak f w scenariuszu głównym

3.11 Usunięcie zadania

Nazwa: UC11 Usunięcie zadania

Opis: Proces odpowiedzialny za usunięcie zadania

Aktorzy: Kierownik projektu

Warunki początkowe:

- W systemie istnieje zadanie należące do projektu.

Warunki końcowe:

- Brak zadania w systemie.

Scenariusz główny:

1. Aktor wybiera interesujący go projekt.
2. System wyświetla listę zadań należących do projektu, listę pracowników przypisanych do projektu oraz dane projektu.
3. Aktor wybiera usunięcie zadania.
4. System wyświetla komunikat w celu potwierdzenia usunięcia.
5. Aktor zatwierdza wybór.
6. System zwraca widok projektu bez usuniętego zadania.

Scenariusz alternatywny - brak potwierdzenia usunięcia zadania:

1. Jak a-d w scenariuszu głównym
2. Aktor nie zatwierdza wyboru.
3. System zwraca poprzedni widok projektu.

3.12 Logowanie czasu pracy

Nazwa: UC12 Logowanie czasu pracy

Opis: Proces zalogowania informacji o tym, czym zajmował się pracownik, kiedy to robił i ile czasu na to poświęcił

Aktorzy: Pracownik

Warunki początkowe:

- Brak zalogowanych informacji z danego dnia, w zadaniu mogą istnieć logi z innych dat, bądź innych użytkowników.

Warunki końcowe:

- Zalogowane w odpowiednim zadaniu informacje.

Scenariusz główny:

1. Aktor wybiera projekt.
2. System wyświetla listę zadań należących do projektu, listę pracowników przypisanych do projektu oraz dane projektu.
3. Aktor wybiera zadanie do którego chce dodać loga.
4. System wyświetla listę logów należących do zadania.
5. Aktor wybiera dodanie nowego loga.
6. System wyświetla formularz z pustymi polami (czas pracy, opis, data początkowa, data końcowa)
7. Aktor uzupełnia puste pola i zatwierdza.

8. System zwraca nowy widok gdzie pojawia się nowo wprowadzony log z widoczną nazwą użytkownika, którego dotyczy.

Scenariusz alternatywny - podanie błędnych danych w logu:

1. Jak a-g w scenariuszu głównym
2. System wyświetla informacje o błędzie
3. Aktor wprowadza prawidłowe dane
4. Jak h w scenariuszu głównym

3.13 Modyfikacja loga z czasem pracy

Nazwa: UC13 Modyfikacja loga z czasem pracy

Opis: Proces modyfikacji logowania.

Aktorzy: Pracownik techniczny

Warunki początkowe:

- Istnieje log w systemie.

Warunki końcowe:

- W systemie istnieje zmodyfikowany log.

Scenariusz główny:

1. Aktor wybiera projekt.
2. System wyświetla listę zadań należących do projektu, listę pracowników przypisanych do projektu oraz dane projektu.
3. Aktor wybiera zadanie.
4. System wyświetla listę logów należących do zadania.
5. Aktor wybiera modyfikację loga.
6. System wyświetla formularz z treścią loga.
7. Aktor zmienia pola i zatwierdza.
8. System zwraca nowy widok gdzie pojawia się zmodyfikowany log z widoczną nazwą użytkownika, którego dotyczy.

Scenariusz alternatywny - podanie błędnych danych w logu:

1. Jak a-g w scenariuszu głównym
2. System wyświetla informacje o błędzie
3. Aktor wprowadza prawidłowe dane
4. System zwraca poprzedni widok zadania.

3.14 Usunięcie loga z czasem pracy

Nazwa: UC14 Usunięcie loga z czasem pracy

Opis: Proces usunięcia loga z czasem pracy

Aktorzy: Pracownik techniczny

Warunki początkowe:

- W systemie istnieje log z czasem pracy.

Warunki końcowe:

- W systemie nie istnieje log z czasem pracy.

Scenariusz główny:

1. Aktor wybiera projekt.
2. System wyświetla listę zadań należących do projektu, listę pracowników przypisanych do projektu oraz dane projektu.
3. Aktor wybiera zadanie.
4. System wyświetla listę logów należących do zadania.
5. Aktor wybiera usunięcie loga.
6. System wyświetla komunikat w celu potwierdzenia usunięcia.
7. Aktor zatwierdza wybór.
8. System zwraca widok zadania bez usuniętego loga.

Scenariusz alternatywny - brak potwierdzenia usunięcia loga:

1. Jak a-d w scenariuszu głównym
2. Aktor nie zatwierdza wyboru.
3. System zwraca poprzedni widok zadania.

3.15 Przypisanie pracownika do projektu

Nazwa: UC15 Przypisanie pracownika do projektu

Opis: Proces przypisanie pracownika do projektu

Aktorzy: Kierownik projektu

Warunki początkowe:

- Do projektu nie ma przypisanego pracownika technicznego.

Warunki końcowe:

- Do projektu jest przypisany pracownik techniczny.

Scenariusz główny:

1. Aktor wybiera projekt.
2. System wyświetla listę zadań należących do projektu, listę pracowników przypisanych do projektu oraz dane projektu.
3. Aktor wybiera dodanie pracownika do projektu.
4. System wyświetla listę pracowników.
5. Aktor wybiera pracownika i zatwierdza.
6. System zwraca widok projektu z odświeżoną listę pracowników.

Scenariusz alternatywny - pracownik niedostępny:

1. Jak a-e w scenariuszu głównym
2. System informuje że pracownik jest niedostępny.
3. System zwraca poprzedni widok projektu.

3.16 Usunięcie pracownika z projektu

Nazwa: UC16 Usunięcie pracownika z projektu

Opis: Proces usunięcia pracownika z projektu

Aktorzy: Kierownik projektu

Warunki początkowe:

- Do projektu jest przypisany pracownik techniczny.

Warunki końcowe:

- Do projektu nie ma przypisanego pracownika technicznego.

Scenariusz główny:

1. Aktor wybiera projekt.
2. System wyświetla listę zadań należących do projektu, listę pracowników przypisanych do projektu oraz dane projektu.
3. Aktor wybiera pracownika z listy pracowników przypisanych do projektu.
4. Aktor wybiera usunięcie pracownika.
5. System wyświetla komunikat w celu potwierdzenia usunięcia.
6. Aktor zatwierdza wybór.
7. System zwraca widok projektu z odświeżoną listę pracowników.

Scenariusz alternatywny - pracownik niedostępny:

1. Jak a-e w scenariuszu głównym
2. Aktor nie potwierdza usunięcia.
3. System zwraca poprzedni widok projektu.

3.17 Zatwierdzenie raportu pracownika

Nazwa: UC17 Zatwierdzenie raportu pracownika

Opis: Proces zatwierdzenia raportu pracownika

Aktorzy: Kierownik projektu

Warunki początkowe:

- W systemie znajdują się logi pracownika nie zaakceptowane przez kierownika projektu.

Warunki końcowe:

- W systemie znajdują się logi pracownika zaakceptowane przez kierownika projektu.

Scenariusz główny:

1. Aktor wybiera opcję wyświetlenia logów projektu.
2. Aktor wybiera projekt z listy.
3. System wyświetla logi wprowadzone przez wszystkich pracowników należących do projektu.
4. Aktor wybiera opcję aktualizacji statusu loga.
5. System wyświetla komunikat w celu potwierdzenia zatwierdzenia loga.
6. Aktor zatwierdza wybór.
7. System zwraca widok logów projektu.

Scenariusz alternatywny - brak potwierdzenia:

1. Jak a-e w scenariuszu głównym
2. Aktor nie wyraża zgody na zatwierdzenie.
3. Jak g w scenariuszu głównym.

3.18 Złożenie wniosku o urlop pracownika

Nazwa: UC18 Złożenie wniosku o urlop pracownika

Opis: Złożenie wniosku o urlop pracownika

Aktorzy: Pracownik techniczny

Warunki początkowe:

- Pracownik chce wziąć urlop.

Warunki końcowe:

- W systemie jest złożony wniosek o urlop.

Scenariusz główny:

1. Aktor wybiera składanie wniosków o urlop.
2. System wyświetla formularz wniosku.
3. Aktor wypełnia wniosek i zatwierdza.
4. System wyświetla informację, o tym, że wniosek został poprawnie złożony.

Scenariusz alternatywny - wniosek niepoprawny:

1. Jak a-c w scenariuszu głównym
2. System wyświetla informację, o tym, że wniosek zawiera błędy.
3. Aktor poprawia wniosek i zatwierdza.
4. Jak d w scenariuszu głównym.

3.19 Przyjęcie wniosku o urlop pracownika

Nazwa: UC19 Przyjęcie wniosku o urlop pracownika

Opis: Przyjęcie wniosku o urlop pracownika

Aktorzy: HR

Warunki początkowe:

- W systemie jest wniosek o urlop.

Warunki końcowe:

- W systemie jest informacja o przyznaniu urlopu.

Scenariusz główny:

1. Aktor wybiera z wniosków z listy.
2. System wyświetla formularz wniosku.
3. Aktor zatwierdza zgodę na przyznanie urlopu.
4. System wyświetla informację, o tym, że wniosek został przyjęty.

Scenariusz alternatywny - wniosek nieprzyznany:

1. Jak a-b w scenariuszu głównym
2. System wyświetla informację, o tym, że wniosek zawiera błędy.
3. Aktor nie zatwierdza zgody na przyznanie urlopu.
4. System wyświetla informację, o tym, że wniosek nie został przyjęty.

3.20 Generowanie raportu dotyczącego pracownika

Nazwa: UC20 Generowanie raportu dotyczącego pracownika

Opis: Generowanie raportu dotyczącego pracownika

Aktorzy: Kierownik projektu, HR

Warunki początkowe:

- W systemie znajdują się logi umieszczone w zadaniach

Warunki końcowe:

- Raport zapisany pliku na dysku komputera.

Scenariusz główny:

1. Aktor wybiera z listy pracownika.
2. Aktor podaje przedział czasowy.
3. System wyświetla wszystkie logi pracownika dla przedziału czasowego.
4. Aktor wybiera generowanie raportu.
5. System generuje raport i zapisuje go na dysku komputera.
6. System potwierdza wygenerowanie raportu.

Scenariusz alternatywny - brak logów dla pracownika:

1. Jak a-b w scenariuszu głównym
2. System wyświetla informację, o tym, że nie ma logów dla pracownika w podanym przedziale czasowym.

3.21 Generowanie raportu dotyczącego projektu

Nazwa: UC21 Generowanie raportu dotyczącego projektu

Opis: Proces generowania raportu dotyczącego projektu

Aktorzy: Kierownik projektu

Warunki początkowe:

- W systemie znajdują się logi umieszczone w zadaniach przypisanych do projektów

Warunki końcowe:

- Raport zapisany pliku na dysku komputera.

Scenariusz główny:

1. Aktor wybiera z listy projekt.
2. Aktor podaje przedział czasowy.
3. System wyświetla wszystkie logi dla projektu dla zadanego przedziału czasowego.

4. Aktor wybiera generowanie raportu.
5. System generuje raport i zapisuje go na dysku komputera.
6. System potwierdza wygenerowanie raportu.

Scenariusz alternatywny - brak logów dla projektu:

1. Jak a-b w scenariuszu głównym
2. System wyświetla informację, o tym, że nie ma logów dla projektu w podanym przedziale czasowym.

3.22 Ostrzeganie przed sytuacjami krytycznymi w projekcie

Nazwa: UC22 Ostrzeganie przed sytuacjami krytycznymi w projekcie

Opis: Ostrzeganie przed sytuacjami krytycznymi w projekcie

Aktorzy: Kierownik projektu

Warunki początkowe:

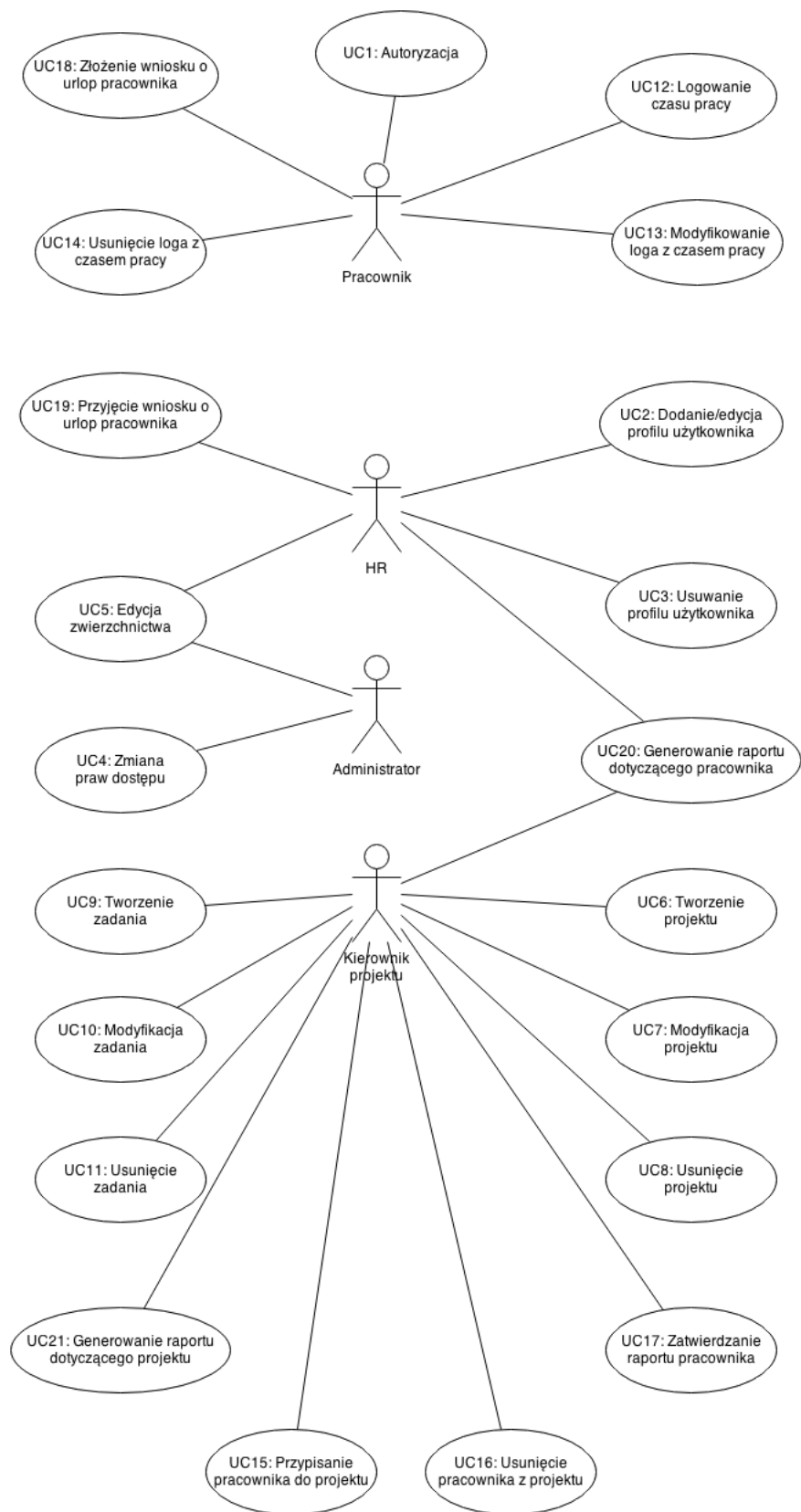
- W projekcie nastąpił problem krytyczny (przekroczenie terminu wykonania, przekroczenia kosztów).

Warunki końcowe:

- W systemie jest informacja o tym, że kierownik projektu otrzymał stosowny komunikat.

Scenariusz główny:

1. System wyświetla komunikat o sytuacji krytycznej.
2. Aktor potwierdza przeczytanie komunikatu.
3. System wyświetla standardowy widok.

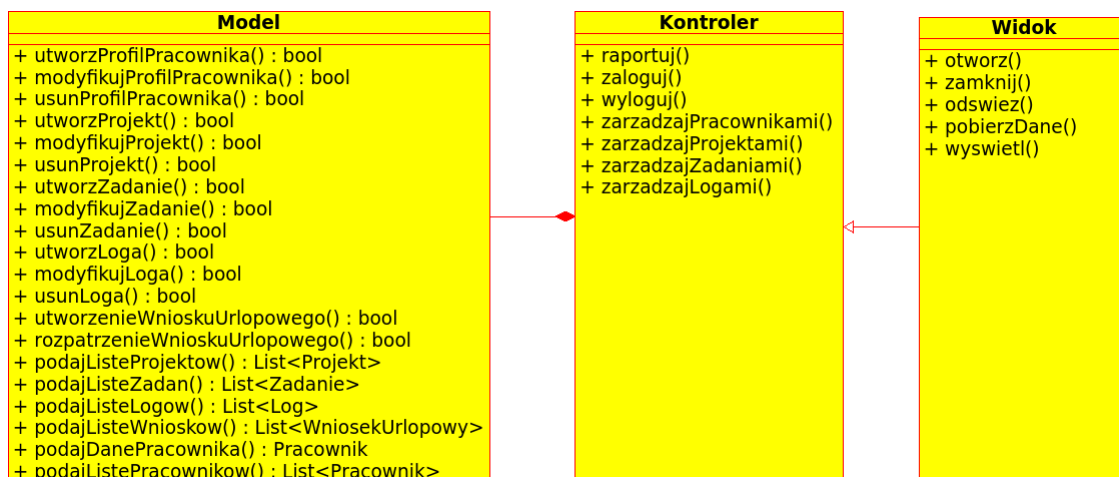


Rysunek 2: Diagram przypadków użycia

4 Architektura systemu

4.1 Diagramy klas

Zdecydowaliśmy się na wzorzec architektoniczny MVC (ang. Model View Controller). Jego zastosowanie umożliwia łatwą modyfikowalność, któregoś z komponentów, bez konieczności zmian w pozostałych. Zgodnie z tym wzorcem w modelu zgromadzone są wszystkie dane naszego systemu. Widok odpowiada za prezentację aktualnego stanu systemu użytkownikowi, a kontroler odpowiada za zmianę modelu, a także odświeżanie widoków. Na rysunku 3 przedstawiono trzy części wzorca MVC naszego systemu.



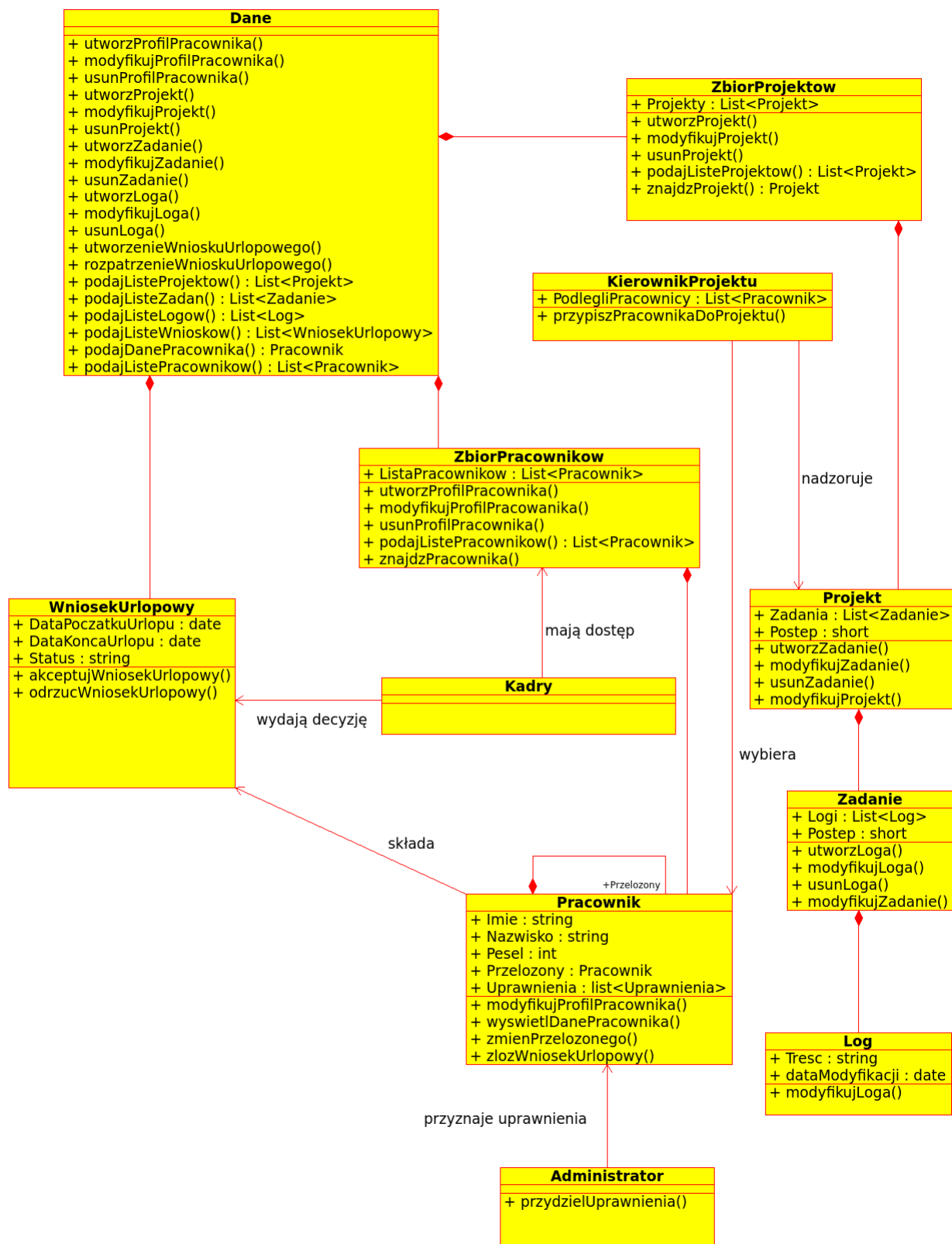
Rysunek 3: Schemat architektury systemu zgodnej ze wzorcem architektonicznym MVC

Na rysunku 4 przedstawiono szczegółowy diagram klas dla modelu. Projekt składa się z zadań składających się z logów, posiadających treść. Każdy projekt ma kierownika projektu. Pracownik może złożyć wniosek urlopowy, a kadry się do niego ustosunkować (przyjąć bądź odrzucić).

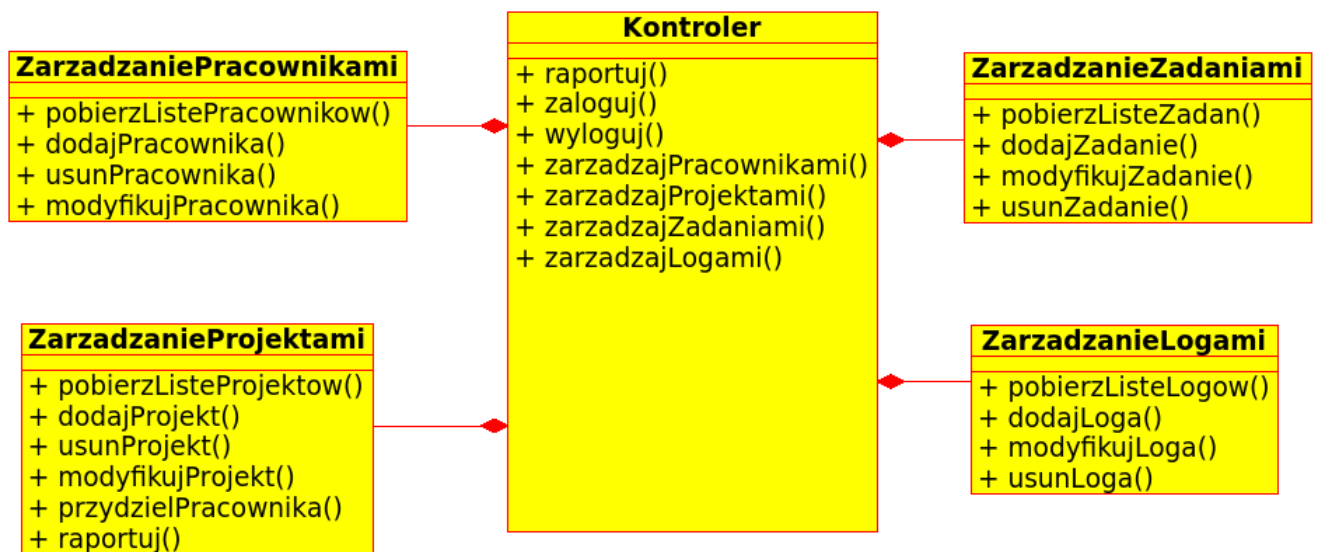
Rysunek 5 prezentuje diagram klas dla kontrolera. Zadaniem kontrolera jest zarządzanie: pracownikami, projektami, zadaniami i logami.

Rysunek 6 przedstawia diagram klas dla okna. Okno jest odpowiedzialne za prezentowanie okien: logowania, raportu, zarządzania logami, zarządzania zadaniami, zarządzania projektami i pracownikami.

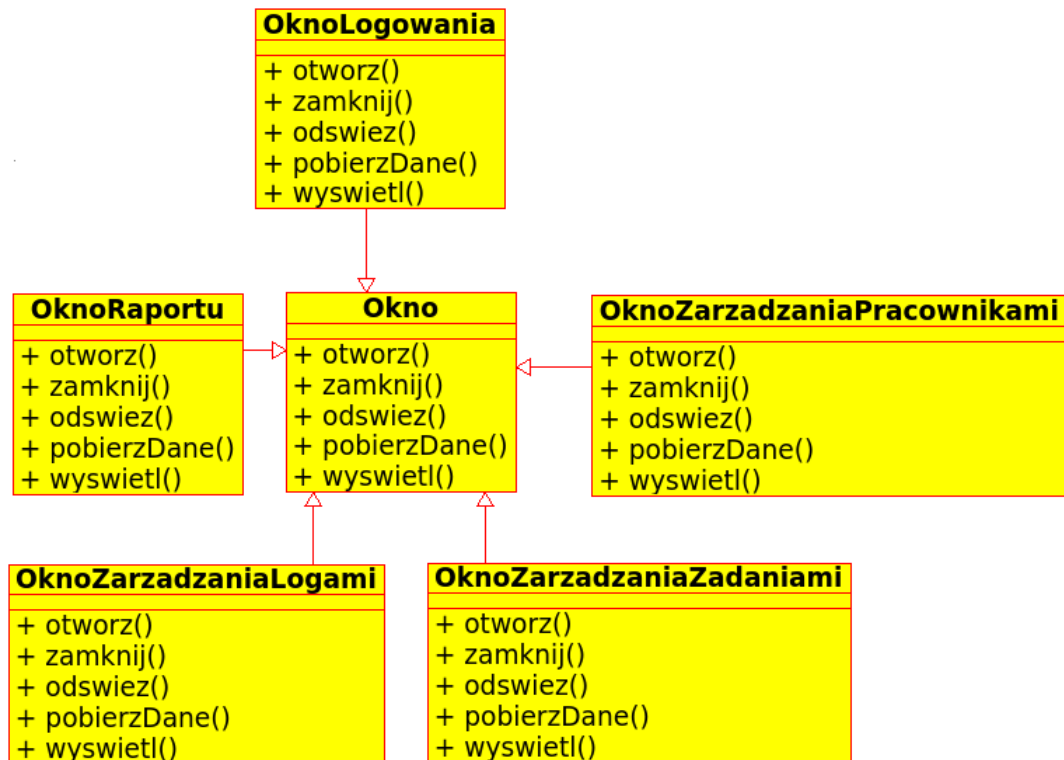
Zatwierdzanie loga przez kierownika projektu, opis relacji, dodać okno zarządzania pracownikami



Rysunek 4: Diagram klas dla modelu.



Rysunek 5: Diagram klas dla kontrolera.



Rysunek 6: Diagram klas dla okna

5 Specyfikacja techniczna

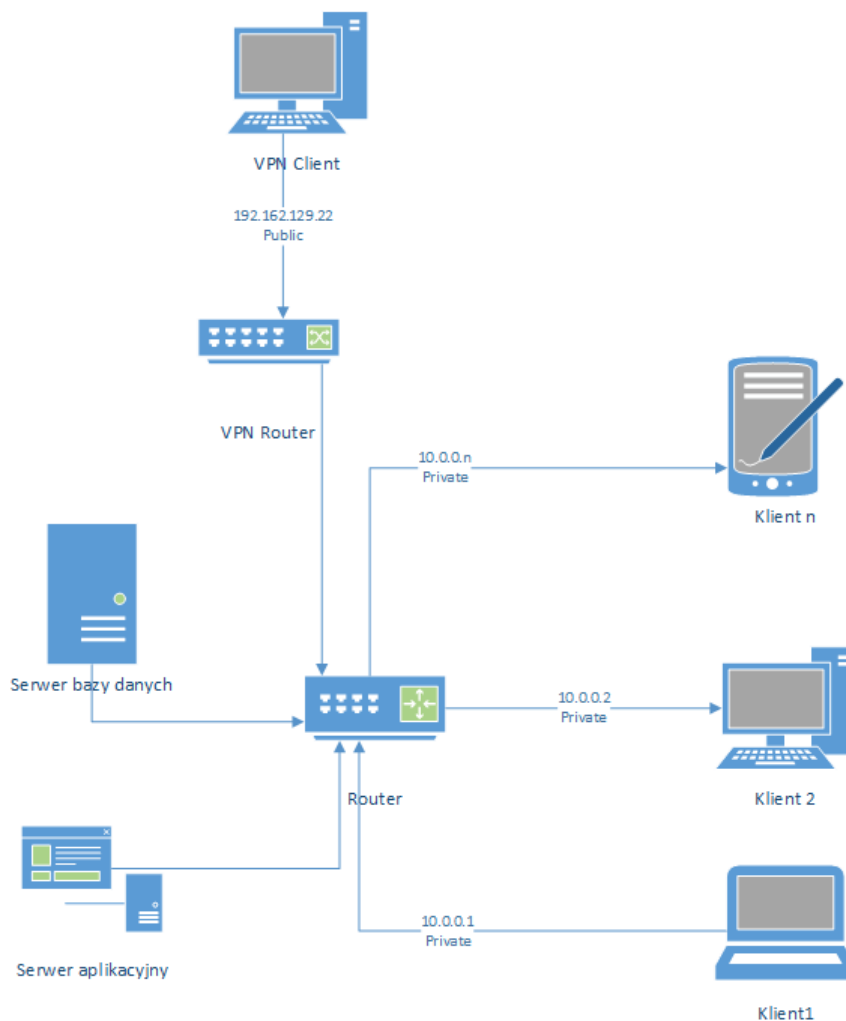
W poniższym rozdziale zostaną przedstawione detale techniczne implementacji systemu oraz szczegóły wdrożenia. Zawiera on fizyczny opis infrastruktury niezbędnej do poprawnego działania systemu, oraz diagramy obrazujące rozmieszczenie głównych komponentów aplikacji.

5.1 Wstępna specyfikacja sprzętu i oprogramowania podstawowego

1. System w wersji podstawowej będzie składał się z 2 serwerów.
 - serwer aplikacji zawierający: backendową część aplikacji, serwer webowy udostępniający usługę, oraz aplikację frontendową
 - serwer przechowujący bazę danych

Każdy z serwerów będzie posiadał zainstalowany i skonfigurowany system Linux.

2. Ze strony klienckiej wykorzystywana jest jedynie przeglądarka
3. Komunikacja między serwerami oraz urządzeniami klientów jest zapewniona przez router sieci lokalnej z brakiem możliwości dostępu z zewnętrznej sieci
4. Opcjonalnie możliwe jest użycie aplikacji z adresu zewnętrznego przez skorzystanie z usługi VPN



Rysunek 7: Schemat infrastruktury sieci

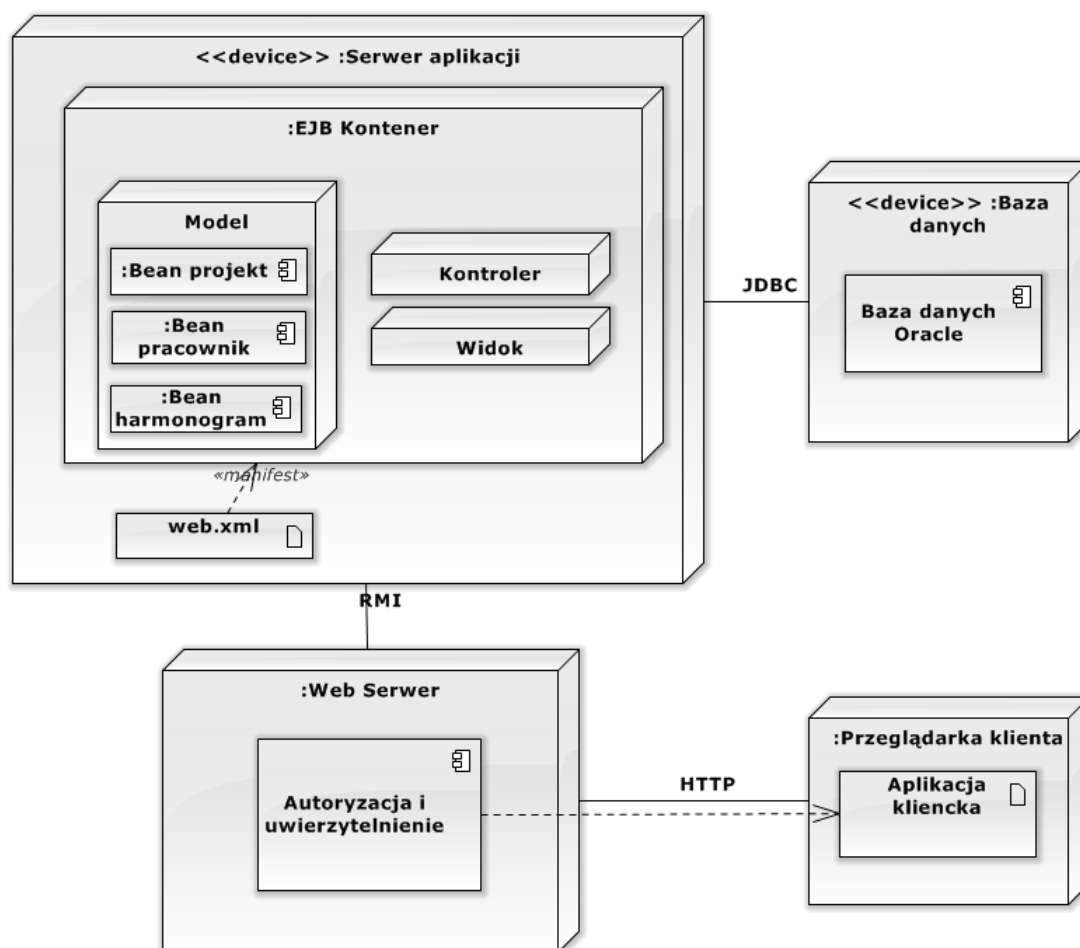
5.2 Specyfikacja technologii realizacji oprogramowania systemu

System zostanie stworzony za pomocą wyszczególnionych w podrozdziale technologii.

1. Aplikacja zostanie zrealizowana za pomocą technologii Java EE z użyciem frameworka Spring, który umożliwia tworzenie systemów zgodnie z architekturą MVC.
2. Do przechowywania danych została wybrana relacyjna baza danych, aby zapewnić łatwe generowanie analiz. Zostanie zrealizowana w technologii Oracle Database 11g. Jednak interfejs aplikacji ma umożliwiać również zamiennie dołączanie bazy Postgres 9.4.
3. Frontend aplikacji klienckiej zostanie zrealizowany w technologii HTML5, CSS 3.0 oraz JavaScript
4. Aplikacja kliencka musi działać i być wyświetlana poprawnie pod przeglądarkami:
 - Chrome (w wersji 8 i wyżej)

- Opera (w wersji 15 do wersji 19)
- Firefox (w wersji 29 do wersji 34)
- Internet Explorer (w wersji 8 i wyżej)

5. Jako serwer webowy zostanie użyty Apache Tomcat 7



Rysunek 8: Schemat wdrożenia

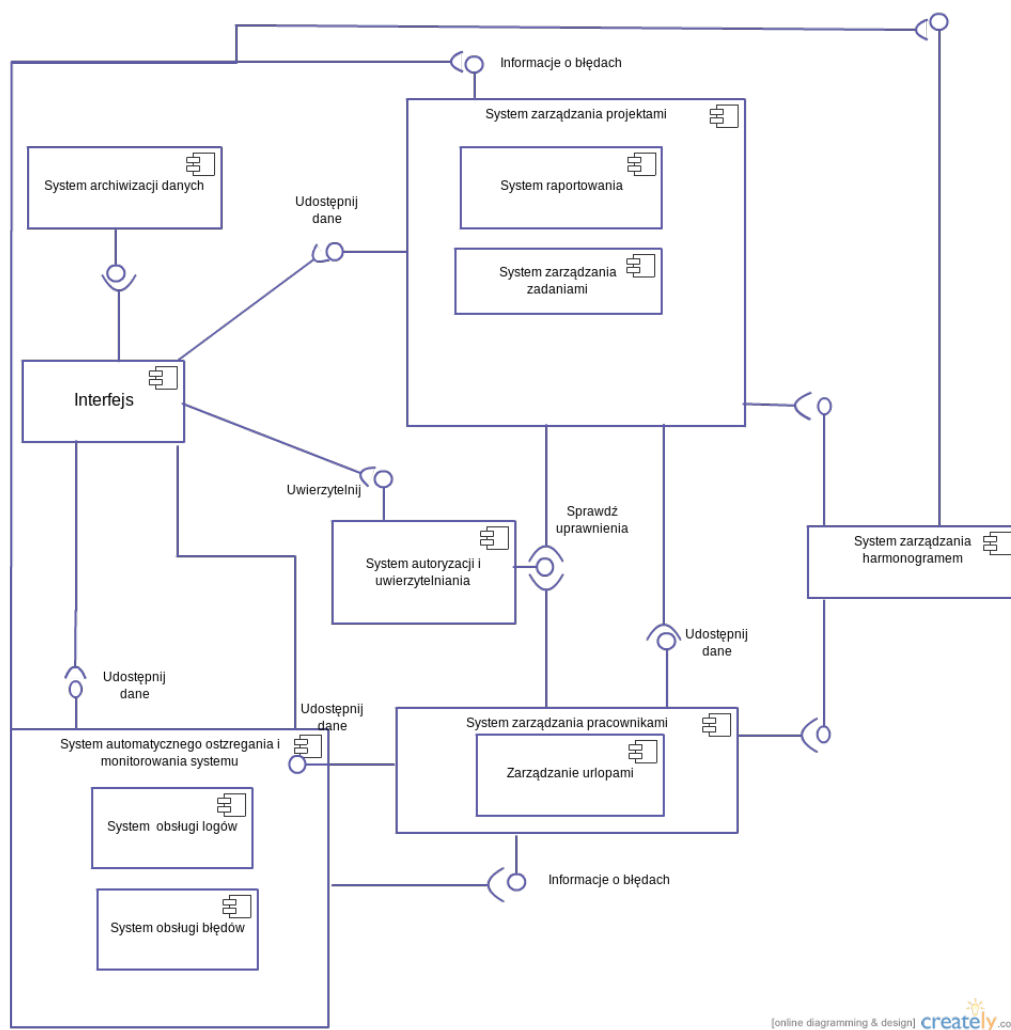
5.3 Specyfikacja parametrów sprzętu

Do stworzenia architektury zostaną użyte komputery o następujących parametrach:

- Dell PowerEdge T320 with x8 HDD Backplane
- Procesor - Intel Quad-Core Xeon E5-2403 (1.80 GHz, 10 M Cache, 6.4 GT/s QPI, 80W)
- RAM - 16GB DDR3 1333 MHz Dual Rank x4 LV RDIMM (2*8GB)
- Dysk twardy - 4x2TB 7200 rpm 3,5

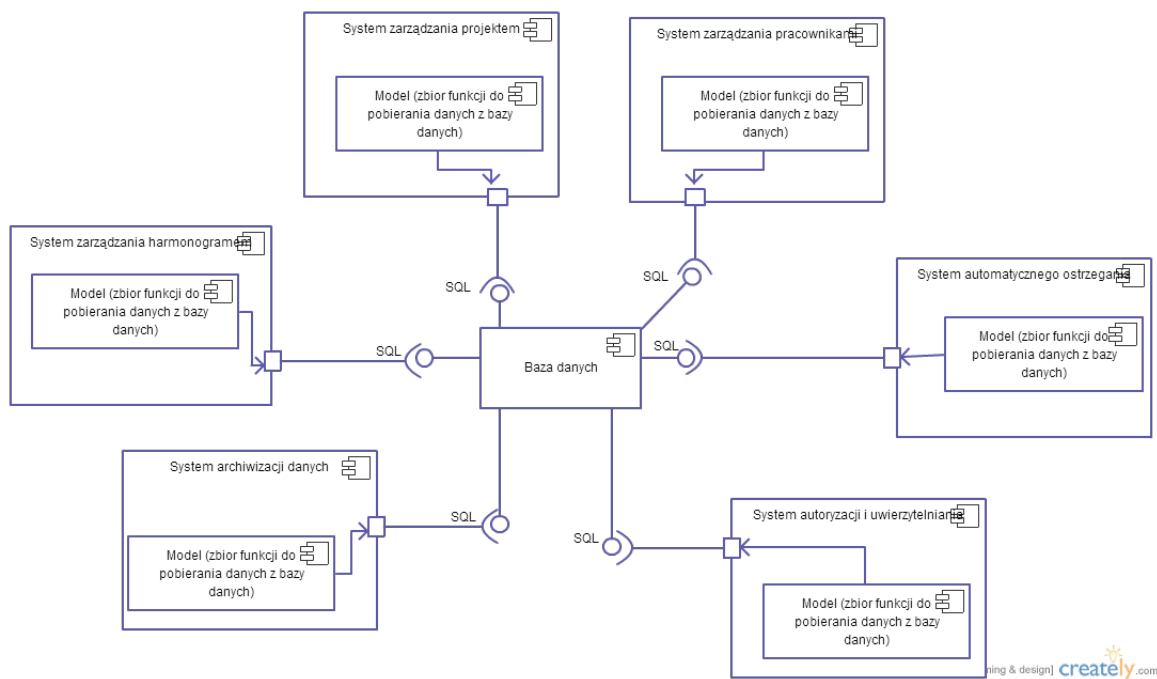
5.4 Diagram komponentów

Diagram komponentów przedstawia podział całego systemu na mniejsze podsystemy. Komponent jest to wymienny, wykonywalny fragment systemu. Zależności między komponentami są przedstawiane w postaci interfejsów tzn. jeden komponent może korzystać z funkcji jakie udostępnia inny komponent. Na rysunku 7 przedstawiony jest diagram komponentów projektowanego systemu



Rysunek 9: Diagram komponentów

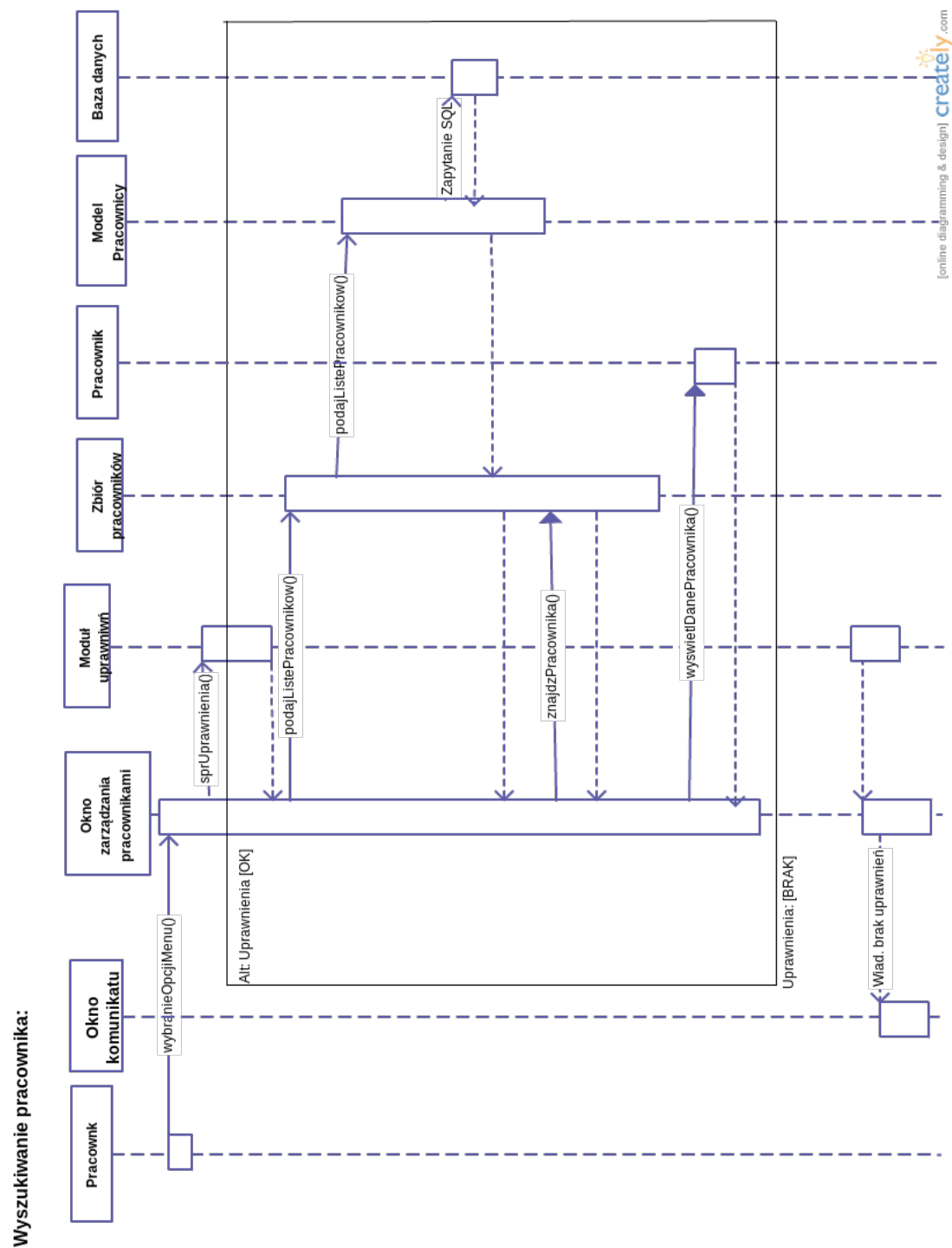
Diagram na rysunku 7 nie pokazuje pełnej struktury wewnętrznej wyszczególnionych podsystemów. Pokazanie zostały tylko najważniejsze komponenty tak aby rysunek pozostał czytelny. Na drugim diagramie komponentów 8 przedstawiono w sposób bardziej szczegółowy to, że pod-systemy które korzystają z bazy danych zawierają w sobie komponenty (modele) których zadaniem jest komunikacja z bazą danych. Taka budowa systemów wynika z architektury MVC.



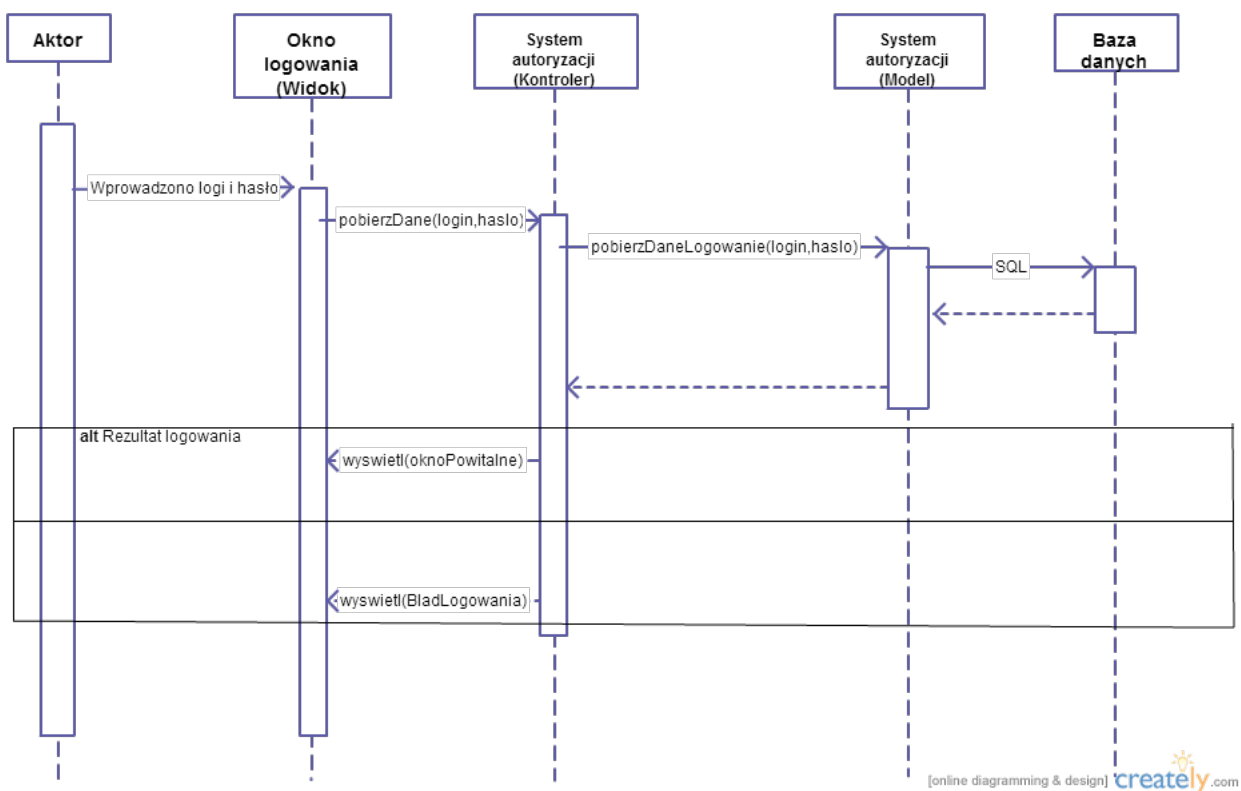
Rysunek 10: Diagram komponentów: dostęp do bazy danych

5.5 Diagramy sekwencji

W tym podrozdziale przedstawiono wybrane diagramy sekwencji. Diagram sekwencji obrazuje interakcje pomiędzy częściami systemu w postaci sekwencji komunikatów np. wywołań funkcji wymienianych między nimi.



Rysunek 11: Diagram sekwencji: Wyszukiwanie pracownika



Rysunek 12: Diagram sekwencji: Logowanie do systemu