

UNIwersYTET WARMIŃSKO – MAZURSKI W OLSZTYNIE

Dokumentacja projektowa

“Ruby Team”

Mateusz Kaczyński

Jarosław Giziński

Arkadiusz Jasiński

Marek Jarzębowski

Olsztyn 14.04.14r

1. Opis systemu, wymagania funkcjonalne i нефункционалне. – Jarosław Giziński
2. Użytkownicy, słownik, lista funkcji, lista przypadków użycia, użyteczność (określenie kryteriów testów). – Marek Jarzębowski
3. Diagram hierarchii funkcji, diagram przypadków użycia. - Arkadiusz Jasiński
4. Projekt relacyjnej bazy danych, diagram klas, opis atrybutów i metod.

1. Opis systemu, wymagania funkcjonalne i нефункционалне.

1.1 Opis systemu.

Trollo to system do zarządzania czasem przy różnego rodzaju projektach (najczęściej zespołowych). Mamy tu podział na dwa typy kont – administratorskie (generowane podczas instalacji systemu) i konto użytkownika, które może założyć administrator dla innych osób. Narzędzie to pozwala na dodawanie projektów, które z kolei możemy przypisywać do danego namespace'u. Do każdego projektu możemy dodawać dowolną ilość zadań, które zostają przypisane do jednego lub kilku użytkowników. Ci z kolei mogą rejestrować czas, który poświęcają do wykonania danego zadania. Trollo pozwala także na dodawanie komentarzy do zadań oraz ładowania plików związanych z wykonaniem danego zadania na serwer.

Graficzny typ interfejsu z wykorzystaniem frameworka bootstrap oraz kontrolek środowiska HTML obsługiwanych przez JavaScript i Ruby zapewnia łatwe w obsłudze oraz intuicyjne zarządzanie kontem.

1.2 Wymagania funkcjonalne.

- zarządzanie projektami
- przypisywanie projektów do namespace'ów
- zarządzanie zadaniami
- zarządzanie czasem w zadaniach
- komentowanie zadań
- dodawanie plików do zadań
- logowanie, uwierzytelnianie użytkownika

1.3 Wymagania niefunkcjonalne.

- komputer wyposażony w środowisko APACHE (lub inny serwer lokalny), RUBY, MYSQL
- przeglądarka internetowa z obsługą języka JAVASCRIPT
- wolne miejsce na dysku
- uprawnienia do korzystania z systemu

2. Użytkownicy, słownik, lista funkcji, lista przypadków użycia, użyteczność.

2.1 Użytkownicy:

- Administrator
- Użytkownik

2.2 Słownik:

Namespace- przestrzeń/katalog w której będą znajdowały się projekty

Projekt- Zbiór plików i danych charakteryzujące prace nad danym „problemem” zebrane w całość. Cechy : nazwa projektu, opis, status(otwarty, zamknięty), czas prac nad projektem, załączniki (dokumenty, pliki), przewidywany czas zakończenia projektu.

Zadanie- opis zadania do wykonania. Zadanie jest przydzielane do użytkownika który ma je wykonać w ramach pracy nad danym projektem.

Cechy: szacowany czas wykonania zadania, przydział do użytkownika, status zadania(ukończone / w trakcie pracy), czas spędzony nad zadaniem

2.3 Lista funkcji

Lista funkcji administratora:

- Zarządzanie dokumentami projektu (CRUD)**
- Zarządzanie namespace'ami (CRUD)**
- Zarządzaj projektami (CRUD)**
- Przydzielenie użytkowników do projektów**
- Zarządzanie komentarzami projektów(CRUD)**
- tworzenie dokumentacji danego projektu**
- Zarządzanie użytkownikami systemu (CRUD)**
- zarządzanie zadaniami w projektach (CRUD)**
- CRUD zadań**
- ustawienie czasu na wykonanie zadania**
- oznaczenie statusu zadania –ukończone / w trakcie prac**
- przydział użytkowników do zadań oraz ich edycja**

Lista funkcji użytkownika

- dodawanie komentarzy do zadań**
- sprawdzenie listy przydzielonych zadań**
- Dodanie czasu spędzonego na wykonywaniu zadania**
- Dodanie dokumentu / pliku do zadania**
- Wyświetlenie listy plików w projekcie**
- Edycja dokumentów**

2.3 Lista przypadków użycia

- Autoryzacja**
- Zarządzanie użytkownikami**
- Zarządzanie namespace'ami**
- Zarządzanie projektami**
- Zarządzanie komentarzami**
- Zarządzanie zadaniami w projekcie**
- Zarządzanie dokumentami projektu**

2.4 Użyteczność / testy

Norma składowa: Bezpieczeństwo

a) Opis: Dane w systemie nie mogą być zmienione lub usunięte przez nieupoważnionego użytkownika.

b) Miara: Punkty z przedziału 1 lub 10 (10 to ocena najlepsza, 1 najgorsza)

c) Metryka

variant 1: Test wykonywany jest przez testera, który jest zalogowany w systemie jako użytkownik, stara się przejąć uprawnienia administratora oraz wprowadzić zmiany w bazie danych lub ustawieniach kont(ustawienia które są docelowo przeznaczone tylko dla administratora). Jeśli uda dokonać zmian w bazie danych system otrzyma 1 pkt, jeśli nie system otrzyma 10 pkt.

variant 2: Test wykonywany jest przez testera, który nie jest zalogowany w systemie, stara się przejąć uprawnienia administratora lub użytkownika oraz wprowadzić zmiany w bazie danych lub ustawieniach kont(ustawienia które są docelowo przeznaczone tylko dla zalogowanych). Jeśli uda dokonać zmian w bazie danych system otrzyma 1 pkt, jeśli nie system otrzyma 10 pkt.

Norma składowa: Czas reakcji systemu

a) Opis: Niezbędny wymóg dla tego typu systemów czasu rzeczywistego, aby system działał sprawnie i komfortowo dla użytkowników.

b) Miara: Punkty z przedziału 1-5 (5to ocena najlepsza - im więcej, tym system jest szybszy)

Czas reakcji systemu na polecenie użytkownika.

1 - powyżej 1s

3 - 1-0,5s

6- 0,5-0,2s

8 - 0,2-0,1s

10- poniżej 0,1s

c) Metryka: Test wykonywany jest przez testera, który jest zalogowany jako użytkownik/administrator , tester ocenia system pod względem szybkości działania z punktu widzenia zalogowanego użytkownika/administratora.

Norma składowa: Intuicyjność systemu.

- a) Opis:** Określenie, czy system jest intuicyjny, prosty i łatwy w użyciu
- b) Miara:** Punkty z przedziału 1-10 (10 to ocena najlepsza - im więcej, tym system jest bardziej intuicyjny)
- c) Metryka:** Test wykonywany jest przez testera który ma korzystać z systemu i przeprowadzić próbę funkcjonalności aplikacji. Tester wystawia oceny wg. własnego uznania i własnych spostrzeżeń.

Norma składowa: Kolorystyka aplikacji

- a) Opis:** Odpowiednio dobrane kolory, możliwość długiego korzystania z systemu bez znacznego zmęczenia oczu bólu głowy
- b) Miara:** Punkty z przedziału 1-10 (10 to ocena najlepsza - im więcej, tym system posiada lepszą kolorystykę)
- c) Metryka:** Tester ma za zadanie korzystać z aplikacji przez godzinę. Po zakończonym zadaniu ma ocenić w skali (1-10) jak ocenia kolorystykę aplikacji i czytelność interfejsu w zależności od odczucia zmęczenia.

3. Diagram hierarchii funkcji, Diagram przypadków Użycia

3.1. Diagram hierarchii funkcji

Legenda:

✓ - pełny dostęp - Użytkownik ma pełny dostęp do funkcji

✓ - ograniczony dostęp - Użytkownik może zarządzać wyłącznie danymi stworzonymi przez siebie lub w danym obrębie(np. "Zwykły" Użytkownik może dodawać komentarze wyłącznie w projektach, do których został przydzielony)

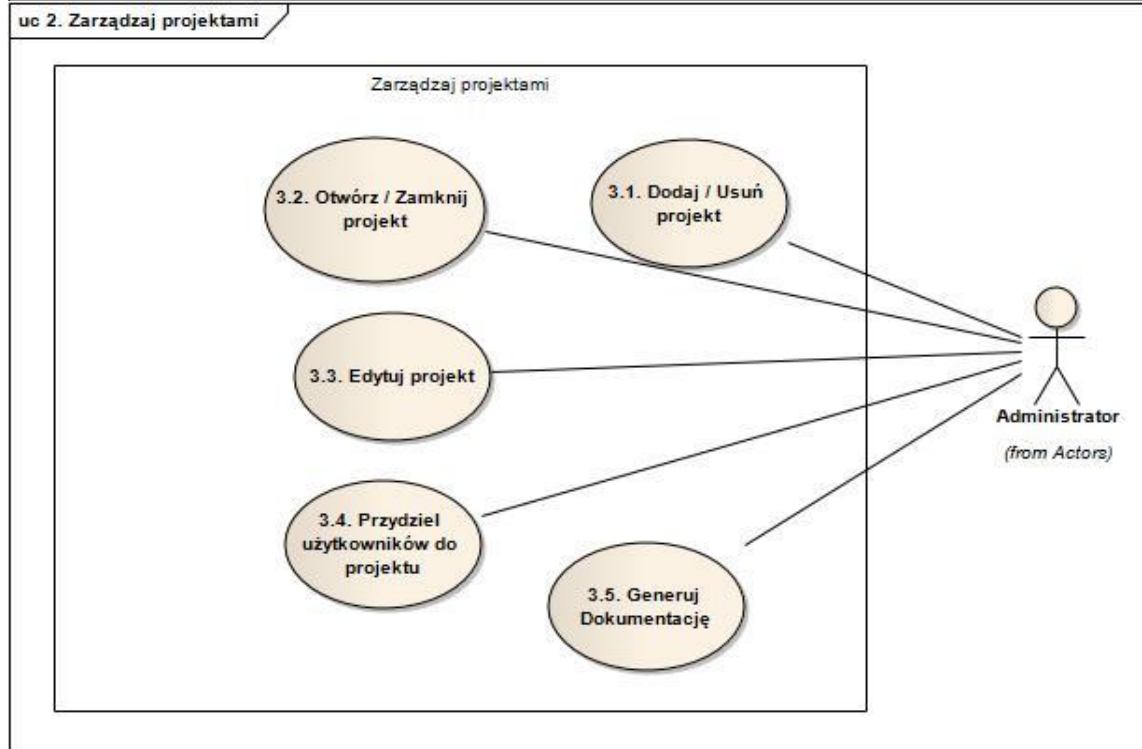
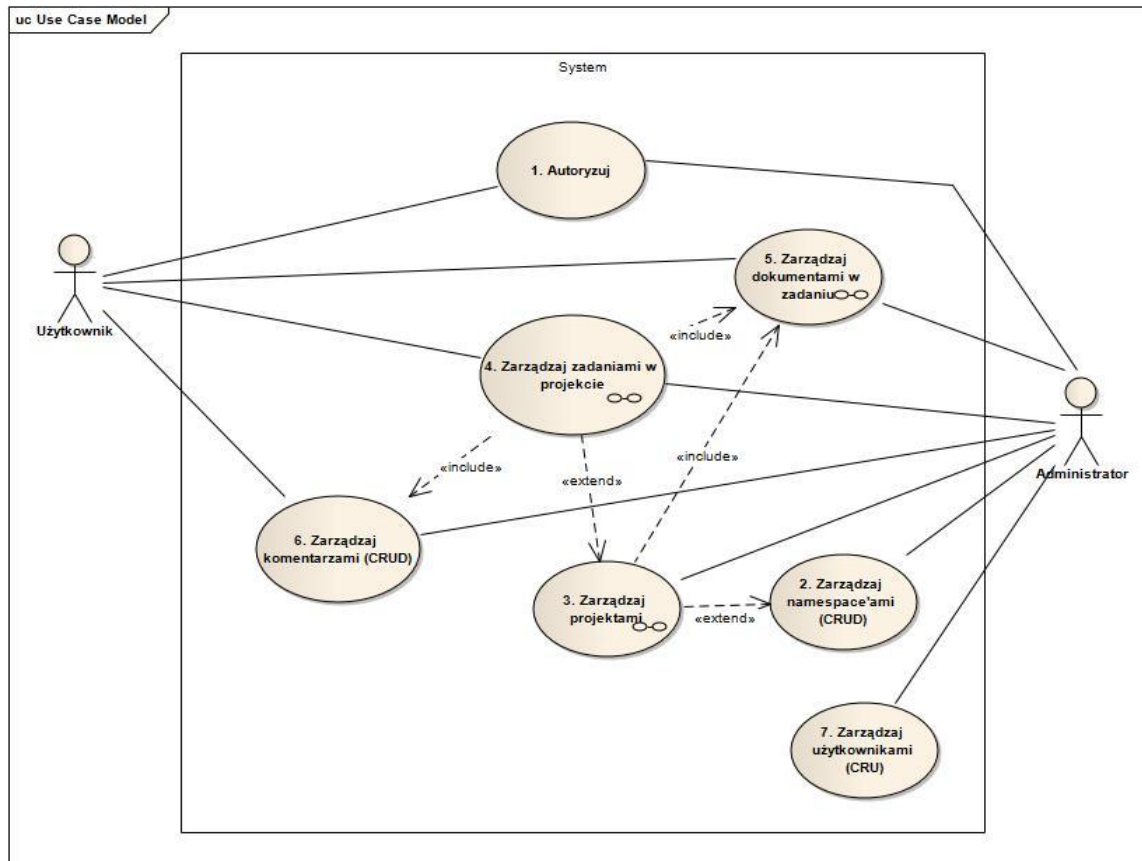
X - brak dostępu - Dany użytkownik nie ma dostępu do danej funkcji

	Administrator				Użytkownik			
	C	R	U	D	C	R	U	D
Zarządzanie użytkownikami	✓	✓	✓ ¹	✓ ²	X	✓	✓	✓
Zarządzanie namespace'ami	✓	✓	✓	✓	X	✓	X	X
Zarządzanie projektami	✓	✓	✓	✓	X	✓	X	X
Przydzielanie użytkowników do projektu	✓	✓	✓	✓	X	X	X	X
Zarządzanie zadaniami w projekcie	✓	✓	✓	✓	X	✓	X	X
Przydzielanie zadań do użytkowników	✓	✓	✓	✓	X	X	X	X
Zarządzanie komentarzami	✓	✓	✓	✓	✓	✓	✓	✓
Zarządzanie czasem spędzonym nad zadaniem	✓	✓	✓	✓	✓	✓	✓	✓
Zarządzanie dokumentami projektu	✓	✓	✓	✓	✓	✓	✓	✓

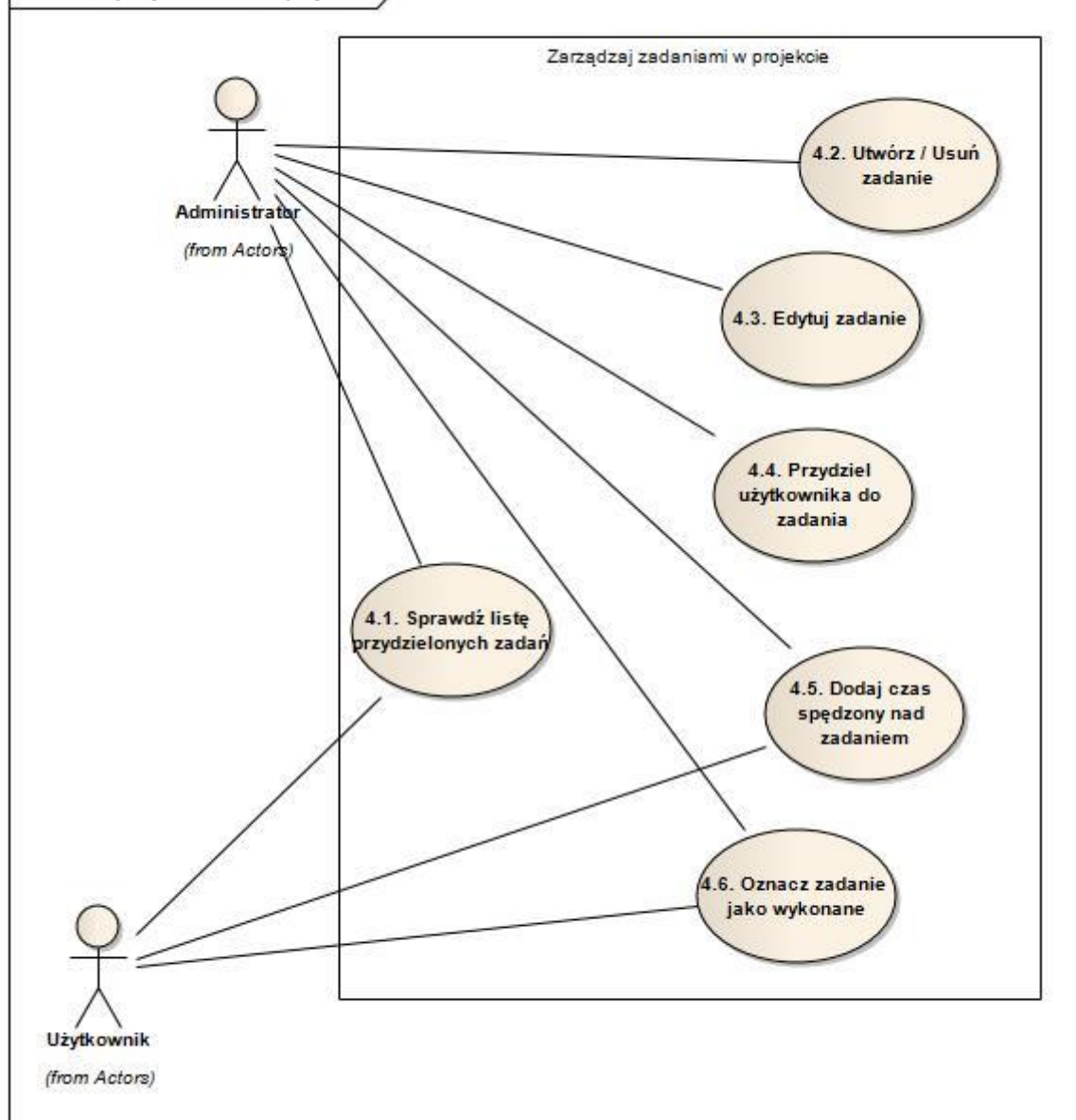
1 - Administrator nie może edytować ani odebrać uprawnień innemu Administratorowi, może nadawać Użytkownikom uprawnienia Administratora

2 - Administrator może usuwać wyłącznie Użytkowników

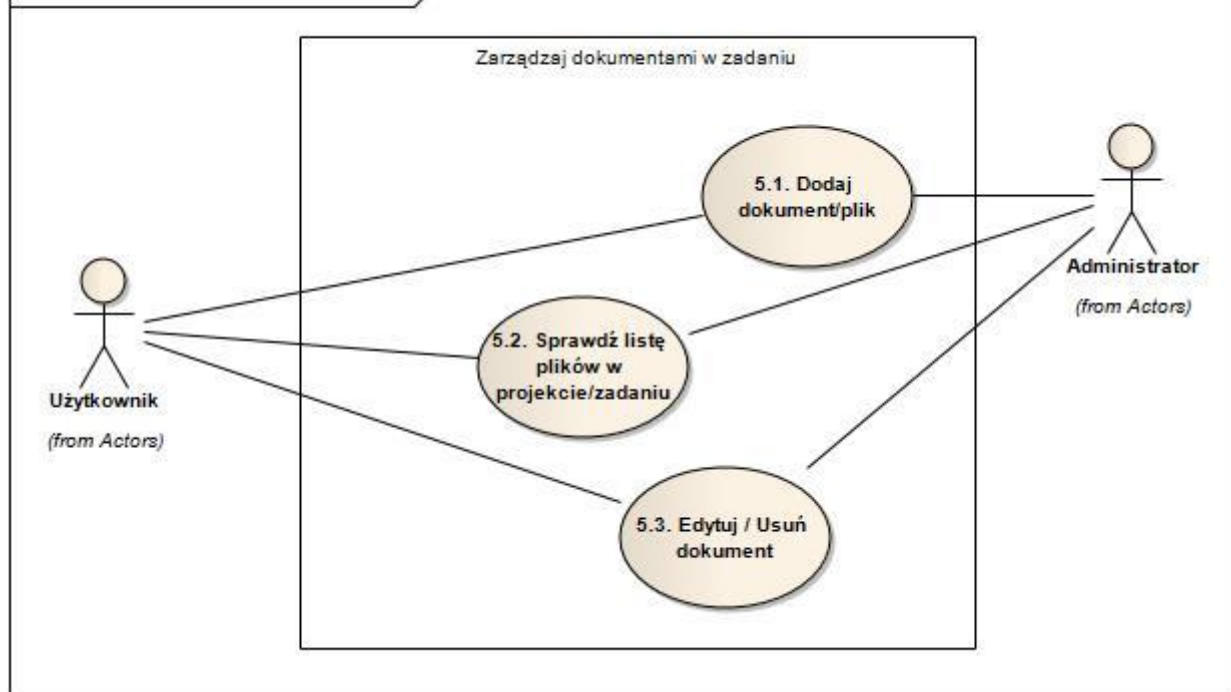
3.2. Diagram Przypadków Użycia



uc 3. Zarządzaj zadaniami w projekcie

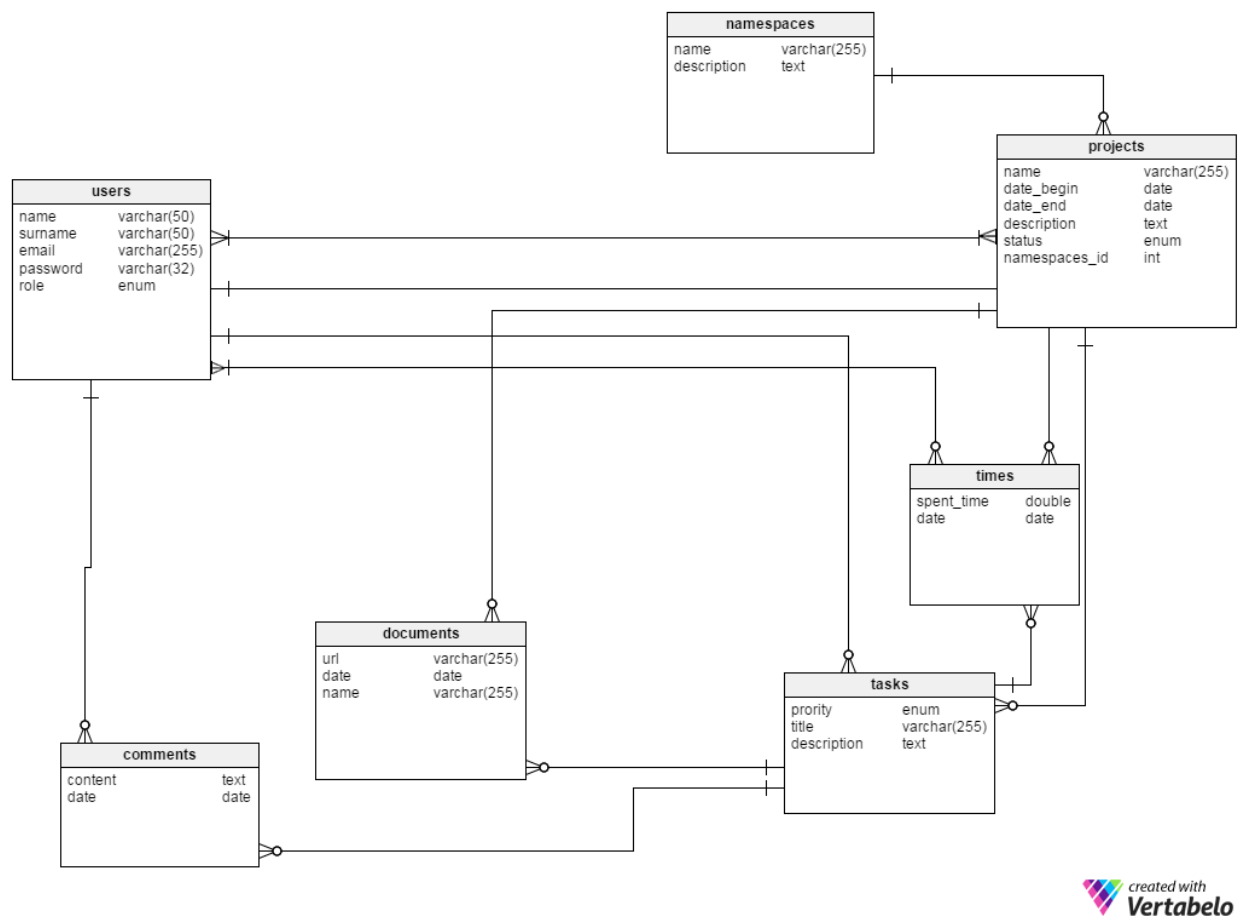


uc 6. Zarządzaj dokumentami w zadaniu



4. Projekt relacyjnej bazy danych, diagram klas, opis atrybutów.

4.1 Diagram klas



4.2 Opis atrybutów

1. users

- a) name – imię użytkownika
- b) surname – nazwisko użytkownika
- c) email – email użytkownika, wykorzystywany również do logowania
- d) password- hasło użytkownika
- e) role

2. projects

- a) name-nazwa projektu
- b) date_begin- data rozpoczęcia projektu
- d) date_end- data zakończenia projektu
- e) description- opis projektu
- f) status-status projektu(zakończony, rozpoczęty)

3. namespaces

- a) name- nazwa namespaceu(np. „projekty dla firmy”, „projekty klienta”)
- b) description- opis namespaceu (np. “Ten namespace zawiera wszystkie projekty dla danego klienta”)

4. times

- a) spent_time- spędzony czas nad zadaniem
- b) date- data kiedy został dodany czas

5. tasks

- a) priority- priorytet zadania(niski, normalny, wysoki)
- b) title- tytuł zadania
- c) description – opis zadania

6. documents

- a) url- link do dokumentu, zdjęcia itp.
- b) date- data dodania dokumentu
- c) name- nazwa dokumentu

7. comments

- a) contents- treść komentarza
- b) date- data dodania komentarza

4.3 Projekt relacyjnej bazy danych

