

**Politechnika Wrocławska**  
**Wydział Informatyki i Telekomunikacji**

---

Kierunek: **Informatyka techniczna**  
Specjalność: **Inżynieria systemów informatycznych**

**PRACA DYPLOMOWA**  
**INŻYNIERSKA**

**Aplikacja internetowa do gromadzenia  
i udostępniania informacji o  
technologiach programistycznych**

**A web application for gathering and  
sharing information about software  
development**

Kamil Dywan

Opiekun pracy  
dr inż, Paweł Rogaliński

## **Streszczenie**

**Słowa kluczowe:**

## **Abstract**

**Keywords:**

# Spis treści

<b>1. Wstęp . . . . .</b>	<b>8</b>
1.1. Wprowadzenie . . . . .	8
1.2. Cel i zakres pracy . . . . .	8
1.3. Układ pracy . . . . .	8
<b>2. Rodzina/Podział technologii . . . . .</b>	<b>9</b>
2.1. Podział technologii ze względu na ich architekturę . . . . .	9
2.2. Podział technologii ze względu na ich zastosowania . . . . .	9
<b>3. Architektura systemu . . . . .</b>	<b>11</b>
3.1. Baza danych . . . . .	11
3.2. Aplikacja serwerowa - Backend . . . . .	11
3.3. Klient - Frontend . . . . .	12
3.4. Warstwa bezpieczeństwa . . . . .	12
3.5. REST . . . . .	12
<b>4. Wymagania funkcjonalne . . . . .</b>	<b>13</b>
<b>5. Wymagania нефunkcjonalne . . . . .</b>	<b>15</b>
<b>6. Diagram przypadków użycia . . . . .</b>	<b>16</b>
<b>7. Podsumowanie . . . . .</b>	<b>19</b>
<b>Literatura . . . . .</b>	<b>20</b>
<b>A. Instrukcja wdrożeniowa . . . . .</b>	<b>21</b>
<b>B. Opis załączonej płyty CD/DVD . . . . .</b>	<b>22</b>

# Spis rysunków

3.1. Architektura systemu . . . . .	11
6.1. Cały diagram przypadków użycia . . . . .	16
6.2. Diagram przypadków użycia o wpisów . . . . .	17
6.3. Diagram przypadków użycia dotyczący opinii o wpisach i technologiach . . . . .	17
6.4. Diagram przypadków użycia zawierający resztę funkcjonalności . . . . .	18

# Spis tabel

# Spis listingów

# Skróty

**GUI** (ang. *graphical user interface*)

# **Rozdział 1**

## **Wstęp**

**1.1. Wprowadzenie**

**1.2. Cel i zakres pracy**

**1.3. Układ pracy**



# Rozdział 2

## Rodzina/Podział technologii

### 2.1. Podział technologii ze względu na ich architekturę

- Technologie informatyczne
  - Język
    - \* Programowania (np. Java, C++)
      - Biblioteka (np. SFML, SDL, OpenGL)
      - Framework (np. Spring, Spring Boot, React, Angular)
    - \* Znaczników (np. TeX, HTML, XML)
    - \* Zapytań (bazy danych – np. SQL, GraphQL)
      - Relacyjne (np. MySQL, Oracle Database, SQLite)
      - Obiektowo-relacyjne (PostgreSQL)
      - NoSQL (MongoDB, Cassandra)
- Środowisko uruchomieniowe
  - System operacyjny (np. Windows, Linux)
  - Wysokopoziomowe (JVM, .NET, node.js)
- Narzędzia (oprogramowanie)
  - System kontroli wersji (np. git)
    - \* Serwisy hostujące gita (GitLab, GitHub)
  - CI/CD (np. Jenkins, GitLab CI)
  - Konteneryzacja (np. docker)
  - Orkiestracja (system do zarządzania, organizacji i planowania zasobów systemu – np. Docker Compose, Kubernetes)

### 2.2. Podział technologii ze względu na ich zastosowania

- Technologiczne
  - Architektura aplikacji
    - \* Rozproszona
    - \* Scentralizowana
  - Warstwa/Rola w systemie
    - \* Frontend
    - \* Backend

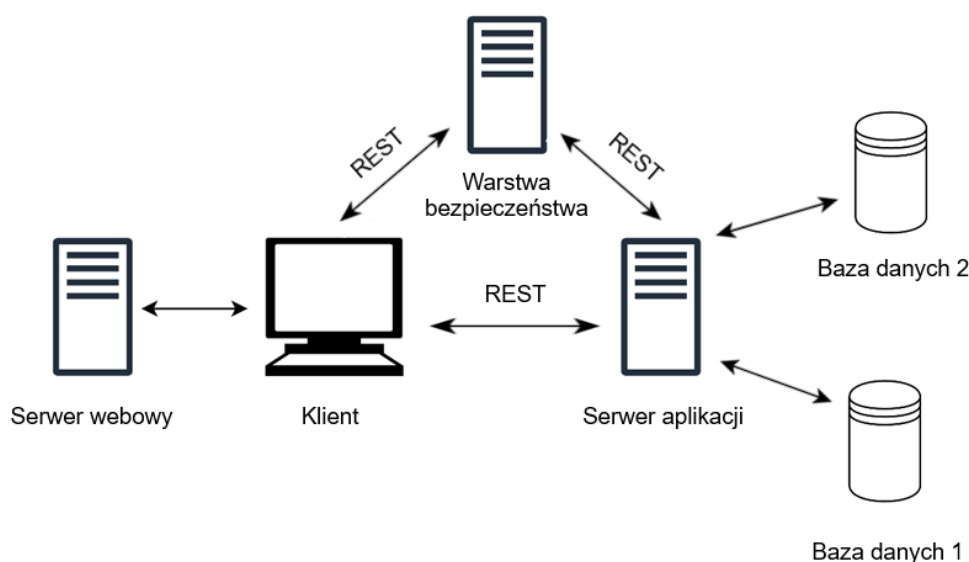
- \* Baza danych
- Typ aplikacji
  - \* Webowa
  - \* Mobilna
  - \* Desktopowa
- Poziom abstrakcji
  - Wysoki
  - Niski
- Dziedzina nauki
  - Sztuczna inteligencja
  - Informatyka
  - Matematyka
  - Fizyka
  - Chemia
  - Biologia
- Grupy odbiorców
  - Naukowcy
  - Edukacja
  - Rząd
  - Administracja
  - Wojsko
  - Motoryzacja
  - Przemysł
  - Korporacja

## Rozdział 3

# Architektura systemu

Realizowany system jest serwisem webowym, który w dużym uogólnieniu można opisać jako system typu klient-serwer. Klient (Klient) wysyła żądanie do serwera (Serwer aplikacji), a następnie serwer odpowiednio przetwarza otrzymane żądania i zwraca klientowi odpowiedź, którą to później odpowiedź klient interpretuje i przedstawia użytkownikowi (w tym przypadku jest to GUI interfejsu webowego).

Szczegółową architekturę systemu dobrze opisuje rysunek 3.1



Rys. 3.1: Architektura systemu

### 3.1. Baza danych

Baza danych to warstwa systemu odpowiedzialna za przechowywanie danych. Baza danych 1 jest główną bazą danych w systemie, a Baza danych 2 jest dodatkową bazą danych, która przechowuje informacje o zawartościach wpisów.

### 3.2. Aplikacja serwerowa - Backend

Backend jest odpowiedzialny za przyjmowanie żądań od klienta, odpowiednie przetwarzanie tych żądań, wykonywanie pewnych operacji na danych przechowywanych w bazie danych na

podstawie otrzymanych danych od klienta i przekazywanie klientowi adekwatnej odpowiedzi. Warstwa ta jest w ścisłym powiązaniu z warstwą bazy danych.

### 3.3. Klient - Frontend

Frontend jest odpowiedzialny za wysyłanie żądań do warstwy backendowej i następnie odpowiednie przetwarzanie oraz wyświetlanie danych otrzymanych w odpowiedzi od backendu. W tym przypadku frontend stanowi strona internetowa renderowana po stronie klienta.

### 3.4. Warstwa bezpieczeństwa

W systemie zostanie dodana warstwa bezpieczeństwa w postaci osobnego serwera, który odpowiada za uwierzytelnienie i autoryzację. Warstwa ta zabezpiecza zarówno warstwę kliencką poprzez blokowanie niektórych podstron, które nie powinny być dostępne dla danego użytkownika, jak i serwer aplikacji poprzez wymóg dostarczania prawidłowego tokenu przy wysyłaniu przez klienta każdego żądania do tego serwera.

Klient może uzyskać token potrzebny do wysyłania zapytań do serwera aplikacji poprzez zalogowanie się do serwera bezpieczeństwa. W przypadku pomyślnego logowania, klient otrzymuje od serwera bezpieczeństwa wygenerowany token, który to jest później przez klienta dostarczany w nagłówku `Authorization: Bearer <token>`. Opisany proces uwierzytelnienia nazywa się uwierzytelnianiem na okaziciela (ang. *Bearer authentication*).

Zastosowany serwer bezpieczeństwa umożliwi również autoryzację użytkowników w oparciu o ich rolę.

### 3.5. REST

Komunikacja w systemie między frontendem i backendem, frontendem i serwerem bezpieczeństwa oraz backendem i serwerem bezpieczeństwa odbywa się za pomocą REST. REST jest to sposób i format w jaki komunikuje się klient z serwerem. Serwer udostępnia klientowi punkty końcowe (end-pointy), do których klient może wysłać żądania http przesyłając przy tym jakieś dane np. tytuł wyszukiwanego wpisu. W skrócie komunikacja REST odznacza się następującymi cechami:

- bezstanowość,
- architektura klient-serwer,
- jednolity interfejs komunikacyjny – dzięki temu możliwe jest np. komunikowanie się systemów zaimplementowanych w różnych językach programowania,
- wykorzystywanie protokołu http.

W przypadku protokołu http można wyróżnić 4 podstawowe typy żądań:

- GET - pobranie zasobów,
- POST - wprowadzenie danych,
- PUT - aktualizacja zasobów,
- DELETE - usuwanie zasobów.

# Rozdział 4

## Wymagania funkcjonalne

- System musi zapewnić uwierzytelnianie oraz autoryzację w ramach 4 ról użytkownika:
  - Niezalogowany użytkownik,
  - Zwykły użytkownik,
  - Recenzent,
  - Administrator.
- System musi zapewnić rejestrację użytkownika
  - Podczas rejestracji użytkownik będzie podawał następujące informacje:
    - \* Nazwa użytkownika,
    - \* Imię,
    - \* Nazwisko,
    - \* Adres e-mail,
    - \* (Opcjonalnie) Awatar,
    - \* Hasło.
  - Będzie można dokonać rejestracji konta zwykłego użytkownika, recenzenta oraz administratora,
  - Rejestracja konta zwykłego użytkownika będzie mogła być wykonana przez użytkownika samodzielnie, a w przypadku kont recenzenta oraz administratora, konta te będą mogły być utworzone jedynie przez administratora.
- System musi zapewnić logowanie użytkownika
  - Podczas logowania użytkownik będzie wpisywał login lub e-mail oraz hasło,
  - Logowanie będzie dostępne dla zwykłego użytkownika, recenzenta oraz administratora,
  - Użytkownik powinien w tym samym momencie móc korzystać jedynie z uprawnień w ramach jednej posiadanej i wybranej przez niego roli,
  - Podczas logowania i po logowaniu powinna być dostępna możliwość wybrania jednej z posiadanych przez użytkownika ról.
- System musi zapewnić zwykłemu użytkownikowi, recenzentowi oraz administratorowi edycję wszystkich swoich danych, które podał podczas rejestracji,
- Wpisy
  - Wpisy będą się składać z następujących elementów:
    - \* Tytuł wpisu,
    - \* Autor wpisu,
    - \* Data utworzenia wpisu,
    - \* Data modyfikacji wpisu,
    - \* Kategoria technologii której dotyczy wpis,

- \* Zastosowanie technologii której dotyczy wpis,
- \* Autor lub autorzy technologii której dotyczy wpis,
- \* Ikona przedstawiająca technologię której dotyczy wpis,
- \* Część główna wpisu,
- \* Zakładka z opiniami o danym wpisie i technologii której wpis dotyczy.
- System musi zapewnić każdemu użytkownikowi przeglądanie wpisów, przy czym wyszukiwanie wpisów będzie możliwe poprzez następujących kryteriach:
  - \* Tytuł wpisu,
  - \* Popularność wpisu (ilość reakcji przy wpisach),
  - \* Autor wpisu,
  - \* Technologia,
  - \* Kategoria technologii,
  - \* Zastosowanie technologii,
  - \* Data powstania wpisu,
  - \* Data modyfikacji wpisu,
  - \* Ocena wpisu.
- System musi zapewnić zwykłemu użytkownikowi dodawanie wpisów,
- System musi zapewnić zwykłemu użytkownikowi oraz administratorowi edytowanie wpisów,
- System musi zapewnić zwykłemu użytkownikowi oraz administratorowi usuwanie wpisów,
- System musi zapewnić recenzentowi weryfikację i akceptację utworzonych i edytowanych wpisów,
- System musi zapewnić recenzentowi oraz administratorowi wycofywanie wpisów,
- System musi zapewnić zwykłemu użytkownikowi wysyłanie odwołania w sprawie nie zaakceptowanego utworzonego lub edytowanego przez siebie wpisu,
- System musi zapewnić recenzentowi wysyłanie odpowiedzi na odwołanie użytkownika w sprawie nie zaakceptowanego wpisu (utworzonego lub edytowanego).
- Opinie
  - Opinie odnośnie wpisów oraz technologii będą się składać z oceny w postaci wartości od 1 do 5 oraz tekstowego komentarza,
  - System musi zapewnić każdemu użytkownikowi przeglądanie opinii o wpisach i technologiach,
  - System musi zapewnić zwykłemu użytkownikowi dodawanie i edytowanie opinii o wpisach i technologiach,
  - System musi zapewnić zwykłemu użytkownikowi oraz administratorowi usuwanie opinii o wpisach i technologiach.
- System musi zapewnić administratorowi przypisywanie i usuwanie użytkownikom ról,
- System musi zapewnić każdemu użytkownikowi kontakt tekstowy z administracją,
- System musi zapewnić administratorowi kontakt tekstowy z każdym użytkownikiem.

# Rozdział 5

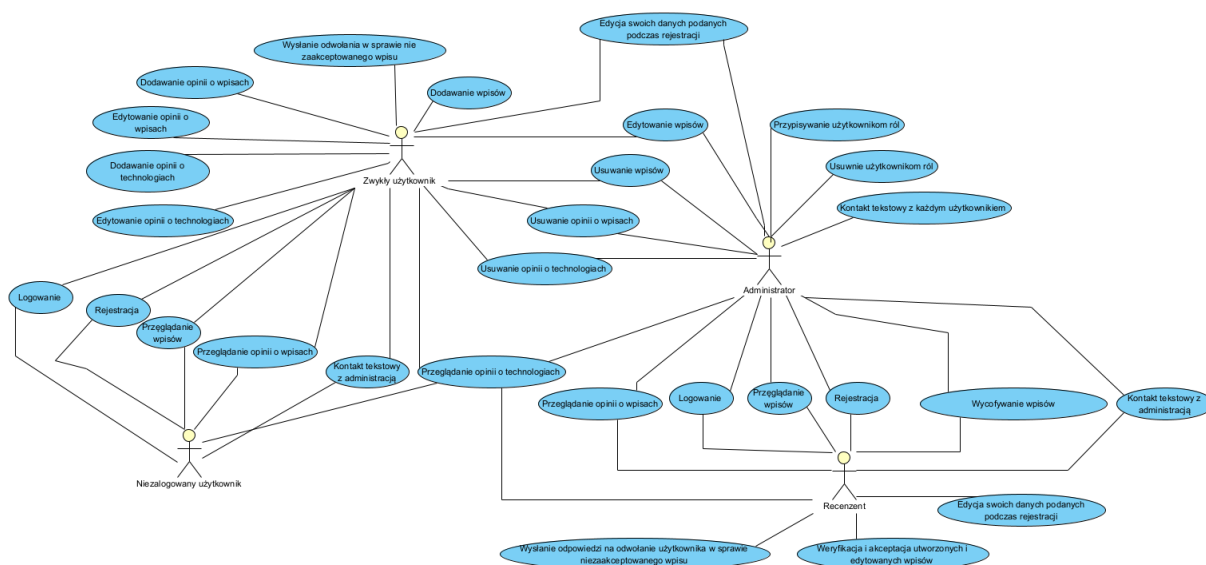
## Wymagania niefunkcjonalne

- Wymagania dotyczące bezpieczeństwa systemu
  - Konta recenzenta oraz administratora mogą być utworzone jedynie przez administratora,
  - Użytkownik o wybranej i posiadanej przez siebie roli będzie miał dostęp do systemu jedynie w zakresie uprawnień przypisanych do tej roli,
  - Dostęp do systemu w ramach kont zwykłego użytkownika, recenzenta oraz administratora jest możliwy jedynie po zalogowaniu się na konto,
  - Zwykły użytkownik może edytować i usuwać dane jedynie te, które zostały przez niego utworzone.
- Wymagania dotyczące liczby zasobów i obciążenia systemu
  - Całkowita liczba użytkowników - 200000,
  - Maksymalna liczba zalogowanych użytkowników w jednym momencie - 50000,
  - Liczba wpisów - 600000,
  - Średni rozmiar wpisu - 20 KB,
  - Średnia liczba zapytań na sekundę - 1000,
  -
- Wykorzystywane technologie i narzędzia
  - Backend - Spring Boot,
  - Frontend - React (główny framework), Typescript (statyczne typowanie), MUI (biblioteka komponentów),
  - Baza danych 1 (główna baza danych) - PostgreSQL,
  - Baza danych 2 (baza danych przechowująca zawartości wpisów) - MongoDB,
  - Warstwa bezpieczeństwa (serwer uwierzytelniania i autoryzacji) - Keycloak,
  - Dokumentacja - LaTeX.

## Rozdział 6

# Diagram przypadków użycia

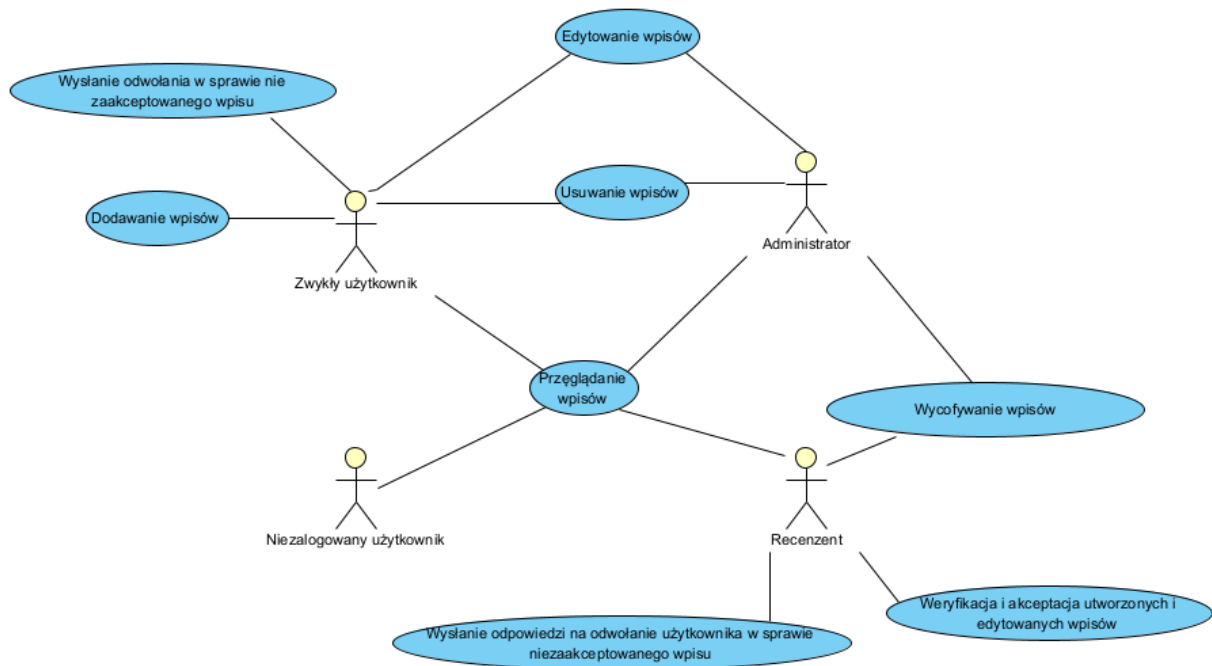
Cały diagram przypadków użycia systemu znajduje się na rysunku 6.1



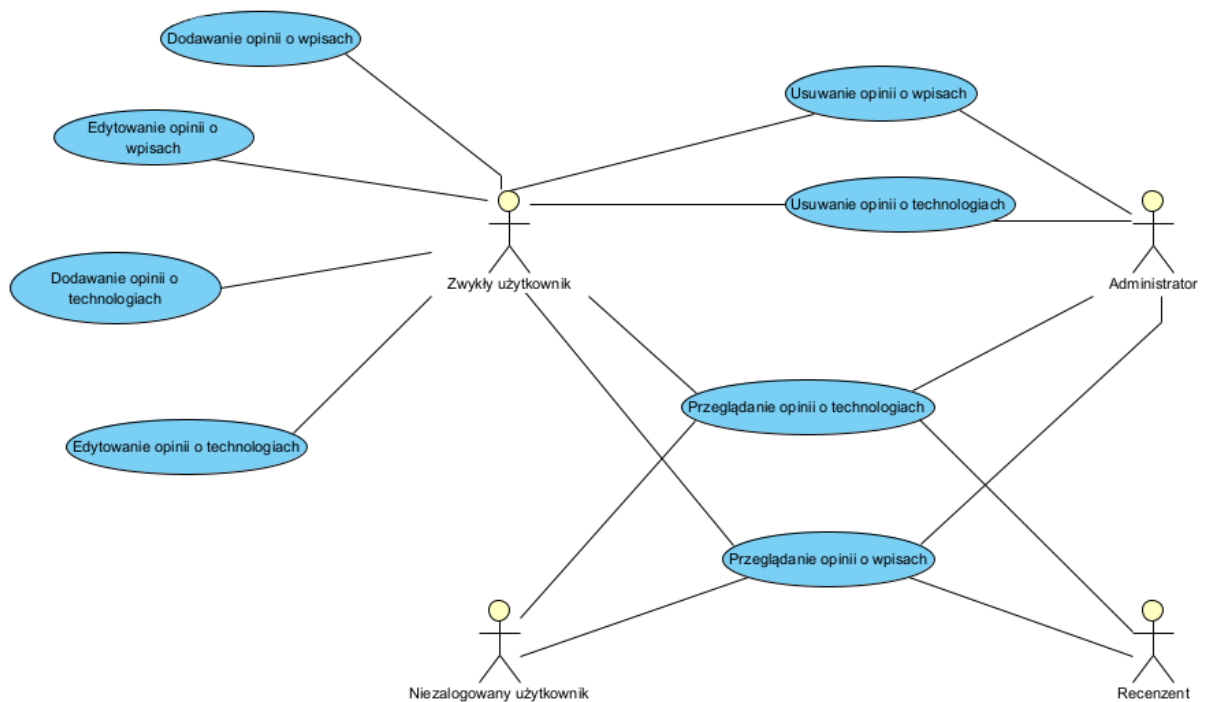
Rys. 6.1: Cały diagram przypadków użycia

Z powodu dość dużej złożoności diagramu na rysunku 6.1, dobrym pomysłem jest podzielenie tego diagramu na kilka mniejszych. Na rysunku 6.2 jest przedstawiony diagram dotyczący wpisów, na rysunku 6.3 widnieje diagram dla opinii o wpisach i technologiach, a na rysunku 6.4 jest diagram zawierający resztę funkcjonalności.

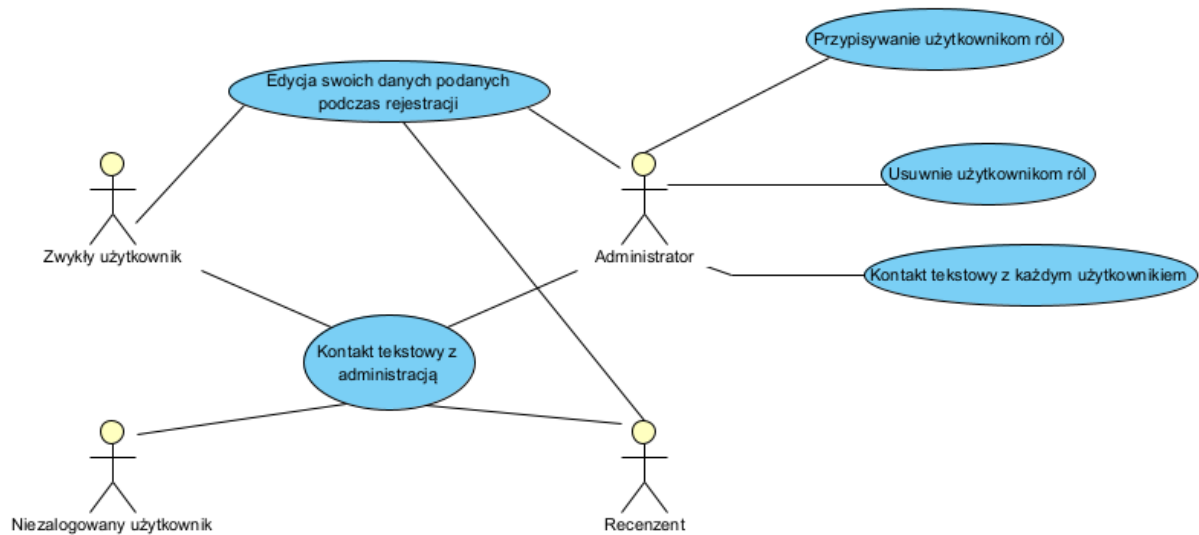




Rys. 6.2: Diagram przypadków użycia o wpisów



Rys. 6.3: Diagram przypadków użycia dotyczący opinii o wpisach i technologiach



Rys. 6.4: Diagram przypadków użycia zawierający resztę funkcjonalności

## **Rozdział 7**

### **Podsumowanie**

# Literatura

**Dodatek A**

**Instrukcja wdrożeniowa**

## **Dodatek B**

### **Opis załączonej płyty CD/DVD**