

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

1. Wstęp	8
1.1. Wprowadzenie	8
1.2. Cel i zakres pracy	8
1.3. Układ pracy	8
2. Architektura systemu	9
2.1. Baza danych	9
2.2. Aplikacja serwerowa - Backend	9
2.3. Klient - Frontend	10
2.4. Warstwa bezpieczeństwa	10
2.5. REST	10
3. Wymagania funkcjonalne	11
3.1. Definicje	11
3.2. Wymagania funkcjonalne poszczególnych ról użytkowników	14
4. Wymagania нефункционалне	18
5. Diagramy przypadków użycia	20
5.1. Identyfikacja aktorów	20
5.2. Diagramy przypadków użycia	20
5.3. Opisy przypadków użycia	23
6. Podsumowanie	28
Literatura	29
A. Instrukcja wdrożeniowa	30
B. Opis załączonej płyty CD/DVD	31

Spis rysunków

2.1. Architektura systemu	9
3.1. Cykl życia artykułu	13
5.1. Dziedziczenie między aktorami	21
5.2. Diagram przypadków użycia dla każdego użytkownika	21
5.3. Diagram przypadków użycia dla niezalogowanego użytkownika	22
5.4. Diagram przypadków użycia dla zalogowanego użytkownika	22
5.5. Diagram przypadków użycia dla recenzenta	22
5.6. Diagram przypadków użycia dla administratora	23
5.7. Prototyp formularza rejestracji	25
5.8. Prototyp formularza z akceptacją regulaminu	26

Spis tabel

5.1. Opis przypadku użycia - rejestracja	24
5.2. Opis przypadku użycia - logowanie	27

Spis listingów

Skróty

GUI (ang. *graphical user interface*)

Rozdział 1

Wstęp

1.1. Wprowadzenie

1.2. Cel i zakres pracy

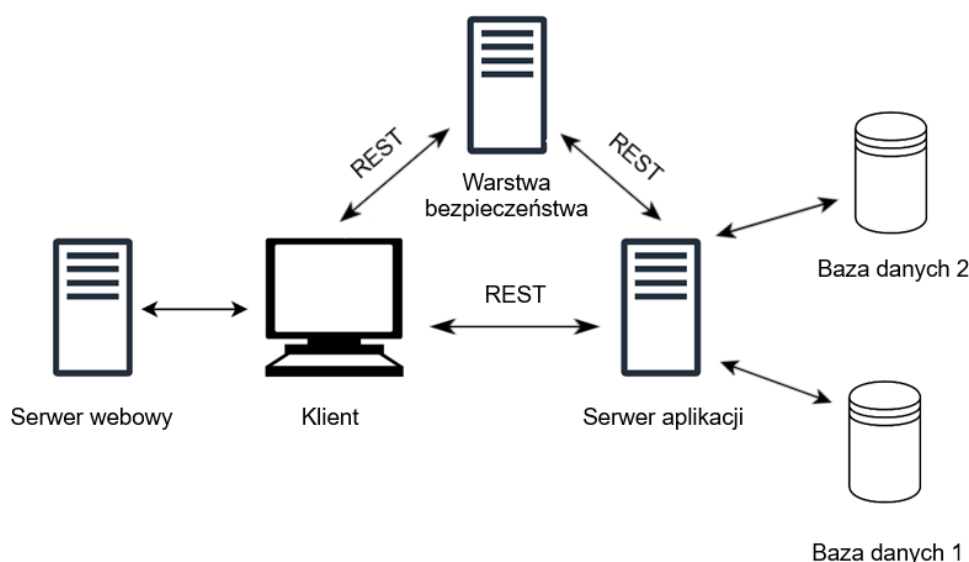
1.3. Układ pracy

Rozdział 2

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 2.1



Rys. 2.1: Architektura systemu

2.1. 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 artykułów.

2.2. 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.

2.3. 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.

2.4. 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 umożliwia również autoryzację użytkowników w oparciu o ich role.

2.5. 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 artykułu. 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 3

Wymagania funkcjonalne

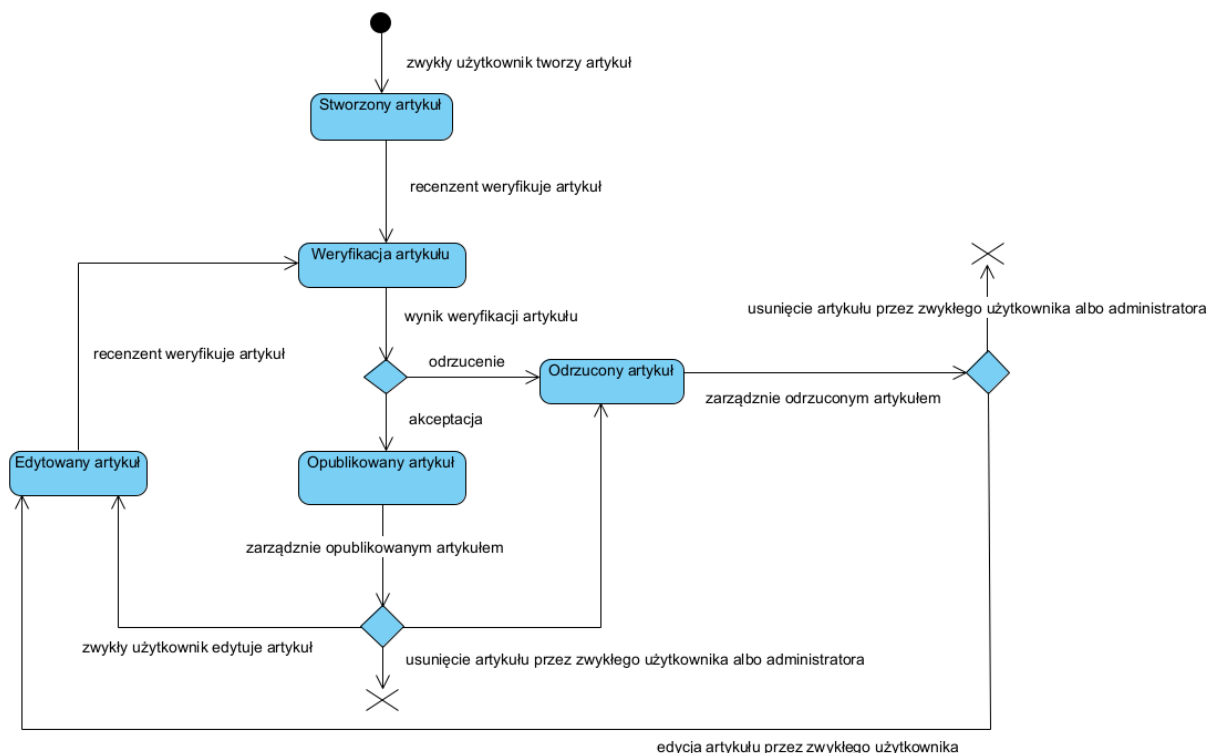
3.1. Definicje

- Technologie - technologie i narzędzia wykorzystywane przez informatyków, a przede wszystkim przez programistów. Technologie będą opisywane następującymi atrybutami:
 - Nazwa,
 - Dostawca (opcjonalne),
 - Ikona (opcjonalne),
 - Kategoria,
 - Zastosowania,
 - Lista opinii,
 - Data pierwszego wydania (opcjonalne),
 - Data ostatniego wydania (opcjonalne).
- Klasyfikacja technologii - podział technologii na kategorie w postaci hierarchicznego drzewa. Można wyróżnić dwa kryteria podziału tego drzewa:
 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)
 - * Język znaczników (np. TeX, HTML, XML)
 - * Język bazy danych (np. SQL, GraphQL)
 - Relacyjna (np. MySQL, Oracle Database, SQLite)
 - Obiektowo-relacyjna (np. PostgreSQL)
 - NoSQL (np. MongoDB, Cassandra)
 - Środowiska uruchomieniowe
 - * System operacyjny (np. Windows, Linux)
 - * Wysokopoziomowe (JVM, .NET, node.js)
 - * Niskopoziomowe (np. Arduino, Raspberry Pi)
 - Narzędzia (oprogramowanie)
 - * Środowisko programistyczne (np. Netbeans, Eclipse, Visual Studio Code, Visual Studio, IntelliJ)
 - * System kontroli wersji (np. git)
 - Serwisy hostujące gita (np. GitLab, GitHub)

- * CI/CD (np. Jenkins, GitLab CI)
 - * Konteneryzacja (np. docker)
 - * Orkiestracja (system do zarządzania, organizacji i planowania zasobów systemu – np. Docker Compose, Kubernetes)
 - Inne (w przypadku, gdy w bazie danych nie znaleziono kategorii technologii ze względu na jej architekturę)
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
- Artykuł - informacja o danej technologii stworzona przez użytkownika. Artykuł będzie opisywany następującymi atrybutami:
 - Status (możliwe wartości: stworzony, weryfikowany, edytowany, odrzucony, opublikowany),
 - Tytuł,
 - Autor,
 - Data utworzenia,

- Data modyfikacji,
- Kategoria technologii wraz z całą hierarchiczną ścieżką prowadzącą do tej kategorii (np. dla Javy będzie to: Technologie Informatyczne / Język Programowania / Java),
- Zastosowania technologii,
- Dostawca technologii (opcjonalne),
- Ikona technologii (opcjonalne),
- Tekst artykułu,
- Lista opinii o artykule,
- Lista opinii o technologii,
- Średnia ocena z co najmniej 5-ciu opinii o artykule.

Artykuły podlegają cyklowi życia, który jest przedstawiony na diagramie maszyny stanów 3.1:



Rys. 3.1: Cykl życia artykułu

Diagram 3.1 przedstawia przejścia do odpowiednich statusów artykułu (zaokrąglone prostokąty) w zależności od podjętych przez użytkowników działań.

- Rola - zestaw uprawnień użytkownika, które określają jego zakres dostępu do systemu oraz określonych funkcji tego systemu. Możliwe jest przypisanie danemu użytkownikowi ról, co jest równoznaczne z uzyskaniem przez tego użytkownika uprawnień zdefiniowanych w tych rolach. Będzie można wyróżnić w systemie następujące role:
 - Administrator (utrzymuje porządek w serwisie) - zarejestrowany i zalogowany użytkownik, który otrzymał od administratora rolę administratora. Odpowiada on przede wszystkim za przydzielanie ról użytkownikom, ustalanie listy technologii, w których dany recenzent jest ekspertem oraz utrzymywanie porządku w serwisie poprzez edytowanie i usuwanie niewłaściwych treści (np. wirusy lub spam). Przy instalacji systemu powinno być tworzone jedno konto użytkownika z rolą administratora, gdyż to właśnie administrator odpowiada za przydzielanie ról innym użytkownikom.
 - Użytkownik - każdy użytkownik korzystający z serwisu. Może on jedynie przeglądać zasoby serwisu,

- Niezalogowany użytkownik - niezalogowany użytkownik mający możliwość przeglądania zasobów serwisu oraz może się zarejestrować i zalogować,
- Zalogowany użytkownik - każdy zarejestrowany i zalogowany użytkownik. Może on przede wszystkim przeglądać i tworzyć artykuły oraz wystawiać opinie o artykułach i opinie o technologiach.
- Recenzent (ekspert od technologii) - zarejestrowany i zalogowany użytkownik, który otrzymał od administratora rolę recenzenta. Jego głównym zadaniem jest weryfikacja artykułów przed ich opublikowaniem w serwisie. Na podstawie utworzonej przez administratora listy technologii, w których dany recenzent jest ekspertem, przydzielane są temu recenzentowi odpowiednie artykuły do zrecenzowania (zweryfikowania). Innym uprawnieniem tego użytkownika jest m.in. zarządzanie klasyfikacją technologii.
- Komentarz do artykułu - tekstowa informacja zwrotna zalogowanego użytkownika o danym artykule,
- Komentarz do technologii - tekstowa informacja zwrotna zalogowanego użytkownika o danej technologii,
- Ocena artykułu - ocena wystawiona przez zalogowanego użytkownika na temat danego artykułu. Ocena ta jest w postaci gwiazdek będąca wartością od 1 do 5, przy czym wartość 1 oznacza ocenę negatywną i wskazuje, że artykuł posiada rażące błędy i powinien zostać edytowany albo usunięty, a wartość 5 oznacza ocenę pozytywną i wskazuje, że artykuł nie posiada rażących błędów oraz informacje w nim zawarte są niezwykle przydatne dla użytkowników,
- Ocena technologii - ocena wystawiona przez zalogowanego użytkownika na temat danej technologii. Ocena ta jest w postaci gwiazdek będąca wartością od 1 do 5, przy czym wartość 1 oznacza ocenę negatywną i wskazuje, że technologia zawiera duże błędy, jest niewydajna, jest trudna w użyciu lub nie ma zastosowań, a 5 oznacza ocenę pozytywną i wskazuje, że technologia nie zawiera rażących błędów, jest wydajna, jest niezbyt trudna w użyciu oraz ma wiele zastosowań lub w przypadku wąskiego zakresu zastosowań, to technologia ta jest mocno konkurencyjna,
- Opinia o artykule - oceniająca informacja zwrotna zalogowanego użytkownika na temat danego artykułu, która zawiera tekstowy komentarz oraz ocenę tego artykułu,
- Opinia o technologii - oceniająca informacja zwrotna zalogowanego użytkownika na temat danej technologii, która zawiera tekstowy komentarz oraz ocenę tej technologii,
- Akceptacja opinii o artykule - ocena wystawiona przez zalogowanego użytkownika dotycząca opinii o artykule. Przyjmuje ona dwie wartości: ocena pozytywna (łapka w górę) oznaczająca, że użytkownik zgadza się z opinią i ocena negatywna (łapka w dół) oznaczająca, iż użytkownik nie zgadza się z opinią,
- Akceptacja opinii o technologii - ocena wystawiana przez zalogowanego użytkownika dotycząca opinii o technologii. Przyjmuje ona dwie wartości: ocena pozytywna (łapka w górę) oznaczająca, że użytkownik zgadza się z opinią i ocena negatywna (łapka w dół) oznaczająca, iż użytkownik nie zgadza się z opinią,

3.2. Wymagania funkcjonalne poszczególnych ról użytkowników

- Użytkownik
 - System powinien zapewnić użytkownikowi możliwość wyszukiwania artykułów po następujących kryteriach:
 - * Tytuł,
 - * Autor,

- * Kategoria technologii,
- * Zastosowania technologii,
- * Dostawca technologii,
- * Zakres dat, w których mieści się data powstania,
- * Zakres dat, w których mieści się data modyfikacji,
- System powinien zapewnić użytkownikowi możliwość sortowania artykułów po następujących wartościach:
 - * Popularność (liczba opinii),
 - * Data powstania,
 - * Data modyfikacji,
 - * Średnia ocena.
- System powinien zapewnić użytkownikowi możliwość wyświetlenia zawartości artykułu,
- System powinien zapewnić użytkownikowi możliwość przeglądania komentarzy o artykule,
- System powinien zapewnić użytkownikowi możliwość przeglądania opinii o artykule,
- System powinien zapewnić użytkownikowi możliwość wyszukiwania technologii po następujących kryteriach:
 - * Nazwa
 - * Kategoria,
 - * Zastosowania,
 - * Dostawca,
 - * Zakres dat, w których mieści się data pierwszego wydania,
 - * Zakres dat, w których mieści się data ostatniego wydania.
- System powinien zapewnić użytkownikowi możliwość sortowania technologii po następujących wartościach:
 - * Popularność (liczba opinii),
 - * Data pierwszego wydania,
 - * Data ostatniego wydania,
 - * Średnia ocena.
- System powinien zapewnić użytkownikowi możliwość wyświetlenia opisu technologii,
- System powinien zapewnić użytkownikowi możliwość przeglądania komentarzy o technologii,
- System powinien zapewnić użytkownikowi możliwość przeglądania opinii o technologii,
- Niezalogowany użytkownik
 - System powinien zapewnić użytkownikowi możliwość zarejestrowania się
 - * Podczas rejestracji użytkownik będzie podawał następujące informacje:
 - Nazwa użytkownika,
 - Imię,
 - Nazwisko,
 - Adres e-mail,
 - (Opcjonalnie) Awatar,
 - Hasło.
 - * Po rejestracji i zalogowaniu, użytkownik uzyskuje uprawnienia zalogowanego użytkownika, a aby uzyskać uprawnienia recenzenta lub administratora, należy w tym celu skontaktować się z administratorem, który będzie w stanie nadać taką rolę,
 - System powinien zapewnić użytkownikowi możliwość zalogowania się
 - * Podczas logowania użytkownik będzie wpisywał login lub e-mail oraz hasło,

- * Użytkownik będzie mógł w tym samym momencie korzystać jedynie z uprawnień w ramach jednej posiadanej i wybranej przez niego roli,
- * Podczas logowania i po zalogowaniu powinna być dostępna dla użytkownika możliwość wybrania jednej z posiadanych przez niego ról.
- Zalogowany użytkownik
 - System powinien zapewnić użytkownikowi możliwość edycji swoich danych,
 - System powinien zapewnić użytkownikowi możliwość wylogowania się,
 - System powinien zapewnić użytkownikowi możliwość zarządzania własnymi artykułami w następującym zakresie:
 - * Dodawanie artykułów, przy czym aby artykuł został opublikowany w serwisie, to artykuł ten musi przejść pozytywnie weryfikację przeprowadzaną przez recenzenta,
 - * Edytowanie artykułów których użytkownik jest autorem, przy czym aby zmieniony artykuł został opublikowany w serwisie, to artykuł ten musi przejść pozytywnie weryfikację przeprowadzaną przez recenzenta,
 - * Usuwanie artykułów których użytkownik jest autorem.
 - System powinien zapewnić użytkownikowi możliwość zarządzania komentarzami o artykule w następującym zakresie:
 - * Dodawanie komentarzy o artykule,
 - * Edytowanie własnych komentarzy o artykule,
 - * Usuwanie własnych komentarzy o artykule,
 - * Usuwanie komentarzy o własnym artykule.
 - System powinien zapewnić użytkownikowi możliwość zarządzania opiniami o artykułach w następującym zakresie:
 - * Dodanie opinii o artykule którego użytkownik nie jest autorem. Użytkownik będzie mógł zostawić tylko jedną opinię dla danego artykułu,
 - * Edytowanie własnych opinii o artykułach,
 - * Usuwanie własnych opinii o artykułach,
 - * Ukrywanie tekstu opinii o własnym artykule,
 - * Dodanie akceptacji opinii o artykule. Użytkownik będzie mógł zostawić akceptację opinii, której nie jest autorem oraz będzie mógł zostawić tylko jedną akceptację dla danej opinii o artykule,
 - * Usunięcie własnej akceptacji opinii o artykule.
 - System powinien zapewnić użytkownikowi możliwość zarządzania opiniami o technologiach w następującym zakresie:
 - * Dodawanie opinii o technologiach. Użytkownik będzie mógł zostawić tylko jedną opinię dotyczącą danej technologii
 - * Edytowanie własnych opinii o technologiach,
 - * Usuwanie własnych opinii o technologiach,
 - * Dodanie akceptacji opinii o technologii. Użytkownik będzie mógł zostawić akceptację opinii, której nie jest autorem oraz będzie mógł zostawić tylko jedną akceptację dla danej opinii o technologii,
 - * Usunięcie własnej akceptacji opinii o technologii.
- Recenzent
 - System powinien zapewnić użytkownikowi możliwość edycji rodziny technologii,
 - System powinien zapewnić użytkownikowi możliwość ustawienia swojej dostępności na „dostępny” albo „nieдоступny”. W przypadku ustawionej niedostępności nie będą takiemu recenzentowi przypisywane artykuły do zweryfikowania.

- System powinien zapewnić użytkownikowi możliwość weryfikacji artykułów
 - * Powinna być możliwość automatycznego przypisywania artykułów do recenzowania recenzentom na podstawie ich dostępności, posiadanych przez nich list technologii, w których są ekspertami oraz liczby recenzowanych przez nich artykułów. W pierwszej kolejności nie będą brani pod uwagę niedostępni recenzenci, następnie będą wybierani recenzenci, którzy są ekspertami w jak najbardziej szczegółowej technologii (np. Java jest bardziej szczegółowa niż języki programowania), której dotyczy artykuł oraz w trzeciej kolejności będą wybierani recenzenci, którzy recenzowali najmniejszą liczbę artykułów.
 - * Weryfikację danego artykułu będzie mógł przeprowadzić jedynie jeden recenzent,
 - * Następnie po przypisaniu artykułu do recenzowania danemu recenzentowi, recenzent ten będzie miał tydzień na akceptację lub odrzucenie artykułu. W przypadku przekroczenia terminu zostanie usunięte przypisanie artykułu do weryfikacji i będzie to skutkowało ponownym przypisaniem artykułu do weryfikacji innemu recenzentowi. Przy akceptacji artykułu opcjonalne jest załączenie wiadomości (np. drobne uwagi), ale przy odrzuceniu artykułu załączenie wiadomości jest obowiązkowe i powinna ona zawierać informacje o tym, dlaczego artykuł został odrzucony.
- Administrator
 - System powinien zapewnić użytkownikowi możliwość edycji danych innych użytkowników,
 - System powinien zapewnić użytkownikowi możliwość zarządzania artykułami w następującym zakresie:
 - * Możliwość edytowania artykułów,
 - * Możliwość usuwania artykułów,
 - * Możliwość wycofywania artykułów.
 - System powinien zapewnić użytkownikowi możliwość usuwania komentarzy o artykule,
 - System powinien zapewnić użytkownikowi możliwość usuwania opinii o artykule,
 - System powinien zapewnić użytkownikowi możliwość usuwania opinii o technologii,
 - System powinien zapewnić użytkownikowi możliwość przypisywania innym użytkownikom ról,
 - System powinien zapewnić użytkownikowi możliwość usuwania przypisanych ról innych użytkowników,

Rozdział 4

Wymagania нефunkcjonalne

- Wymagania dotyczące bezpieczeństwa systemu
 - Użytkownik o posiadanej przez siebie roli będzie miał dostęp do systemu jedynie w zakresie uprawnień przypisanych do tej roli,
 - Używanie zewnętrznego serwera uwierzytelniającego i autoryzacyjnego,
 - Używanie tokenu na okaziciela (ang. *Bearer token*) przy wykonywaniu zapytań do Backendu,
 - Nieprzesyłanie hasła, jeśli nie jest to konieczne,
 - Tworzone hasła powinny spełniać następujące kryteria:
 - * Minimalna długość - 8,
 - * Co najmniej jedna mała litera,
 - * Co najmniej jedna duża litera,
 - * Co najmniej jedna cyfra.
- Wymagania dotyczące obciążenia systemu
 1. Całkowita liczba użytkowników - 200000 (liczba przyjęta z góry),
 2. Średni dzienny czas spędzania użytkowników w serwisie - 1 godzina (liczba przyjęta z góry),
 3. Średnia liczba zalogowanych użytkowników w jednym momencie - 8333 ($1 / (24 / 2.)$),
 4. Liczba artykułów - 600000 (liczba przyjęta z góry),
 5. Średni rozmiar artykułu - 8.2 KB (średnia waga ikony technologii - 4.2 KB, średnia waga zawartości artykułu - 4 KB (sam tekst) - (przyjęto, że zostaną wzięte pod uwagę najbardziej znaczące dane)),
 6. Średni czas spędzania użytkownika na jednym artykule - 2 minuty
 7. Średnia liczba zapytań do Bazy danych 1 na minutę dla jednego użytkownika - 2.5 (Założono, że użytkownik średnio spędza 2 minuty na jednym artykule, a aby wyświetlić kolejny artykuł, to użytkownik najczęściej wpisuje tytuł tego artykułu, artykuł jest następnie wyszukiwany (1 zapytanie), a następnie do wyświetlenia otrzymanego artykułu wywoływane są 4 zapytania (zapytanie o podstawowe dane o artykule (tytuł, autor, itp.), zapytanie o komentarze oraz zapytanie o opinie o artykule, zapytanie o opinie o technologii), czyli łącznie 5 zapytań co 2 minuty)
 8. Średnia liczba zapytań do Bazy danych 1 na sekundę - 347 ($7 / 60 * 3.$),
 9. Średnia liczba zapytań do Bazy danych 2 na minutę dla jednego użytkownika - 0.5 (Założono, że użytkownik średnio spędza 2 minuty na jednym artykule, a z powodu tego, że w tej bazie danych przechowywane są jedynie zawartości artykułów, zatem można uznać, że co 2 minuty wywoływane jest 1 zapytanie)
 10. Średnia liczba zapytań do Bazy danych 2 na sekundę - 69 ($9 / 60 * 3.$),

- 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 artykułów) - MongoDB,
 - Warstwa bezpieczeństwa (serwer uwierzytelniania i autoryzacji) - Keycloak,
 - Dokumentacja - LaTeX.

Rozdział 5

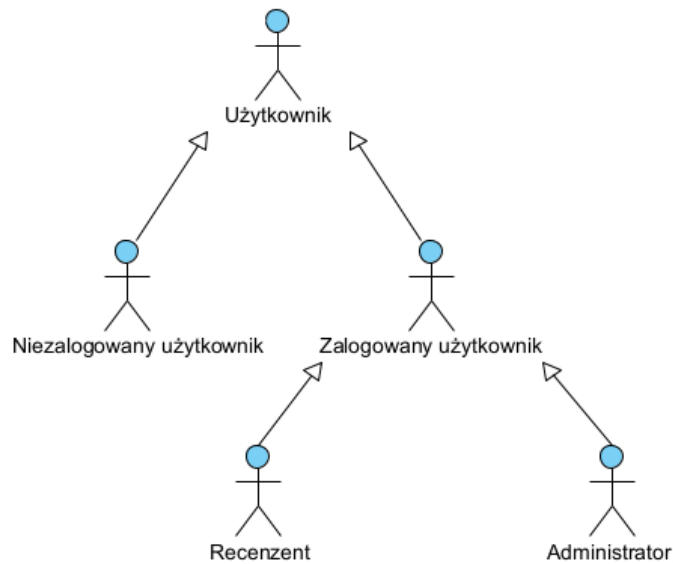
Diagramy przypadków użycia

5.1. Identyfikacja aktorów

- Użytkownik - rola przypisywana każdemu użytkownikowi, która uprawnia go do przeglądania zasobów serwisu,
- Niezalogowany użytkownik - rola przypisywana każdemu niezalogowanemu użytkownikowi, która go użytkownika do rejestracji i logowania,
- Zalogowany użytkownik - rola przypisywana każdemu zalogowanemu użytkownikowi, która uprawnia go do m.in. tworzenia artykułów, czy zostawiania komentarzy,
- Recenzent - rola przypisywana użytkownikom przez administratora, którzy będą się zajmować przede wszystkim weryfikacją artykułów.
- Administrator - rola przypisywana użytkownikom przez administratora, którzy będą się zajmować przede wszystkim utrzymywaniem porządku w serwisie. Rola ta charakteryzuje się największymi uprawnieniami.

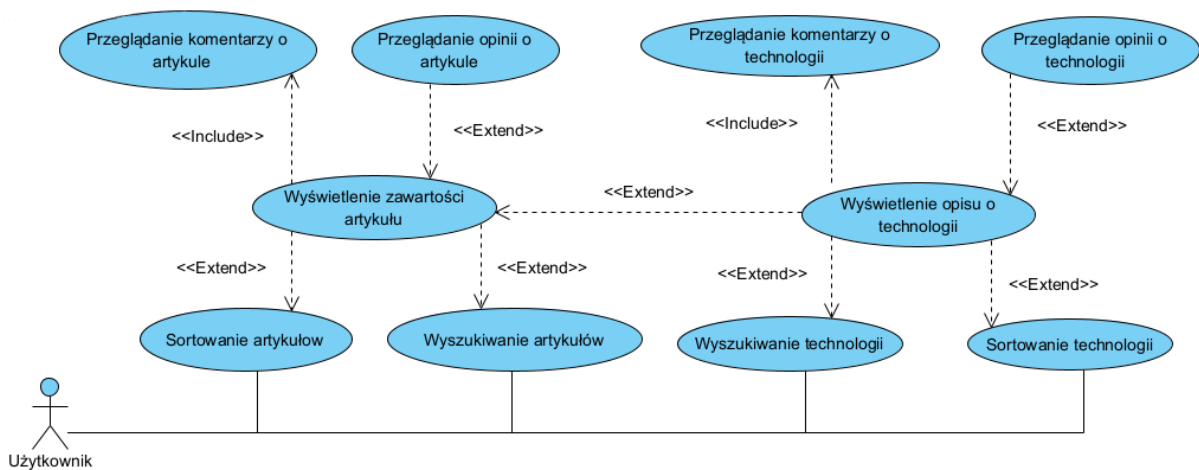
5.2. Diagramy przypadków użycia

Zdecydowano się na zastosowanie dziedziczenia między aktorami, gdyż zdefiniowane role charakteryzują się tym, że posiadają wiele wspólnych uprawnień (przypadków użycia) i działanie to pozwala na zmniejszenie złożoności systemu. Dziedziczenie zostało przedstawione na rysunku 5.1.

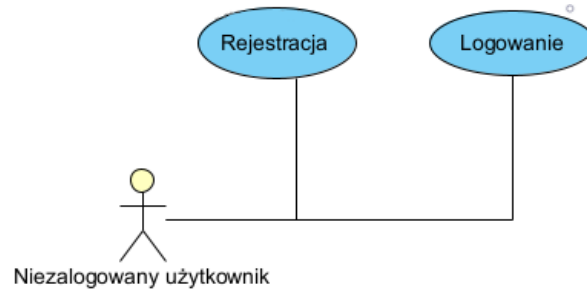


Rys. 5.1: Dziedziczenie między aktorami

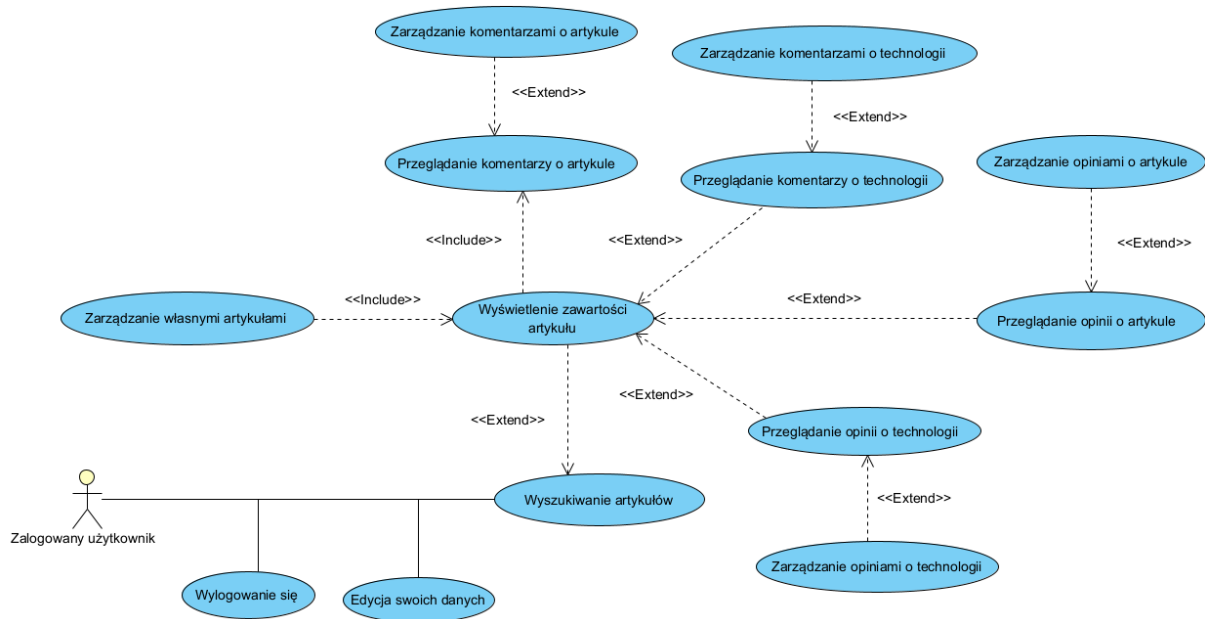
Z powodu dosyć dużej złożoności systemu pod względem liczby wymagań funkcjonalnych i liczby ról, to dla zachowania czytelności diagramu, stworzono osobne diagramy dla każdej roli. Na rysunku 5.2 jest pokazany diagram dotyczący użytkownika (każdego użytkownika), na rysunku 5.3 jest przedstawiony diagram dla niezalogowanego użytkownika, na rysunku 5.4 widnieje diagram dotyczący zalogowanego użytkownika, na rysunku 5.5 jest diagram zawierający funkcjonalności recenzenta, a na rysunku 5.6 jest przedstawiony diagram dla administratora.



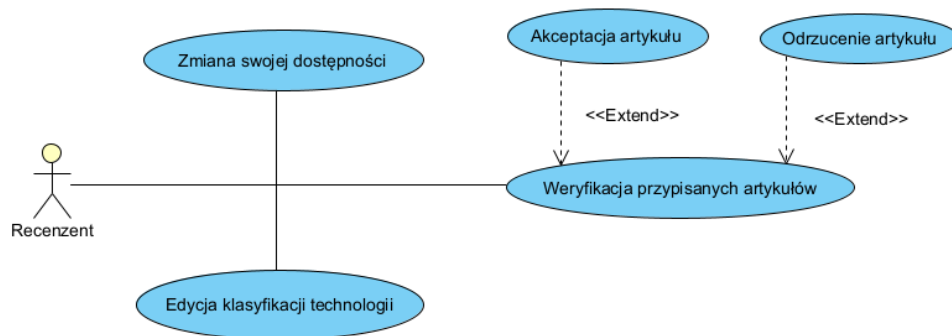
Rys. 5.2: Diagram przypadków użycia dla każdego użytkownika



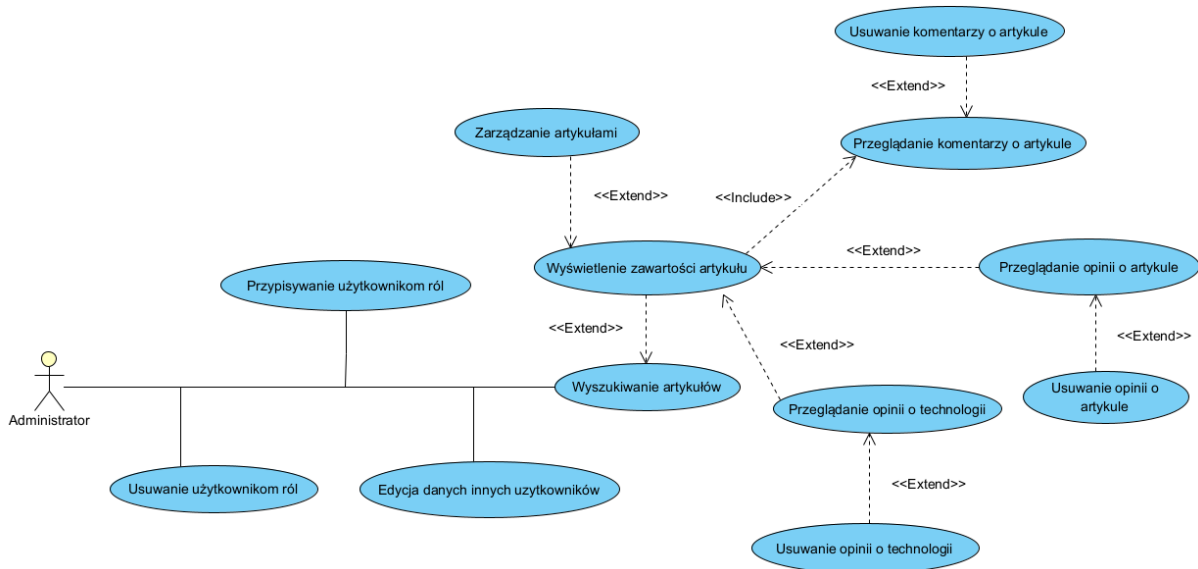
Rys. 5.3: Diagram przypadków użycia dla niezalogowanego użytkownika



Rys. 5.4: Diagram przypadków użycia dla zalogowanego użytkownika



Rys. 5.5: Diagram przypadków użycia dla recenzenta



Rys. 5.6: Diagram przypadków użycia dla administratora

5.3. Opisy przypadków użycia

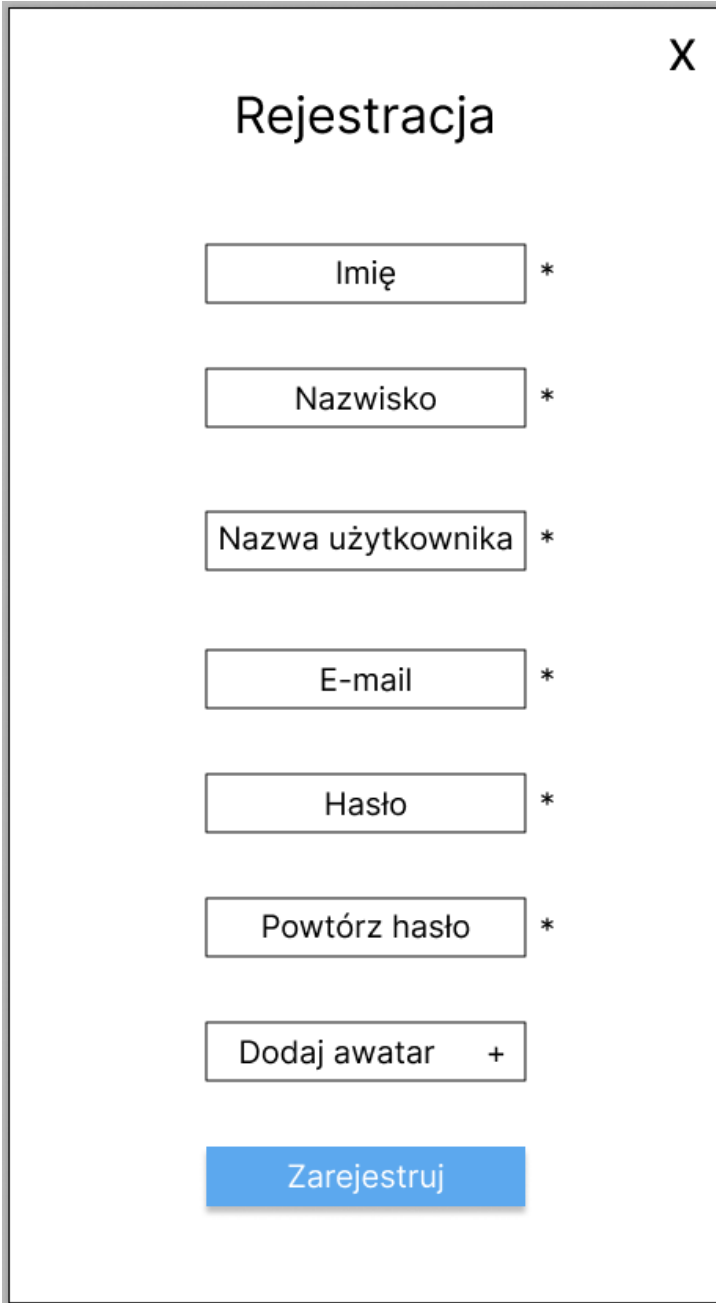
Dla każdego przypadku użycia zdefiniowanego w poprzednim punkcie, zrobiono szczegółowe opisy, które składają się z następujących informacji:

- Nazwa,
- Krótki opis,
- Występujący aktorzy,
- Warunki początkowe,
- Warunki końcowe,
- Przebieg,
- Alternatywne przebiegi.

Dla każdego przypadku użycia zrobiono jeszcze prototypy ekranu użytkownika, aby zaplanować wygląd GUI serwisu oraz aby łatwiej można było pojąć opisy przypadków użycia.

Tab. 5.1: Opis przypadku użycia - rejestracja

Nazwa:	Rejestracja
Opis:	Utworzenie konta w serwisie z uprawnieniami zwykłego użytkownika
Aktorzy:	Niezałogowany użytkownik
Warunki początkowe:	Otwarta strona główna (Rys)
Warunki końcowe:	Założenie konta z uprawnieniami zwykłego użytkownika
Przebieg:	<ol style="list-style-type: none"> 1. Użytkownik klika przycisk „Rejestracja”, 2. Aplikacja wyświetla formularz do rejestracji (Rys. 5.7), 3. Użytkownik wypełnia pola w formularzu, 4. Użytkownik klika przycisk „Zarejestruj”, 5. Serwis przeprowadza walidację danych pod względem wymaganych pól oraz narzuconych rozmiarów i formatów danych, 6. Po pozytywnej walidacji danych, aplikacja wyświetla regulamin serwisu, 7. Użytkownik przegląda cały regulamin i go akceptuje poprzez kliknięcie w pole wyboru po lewej stronie od napisu „Akceptuje regulamin” (Rys. 5.8), 8. Serwis tworzy nowe konto.
Alternatywny przebieg:	<p>Użytkownik anuluje rejestrację</p> <ol style="list-style-type: none"> 2a. Użytkownik klika przycisk „X” lub klika w obszar poza formularzem, 3a. Powrót do strony głównej.
Alternatywny przebieg:	<p>Wprowadzone przez użytkownika dane nie przeszły walidacji</p> <ol style="list-style-type: none"> 2b. Aplikacja zmienia kolor obramowania pól które nie przeszły walidacji na czerwono oraz wyświetla pod tymi polami adekwatny komunikat o błędzie, 3b. Powrót do punktu 2.
Alternatywny przebieg:	<p>Użytkownik nie akceptuje regulaminu</p> <ol style="list-style-type: none"> 6c. Użytkownik klika w pole wyboru po lewej stronie od napisu „Nie akceptuje regulaminu” lub klika przycisk „X” 7c. Powrót do strony głównej.



A registration form prototype titled "Rejestracja" with a close button "X" in the top right corner. The form contains seven input fields, each followed by an asterisk indicating it is required: "Imię", "Nazwisko", "Nazwa użytkownika", "E-mail", "Hasło", and "Powtórz hasło". Below these is a field labeled "Dodaj awatar" with a plus sign icon. At the bottom is a blue button labeled "Zarejestruj".

Rejestracja X

Imię *

Nazwisko *

Nazwa użytkownika *

E-mail *

Hasło *

Powtórz hasło *

Dodaj awatar +

Zarejestruj

Rys. 5.7: Prototyp formularza rejestracji

X

Regulamin

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque convallis, justo ac condimentum auctor, sapien enim pellentesque enim, in maximus enim mi vitae arcu. Phasellus iaculis, orci eu vestibulum sersuscorper magna eget nisi facilisis, id luctus elit facilisis. Pellentesque vel dapibus sapien. Nam in venenatis tortor, rutrum finibus turpis. Pellentesque hendrerit felis quam, eget egestas odio condimentum a. Nulla fringilla bibendum sagittis. Vivamus ut interdum lectus. In vehicula, libero et sollicitudin molestie, erat tortor ullamcorper est, a tincidunt turpis quam sit amet sapien.

Akceptuję regulamin

Nie akceptuję regulaminu

Rys. 5.8: Prototyp formularza z akceptacją regulaminu

Tab. 5.2: Opis przypadku użycia - logowanie

Nazwa:	Logowanie
Opis:	Zalogowanie się użytkownika
Aktorzy:	Niezalogowany użytkownik
Warunki początkowe:	Otwarta strona główna (Rys)
Warunki końcowe:	Zalogowanie się użytkownika i uzyskanie przez niego dostępu do funkcji serwisu adekwatnych do jego uprawnień
Przebieg:	<ol style="list-style-type: none"> 1. Użytkownik klika przycisk „Logowanie”, 2. Aplikacja wyświetla formularz do logowania (Rys. 5.7), 3. Użytkownik wypełnia pola w formularzu, 4. Użytkownik klika przycisk „Zaloguj”, 5. Serwis przeprowadza walidację danych pod względem wymaganych pól oraz narzuconych rozmiarów i formatów danych, 6. Po pozytywnej walidacji danych, użytkownik zostaje zalogowany do systemu i będzie mógł korzystać z funkcji systemu w zakresie adekwatnym do jego uprawnień,
Alternatywny przebieg:	<p>Użytkownik anuluje logowanie</p> <ol style="list-style-type: none"> 2a. Użytkownik klika przycisk „X” lub klika w obszar poza formularzem, 3a. Powrót do strony głównej.
Alternatywny przebieg:	<p>Wprowadzone przez użytkownika dane nie przeszły walidacji</p> <ol style="list-style-type: none"> 2b. Aplikacja zmienia kolor obramowania pól które nie przeszły walidacji na czerwono oraz wyświetla pod tymi polami adekwatny komunikat o błędzie, 3b. Powrót do punktu 2.

Tab. 5.3: Opis przypadku użycia - wylogowanie

Nazwa:	Wylogowanie
Opis:	Wylogowanie się użytkownika
Aktorzy:	Zalogowany użytkownik
Warunki początkowe:	Zalogowanie się użytkownika
Warunki końcowe:	Wylogowanie użytkownika z serwisu i
Przebieg:	<ol style="list-style-type: none"> 1. Użytkownik klika przycisk „Wyloguj”, 2. Aplikacja wyświetla komunikat o wylogowaniu i przekierowuje użytkownika do strony głównej 3. Następuje wylogowanie użytkownika.

Rozdział 6

Podsumowanie

Literatura

Dodatek A

Instrukcja wdrożeniowa

Dodatek B

Opis załączonej płyty CD/DVD