

SYSTEM KOORDYNACJI POMOCY HUMANITARNEJ

Dokumentacja techniczna i użytkowa

Streszczenie

Niniejszy dokument przedstawia dokumentację techniczną oraz użytkową systemu koordynacji pomocy humanitarnej (SKPH). Zawiera on diagramy oraz informacje przedstawiające założenia, działanie oraz sposób użytkowania systemu.

Spis treści

1	Wprowadzenie.....	4
1.1	Krótki opis projektu i jego celu.....	5
1.2	Zakres dokumentacji.....	6
1.3	Lista zespołów odpowiedzialnych za poszczególne moduły i architekturę.....	7
2	Część I: Dokumentacja techniczna.....	8
2.1	Opis wymagań.....	9
2.2	Diagram komponentów	10
2.3	Moduł Auth	11
2.3.1	Cel	11
2.3.2	Założenia.....	11
2.3.3	Wymagania	11
2.3.4	Diagramy.....	12
2.4	Moduł Komunikacji	21
2.4.1	Cel	21
2.4.2	Założenia.....	21
2.4.3	Wymagania	21
2.4.4	Diagramy.....	22
2.5	Moduł Map	34
2.5.1	Cel	34
2.5.2	Założenia.....	34
2.5.3	Wymagania	34
2.5.4	Diagramy.....	34
2.6	Moduł Lang	46
2.6.1	Cel	46
2.6.2	Założenia.....	46
2.6.3	Wymagania	46
2.6.4	Diagramy.....	46
2.7	Moduł Report.....	53
2.7.1	Cel	53
2.7.2	Założenia.....	53
2.7.3	Wymagania	53
2.7.4	Diagramy.....	54
2.8	Moduł Request.....	64
2.8.1	Cel	64

Wprowadzenie

2.8.2	Założenia	64
2.8.3	Wymagania	64
2.8.4	Diagramy.....	65
2.9	Moduł Resource	74
2.9.1	Cel	74
2.9.2	Założenia	74
2.9.3	Wymagania	74
2.9.4	Diagramy.....	75
2.10	Moduł Task	87
2.10.1	Cel.....	87
2.10.2	Założenia.....	87
2.10.3	Wymagania.....	87
2.10.4	Diagramy	88
2.11	Moduł User	102
2.11.1	Cel.....	102
2.11.2	Założenia.....	102
2.11.3	Wymagania.....	102
2.11.4	Diagramy	103
2.12	Diagram wdrożeń systemu.....	120
3	Część II: Dokumentacja użytkownika.....	121
3.1	Budowa strony i język aplikacji.....	122
3.1.1	Panel nawigacyjny.....	122
3.1.2	Zawartość zasobu	122
3.1.3	Zmiana języka aplikacji.....	122
3.2	Rejestracja i logowanie	124
3.2.1	Rejestracja.....	124
3.2.2	Logowanie.....	125
3.3	Komunikator - przesyłanie i odbiór wiadomości tekstowych.....	127
3.3.1	Wybór użytkownika do komunikacji	127
3.3.2	Wysyłanie/odbiór wiadomości	127
3.4	Mapa	129
3.4.1	Role użytkowników i ich uprawnienia.....	129
3.4.2	Ekran główny.....	129
3.4.3	Wyświetlanie lokalizacji zasobu	130
3.4.4	Dodawanie nowych zgłoszeń	130
3.4.5	Nawigacja i interakcja z mapą.....	132

Wprowadzenie

3.5	Raporty.....	135
3.5.1	Uprawnienia do generowania raportów.....	135
3.5.2	Raporty Darczyńcy	135
3.5.3	Raporty Organizacji pomocowej oraz Przedstawicieli władz.	138
3.6	Zgłoszenia.....	140
3.6.1	Funkcje dla poszkodowanego	140
3.6.2	Funkcje dla przedstawiciela władz	142
3.7	Zasoby	144
3.7.1	Organizacja	144
3.7.2	Władze	144
3.7.3	Darczyńca.....	144
3.7.4	Rodzaje zasobów	144
3.7.5	Statusy zasobów	144
3.7.6	Widok darczyńcy.....	145
3.7.7	Widok władz	147
3.7.8	Widok organizacji.....	148
3.8	Zadania.....	150
3.8.1	Funkcjonalności według roli użytkowników.....	150
3.8.2	Opis roli i uprawnienie użytkownika	151
3.8.3	Kompletny opis funkcjonalności i elementów	151
3.9	Użytkownik.....	156
3.9.1	Wypełnienie oraz edycja danych o profilu.....	156
3.9.2	Wypełnienie danych użytkownika	157
3.9.3	Wypełnienie danych organizacji	158
3.9.4	Edycja danych użytkownika	159
3.9.5	Edycja danych organizacji	159
3.9.6	Przeglądanie organizacji	160
3.9.7	Aplikowanie do organizacji	160
3.9.8	Zarządzanie wnioskami.....	162
3.9.9	Dodatkowe informacje o wolontariuszu.....	162
3.9.10	Akceptacja wniosku	163
3.9.11	Odrzucenie wniosku	163
3.9.12	Zarządzanie wolontariuszami	164
3.9.13	Dodatkowe informacje o wolontariuszu	164
3.9.14	Usunięcie wolontariusza.....	165

1 Wprowadzenie

Wprowadzenie

1.1 Krótki opis projektu i jego celu.

Celem projektu było stworzenie systemu informatycznego wspomagającego koordynację działań pomocowych podczas sytuacji kryzysowych, takich jak klęski żywiołowe czy katastrofy ekologiczne. System ma umożliwić sprawne zarządzanie zgłoszeniami poszkodowanych, dystrybucję zasobów, komunikację pomiędzy wszystkimi zaangażowanymi stronami oraz monitorowanie i raportowanie stanu akcji pomocowych. Dążyliśmy do zbudowania platformy, która będzie narzędziem wsparcia dla organizacji pomocowych, wolontariuszy, darczyńców, poszkodowanych oraz przedstawicieli władz.

1.2 Zakres dokumentacji.

Dokumentacja składa się z dwóch kluczowych części, tj. **części technicznej** oraz **części użytkownika**.

- Dzięki **części technicznej** czytelnik zapozna się z opisami wymagań oraz diagramami wykonanymi w notacji UML dla poszczególnych modułów oraz całego systemu. W celu zlokalizowania odpowiedniego diagramu należy w spisie treści odnaleźć odpowiedni moduł oraz przejście do podrozdziału „Diagramy”, który udostępnia wszelkie wymagane informacje oraz wizualizacje diagramów kolejno: przypadków użycia, klas, interakcji, czynności, maszyny stanowej, komponentów, pakietów, przeglądu interakcji, strukturalnych, harmonogramowania. Dodatkowo na końcu dokumentacji technicznej czytelnik może zapoznać się z diagramem wdrożeń całego systemu. Poza diagramami czytelnik zapozna się również z celem, założeniami oraz wymaganiami funkcjonalnymi i niefunkcjonalnymi dla każdego modułu systemu.
- **Część dokumentacji użytkownika** zwana również jako „podręcznik użytkownika” pozwala czytelnikowi zapoznać się z mechanizmami działania interfejsu użytkownika, dzięki któremu dostępne są akcje opisane w przypadkach użycia.

Wprowadzenie

1.3 Lista zespołów odpowiedzialnych za poszczególne moduły i architekturę.

Moduły:

- Auth &Lang
 - Adrian Michałek, Maksymilian Paluśkiewicz
- Chat & Map
 - Natalia Nykiel, Patryk Sałyga
- Report
 - Dawid Frontczak, Mateusz Michalec
- Request
 - Julia Ruszer, Wiktor Żelechowski
- Resource
 - Kuba Cendalski, Kacper Przybylski
- Task
 - Kacper Stasiak, Kacper Witek
- User
 - Jan Ślądowski, Dominik Gałkowski

Architekci:

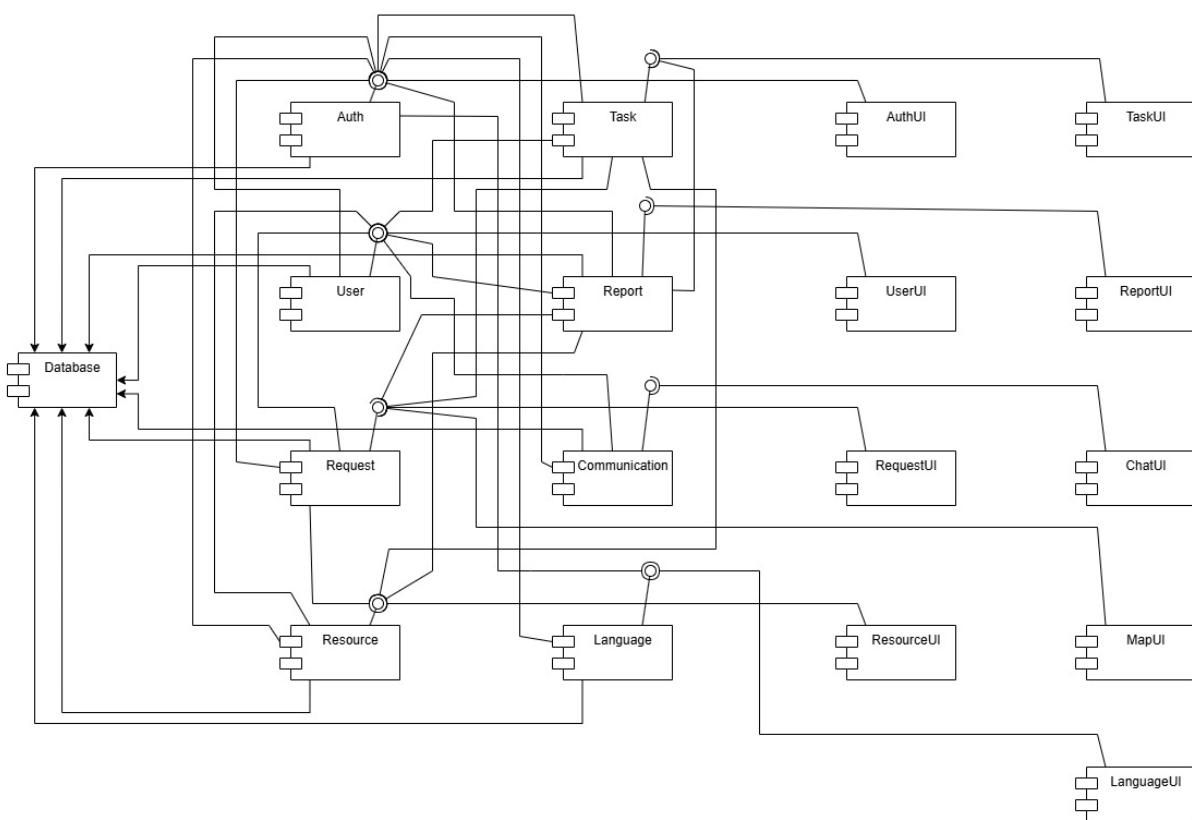
Łukasz Łyczko, Wiktor Glaner

2 Część I: Dokumentacja techniczna

2.1 Opis wymagań

Opis wymagań został podzielony i rozmieszczony zgodnie z zastosowaniem i zrealizowaniem go dla danego modułu. W każdym rozdziale dokumentacji technicznej dotyczącym danego działu czytelnik odnajdzie podrozdział „wymagania”, który jest listą wymagań funkcjonalnych oraz niefunkcjonalnych danego modułu, a co za tym idzie części systemu.

2.2 Diagram komponentów



2.3 Moduł Auth

2.3.1 Cel

Celem modułu jest zapewnienie wysokiego poziomu ochrony danych oraz kontrolowanego dostępu do informacji dla wszystkich uczestników systemu. Moduł ten ma na celu zagwarantowanie, że jedynie autoryzowani użytkownicy mogą uzyskać dostęp do systemu, komunikować się między sobą oraz przetwarzanie wrażliwe dane. Dzięki wdrożonym mechanizmom bezpieczeństwa, moduł ten zwiększa ochronę przed potencjalnymi zagrożeniami oraz buduje zaufanie wśród użytkowników systemu.

2.3.2 Założenia

Bezpieczna autoryzacja i uwierzytelnianie: System powinien umożliwiać bezpieczne logowanie i weryfikację tożsamości użytkowników, przy zastosowaniu zabezpieczenia hasłem.

Kontrola dostępu: Moduł powinien umożliwiać nadawanie uprawnień dostępu w zależności od roli użytkownika, np. wolontariusza, darczyńcy czy przedstawiciela władz, aby zapewnić dostęp tylko do niezbędnych danych i funkcji.

Ochrona prywatności danych: Wszystkie dane użytkowników oraz informacje o działaniach pomocowych powinny być chronione przed nieautoryzowanym dostępem i przetwarzaniem, w zgodzie z obowiązującymi standardami ochrony danych

2.3.3 Wymagania

1. Funkcjonalne

- System musi zapewniać bezpieczne mechanizmy uwierzytelniania w celu ochrony danych wrażliwych i zapewnienia dostępu tylko dla uprawnionych użytkowników
 - Silne mechanizmy uwierzytelniania zapobiegające nieautoryzowanemu dostępowi
- Rejestracja i logowanie użytkowników
 - Unikalne loginy
 - Silne hasła
- Haszowanie haseł
 - Hasła powinny być zaszyfrowane przy użyciu silnych algorytmów hashujących aby nie były widoczne nawet w bazie danych
- Stosowanie tokenów i innych mechanizmów zabezpieczających
 - system powinien stosować tokeny JWT do zarządzania sesją użytkownika, aby zapewnić dostęp do systemu uprawnionym użytkownikom
- Dostęp do systemu tylko po zalogowaniu
 - system powinien implementować zabezpieczenia, które zapewnią dostęp tylko i wyłącznie po zalogowaniu

2. Niefunkcjonalne

- Bezpieczeństwo
 - hasła muszą być przechowywane w bezpieczny sposób przy użyciu algorytmów szyfrujących
 - system musi zapewniać bezpieczne i szyfrowane połączenie z bazą danych

→ Wydajność

- czas odpowiedzi podczas logowania i rejestracji nie powinien przekraczać 2 sekund

→ Dostępność

- moduł powinien być dostępny 99% czasu

→ Użyteczność

- formularze rejestracji i logowania powinny być czytelne i łatwe do wypełnienia
- komunikaty błędów muszą być jasne i zrozumiałe dla każdego potencjalnego użytkownika

→ Skalowalność

- moduł musi obsługiwać jednocześnie rejestracje i logowanie co najmniej 500 użytkowników

2.3.4 Diagramy

2.3.4.1 Diagram przypadków użycia

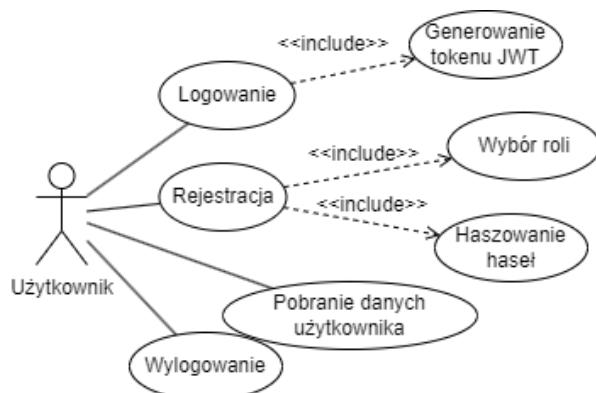
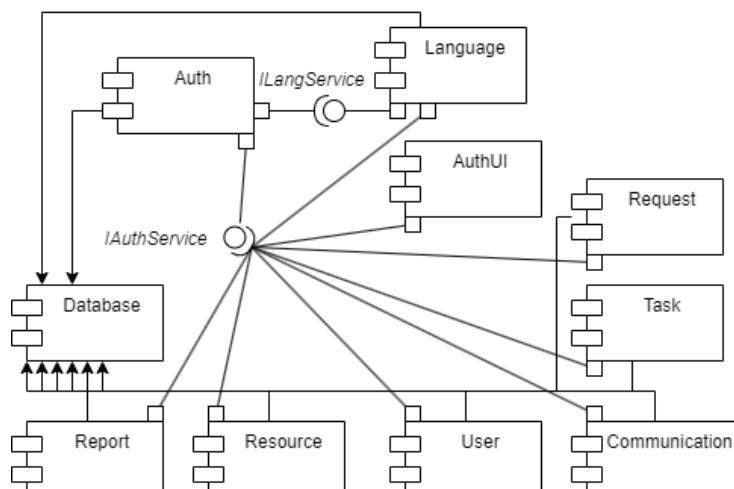


Diagram przypadków użycia przedstawia interakcje między użytkownikiem a modułem autoryzacji (Auth) w kontekście realizowanych funkcji.

Każdy użytkownik systemu dzięki modułowi autoryzacji ma możliwość rejestracji, logowania się, pobierania swoich zasobów oraz wylogowania.

2.3.4.2 Diagram komponentów



Część I: Dokumentacja techniczna

Diagram przedstawia strukturę komponentów w kontekście modułu autoryzacji (*Auth* oraz *AuthUI*). Moduł *Auth* udostępnia interfejs *IAuthService*, który to udostępnia API, dzięki któremu dostęp do innych modułów jest ograniczony tylko do zalogowanych użytkowników oraz tych posiadających odpowiednie uprawnienia. Interfejs *IAuthService* daje również możliwość wykonania zapytań do tabel w bazie danych, którymi zarządza moduł *IAuth* (głównie wykorzystywane przez moduł *User*).

Moduł *Auth* wykorzystuje interfejs *ILangService* udostępniany przez moduł *Language* w celu zapisywania w bazie danych ostatnio wybranego języka przez użytkownika.

2.3.4.3 Diagram pakietów

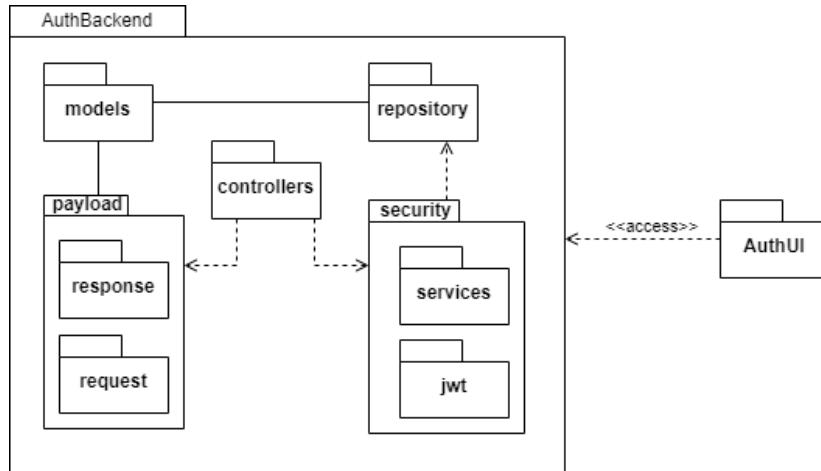
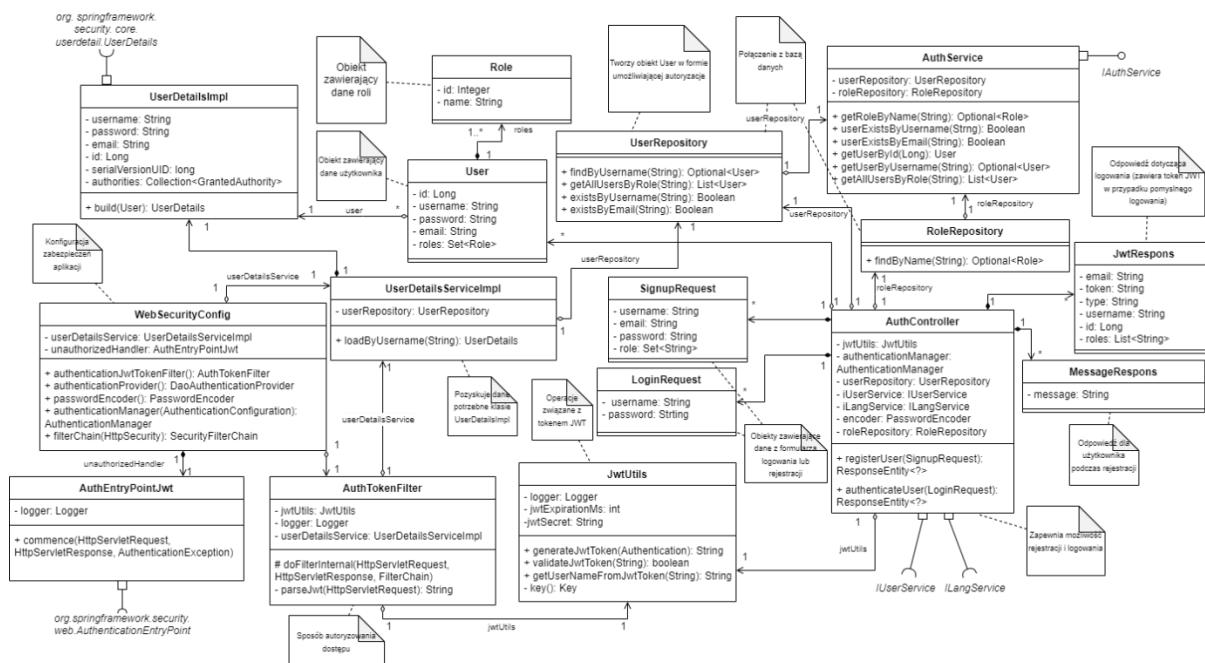


Diagram przedstawia strukturę pakietów wewnętrz modułu autoryzacji *Auth*. Każdy pakiet grupuje klasy odpowiedzialne za określoną funkcjonalność systemu. Pakiet *models* jest odzwierciedleniem tabel w bazie danych, *repository* odpowiada za wysyłanie zapytań do bazy danych, *payload* zawiera modele żądań i odpowiedzi, *security* odpowiada za autoryzację i uwierzytelnianie, *controllers* zawiera endpointy a *AuthUI* jest reprezentacją graficzną dla użytkownika.

2.3.4.4 Diagram klas



Część I: Dokumentacja techniczna

Diagram klas przedstawia strukturę modułu *Auth* w postaci klas, ich atrybutów, metod oraz relacji. Głównym elementem modułu jest klasa *AuthController*, która poza udostępnianiem endpointów inicjalizuje proces autoryzacji i uwierzytelnienia. Klasa ta wykorzystuje interfejsy *IUserService* oraz *ILangService*. Klasy *WebSecurityConfig*, *AuthEntryPointJWT*, *AuthTokenFilter*, *UserDetailServiceImpl*, *UserDetailsImpl* ściśle korzystają z framework'a Spring, a konkretnie z modułu Security w celu zapewnienia odpowiedniego poziomu bezpieczeństwa.

Klasa *AuthService* zawiera wszystkie metody udostępniane przez interfejs *IAuthService*, aby inne moduły mogły wykorzystywać potrzebne funkcjonalności modułu *Auth*.

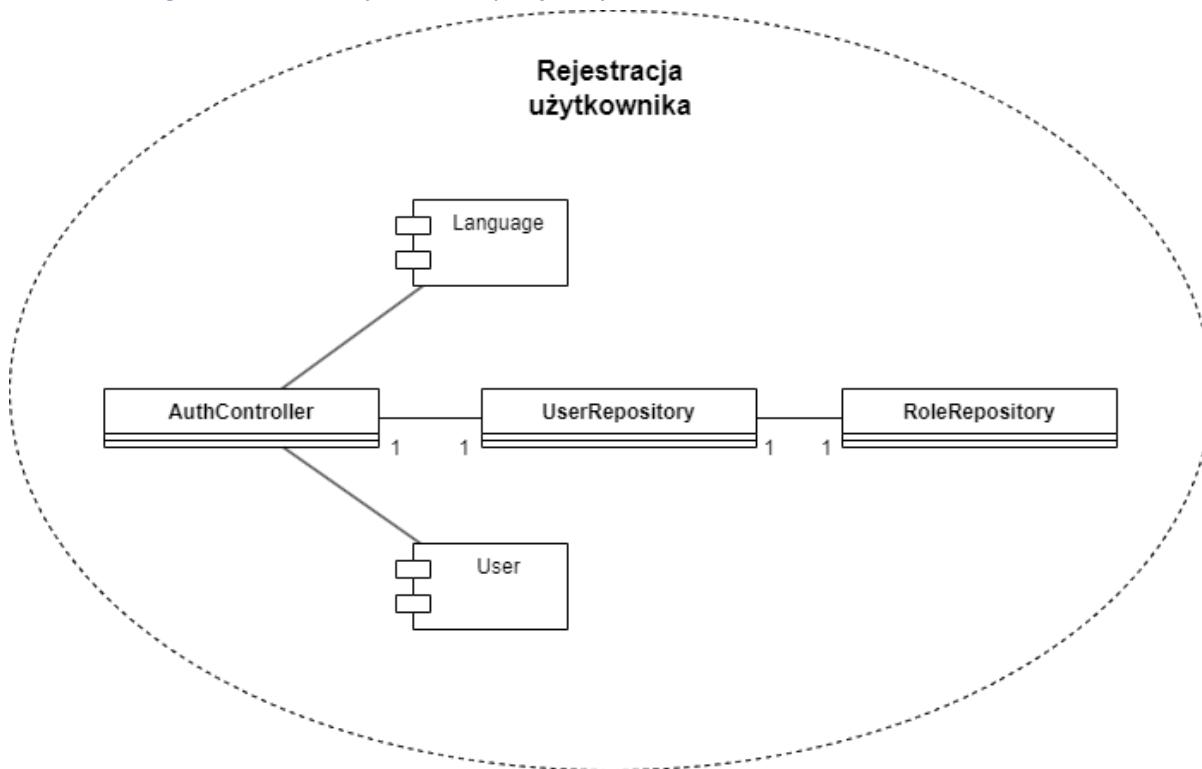
2.3.4.5 Diagram strukturalny (struktur połączonych)

Diagram przedstawia kooperację klas modułu *Auth* podczas rejestracji użytkownika. W tym procesie wykorzystywane są również moduły *Language* oraz *User*.

Po przejściu na endpoint odpowiedzialny za rejestrację, sprawdzana jest baza danych. Jeśli taki użytkownik nie istnieje już w systemie, rejestracja zakończy się pomyślnie i nowy użytkownik zostaje zapisany w bazie danych.

2.3.4.6 Diagram sekwencji - rejestracja

Scenariusz przypadku użycia - rejestracja

Nazwa	Rejestracja
Poziom ważności	Wysoki
Typ przypadku użycia	Niezbędny
Aktorzy	Użytkownik
Krótki opis	Rejestracja nowego użytkownika
Warunki wstępne	Użytkownik nie może być zalogowany
Warunki końcowe	Stworzenie nowego użytkownika
Główny przepływ zdarzeń	1. Użytkownik wypełnia formularz rejestracji

Część I: Dokumentacja techniczna

	2. System wyświetla formularz logowania
Alternatywne przepływy zdarzeń	2a. System wyświetla komunikat o niepowodzeniu rejestracji
Specjalne wymagania	1. System nie powinien przetwarzać żądania dłużej niż 2 sekundy

Diagram sekwencji – rejestracja

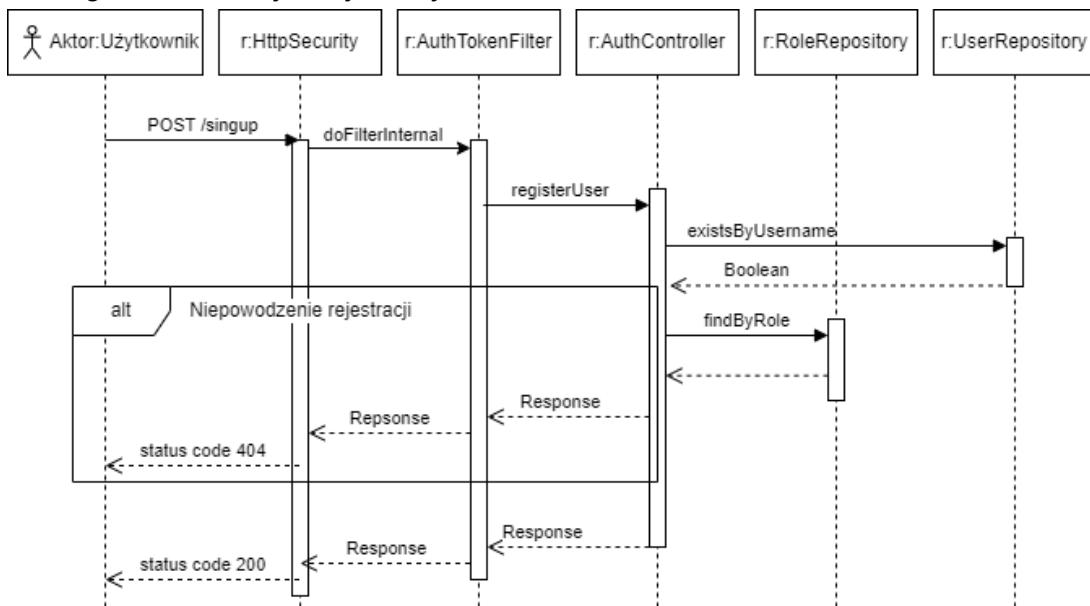


Diagram przedstawia interakcje pomiędzy obiektami *HttpSecurity*, *AuthTokenFilter*, *AuthController*, *RoleRepository*, *UserRepository*. Obiekty wspólnie realizują proces rejestracji użytkownika w systemie.

Obiekty *Repository* odpowiadają za komunikację z bazą danych, a obiekty *HttpSecurity*, *AuthTokenFilter* oraz *AuthController* odpowiadają za logikę procesu rejestracji.

2.3.4.7 Diagram Sekwencji - logowanie

Scenariusz przypadku użycia - logowanie

Nazwa	Logowanie
Poziom ważności	Wysoki
Typ przypadku użycia	Niezbędny
Aktorzy	Użytkownik
Krótki opis	Logowanie istniejącego użytkownika
Warunki wstępne	Użytkownik nie może być zalogowany
Warunki końcowe	Dostęp do systemu
Główny przepływ zdarzeń	1. Użytkownik wypełnia formularz logowania 2. Przejście do systemu

Część I: Dokumentacja techniczna

Alternatywne przepływy zdarzeń	2a. System wyświetla komunikat o niepowodzeniu logowania
Specjalne wymagania	1. System nie powinien przetwarzać żądania dłużej niż 2 sekundy

Diagram sekwencji - logowanie

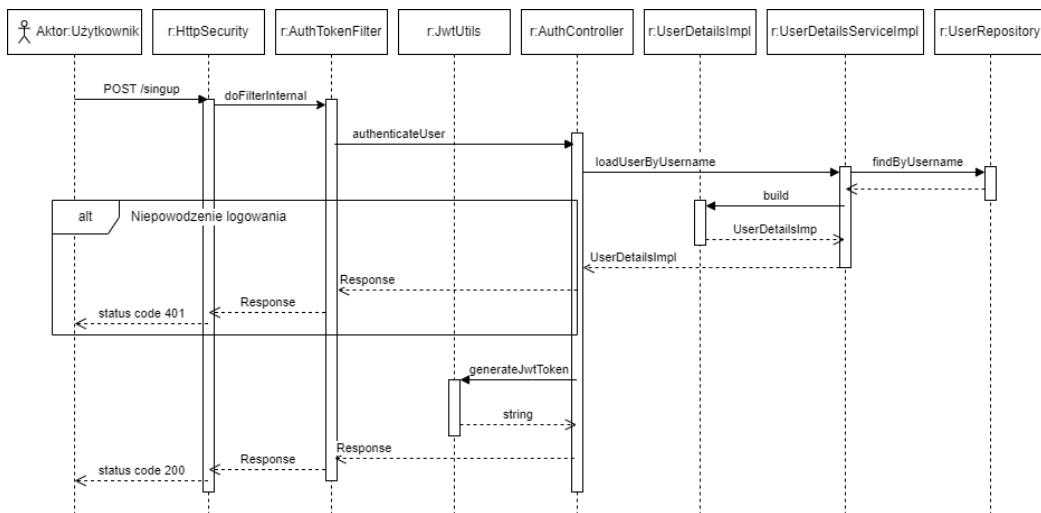


Diagram przedstawia interakcje pomiędzy obiektami *HttpSecurity*, *AuthTokenFilter*, *JwtUtils*, *AuthController*, *UserDetailsImpl*, *UserDetailsServiceImpl*, *UserRepository*. Obiekty wspólnie realizują proces logowania użytkownika do systemu.

Obiekt *UserRepository* odpowiada za komunikację z bazą danych, *JwtUtils* za operacje na tokenie JWT, *AuthController*, *UserDetailsImpl*, *UserDetailsServiceImpl*, *HttpSecurity*, *AuthTokenFilter* odpowiadają za logikę oraz bezpieczeństwo.

2.3.4.8 Diagram sekwencji - pobranie zasobów użytkownika

Scenariusz przypadku użycia – pobieranie danych użytkownika

Nazwa	Pobieranie danych użytkownika
Poziom ważności	Średni
Typ przypadku użycia	Ważny
Aktorzy	Użytkownik
Krótki opis	Pobranie informacji o użytkowniku z systemu
Warunki wstępne	Użytkownik musi być zalogowany
Warunki końcowe	Wyświetlenie informacji o użytkowniku
Główny przepływ zdarzeń	1. Użytkownik przechodzi na stronę profilu

Część I: Dokumentacja techniczna

	2. System wyświetla informacje o użytkowniku
Alternatywne przepływy zdarzeń	2a. System wyświetla komunikat o niepoprawnych danych uwierzytelniających
Specjalne wymagania	1. System nie powinien przetwarzać żądania dłużej niż 2 sekundy

Diagram sekwencji - pobieranie danych użytkownika

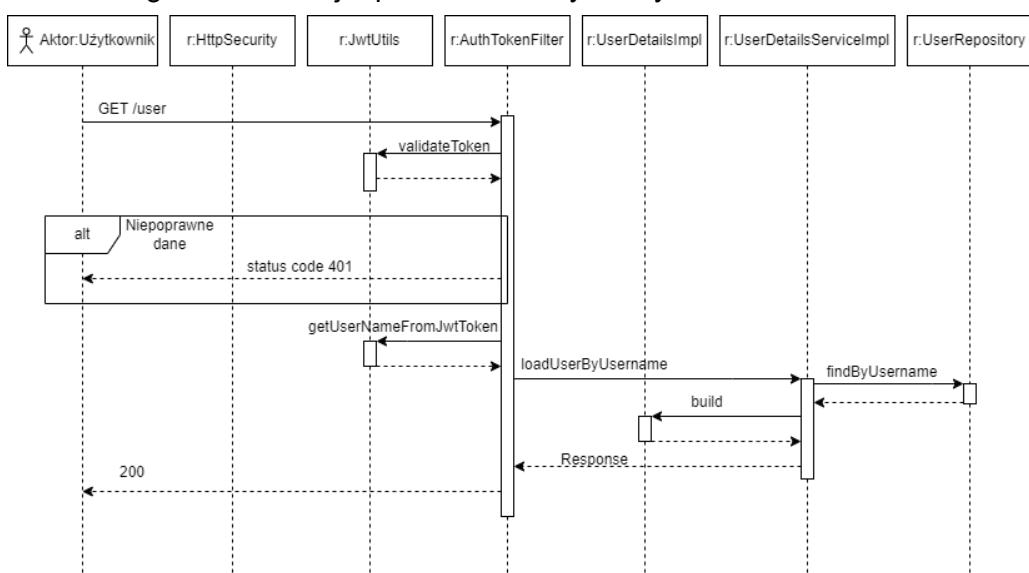


Diagram przedstawia interakcje pomiędzy obiektami *HttpSecurity*, *AuthTokenFilter*, *UserDetailsImpl*, *UserDetailsServiceImpl*, *UserRepository*. Obiekty wspólnie realizują proces pobierania danych użytkownika z systemu.

Obiekt *UserRepository* odpowiada za komunikację z bazą danych, *HttpSecurity*, *AuthTokenFilter*, *UserDetailsImpl*, *UserDetailsServiceImpl* odpowiadają za logikę oraz uwierzytelnienie użytkownika.

Część I: Dokumentacja techniczna

2.3.4.9 Diagram przeglądu interakcji

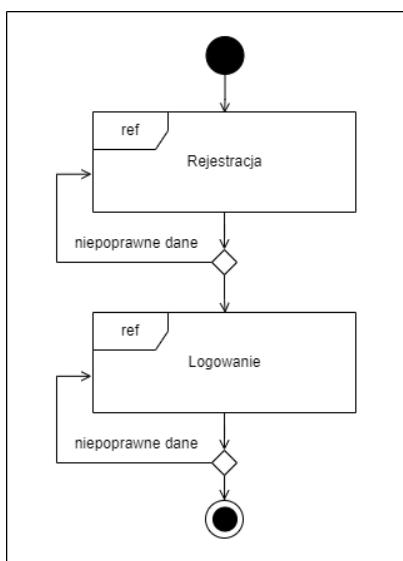


Diagram przedstawia przepływ sterowania pomiędzy rejestracją a logowaniem. Aby użytkownik mógł się zalogować musi najpierw przejść proces rejestracji. Rejestracja i logowanie na powyższym diagramie odnoszą się do diagramu sekwencji – rejestracja oraz diagramu sekwencji – logowanie.

2.3.4.10 Diagram czynności - rejestracja

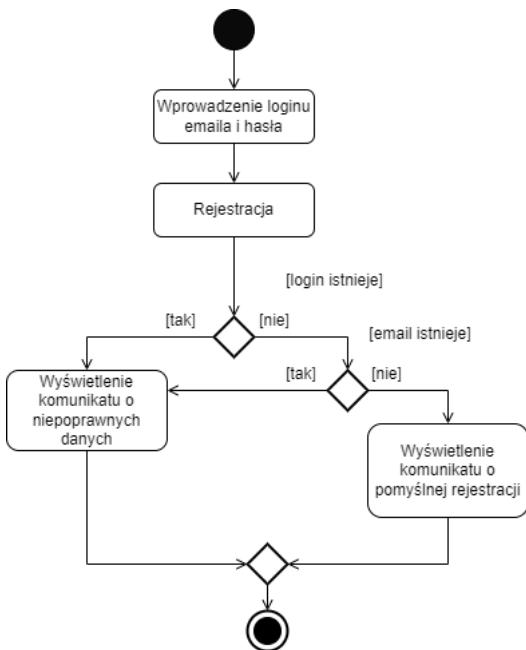


Diagram czynności przedstawia proces rejestracji, który obejmuje wypełnienie formularza przez użytkownika oraz weryfikację danych. W zależności od podanych danych system wyświetli komunikat o powodzeniu lub niepowodzeniu rejestracji. W przypadku pomyślnej rejestracji system przeniesie użytkownika do formularza logowania, w innym wypadku użytkownik zostanie poproszony o ponowne wypełnienie formularza rejestracji.

2.3.4.11 Diagram czynności – logowanie

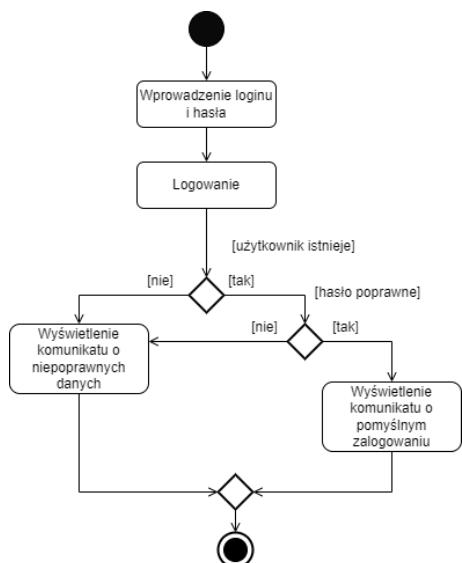


Diagram czynności przedstawia proces logowania, który obejmuje wypełnienie formularza przez użytkownika oraz weryfikację danych. W zależności od podanych danych system wyświetli komunikat o powodzeniu lub niepowodzeniu logowania. W przypadku pomyślnego logowania system wpuszcza użytkownika do systemu, w innym wypadku użytkownik zostanie poproszony o ponowne wypełnienie formularza.

2.3.4.12 Diagram maszyny stanowej (klasa AuthController)

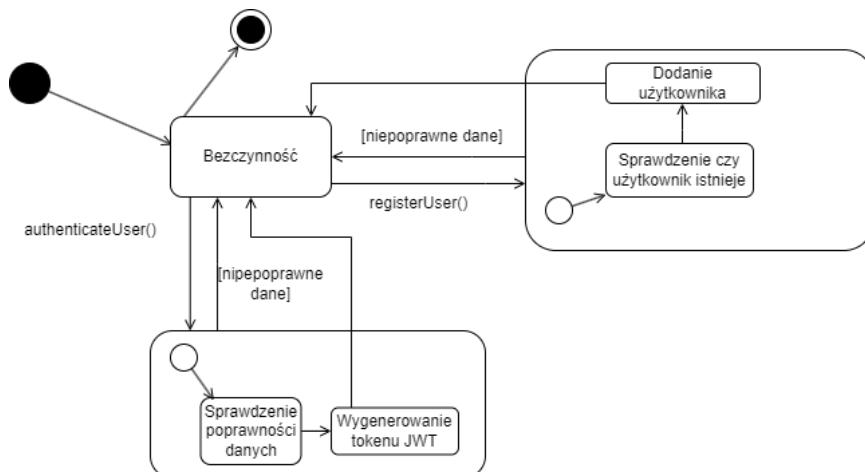


Diagram przedstawia zmianę stanu obiektu `AuthController`. Obiekt domyślnie jest w stanie bezczynności. Po wejściu na endpoint odpowiedzialny za rejestrację wykonywana jest metoda `registerUser()`, która przechodzi przez cały proces rejestracji, w którym obiekt osiąga stany takie jak „Sprawdzenie czy użytkownik istnieje” oraz „Dodanie użytkownika”. Po wejściu na endpoint odpowiedzialny za logowanie wykonywana jest metoda `authenticateUser()`, która przechodzi przez cały proces logowania, w którym obiekt osiąga stany takie jak „Sprawdzenie poprawności danych” oraz „Wygenerowanie tokenu JWT”.

Część I: Dokumentacja techniczna

2.3.4.13 Diagram harmonogramowania – rejestracja

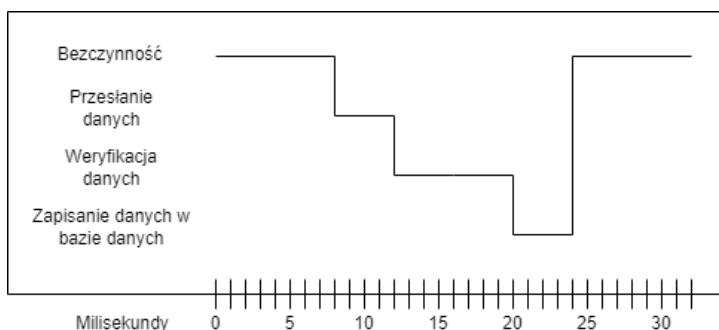


Diagram przedstawia zmiany w systemie w czasie podczas rejestracji użytkownika. Po wypełnieniu formularza przez użytkownika dane są przesyłane do modułu *Auth*, gdzie są weryfikowane, a następnie (gdy są poprawne) zapisywane są do bazy danych, a do użytkownika wysyłana jest odpowiedź.

2.4 Moduł Komunikacji

2.4.1 Cel

Celem modułu jest umożliwienie szybkiej i sprawnej komunikacji między wszystkimi aktorami systemu, aby usprawnić koordynację działań pomocowych. Dzięki temu modułowi organizacje pomocowe, poszkodowani, wolontariusze, darczyńcy oraz przedstawiciele władz będą mogli wymieniać informacje na temat bieżących potrzeb, zasobów i działań.

2.4.2 Założenia

Komunikacja powinna być intuicyjna i zapewniać szybki przepływ informacji.

System powinien umożliwiać zarówno komunikację bezpośrednią między poszczególnymi aktorami, jak i grupową, np. między organizacjami pomocowymi i wolontariuszami.

2.4.3 Wymagania

1. Funkcjonalne:

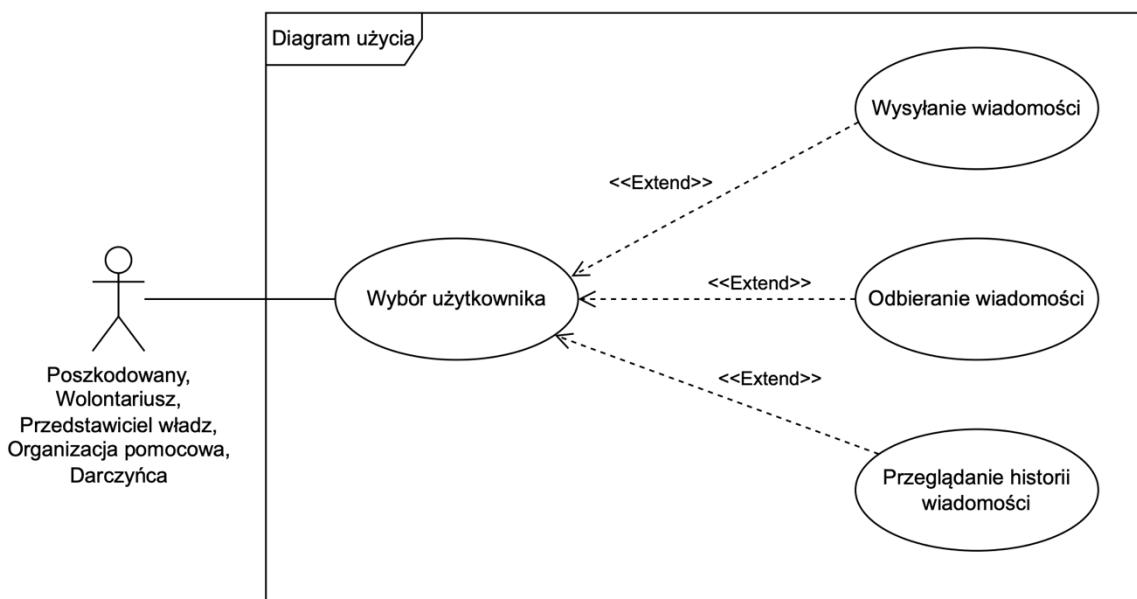
- Wybór użytkownika: System umożliwia wybór użytkownika, z którym użytkownik chce podjąć konwersacje.
- Wysyłanie wiadomości: System umożliwia wysyłanie wiadomości.
- Odbieranie wiadomości: System umożliwia odbieranie wiadomości.
- Przeglądanie historii wiadomości: System przechowuje historię komunikacji, pozwalając na późniejsze przeglądanie i analizę działań podjętych w trakcie akcji pomocowych.

2. Niefunkcjonalne:

- Czas dostarczenia wiadomości: Wiadomości powinny być dostarczane w czasie nie dłuższym niż 3 sekundy od momentu wysłania.
- Intuicyjny interfejs: Interfejs powinien być prosty i intuicyjny, umożliwiający szybkie wysyłanie i odbieranie wiadomości.

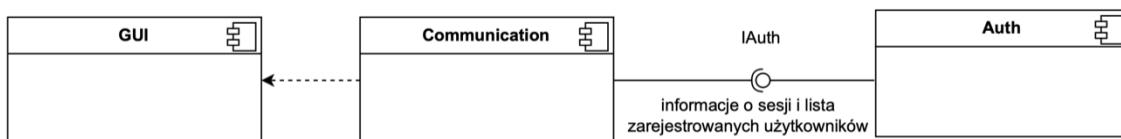
2.4.4 Diagramy

2.4.4.1 Diagram przypadków użycia:



Powyższy diagram ilustruje relacje między aktorami a przypadkami użycia, wskazując, kto i w jaki sposób korzysta z systemu. Dostęp do naszego modułu ma każdy użytkownik. Wszyscy użytkownicy współdziałają z systemem przez wybór użytkownika, co jest ich podstawowym zadaniem. Następnie mogą wysyłać, odbierać oraz przeglądać wiadomości. Przypadki użycia te rozszerzają podstawowy przypadek użycia.

2.4.4.2 Diagram komponentów:



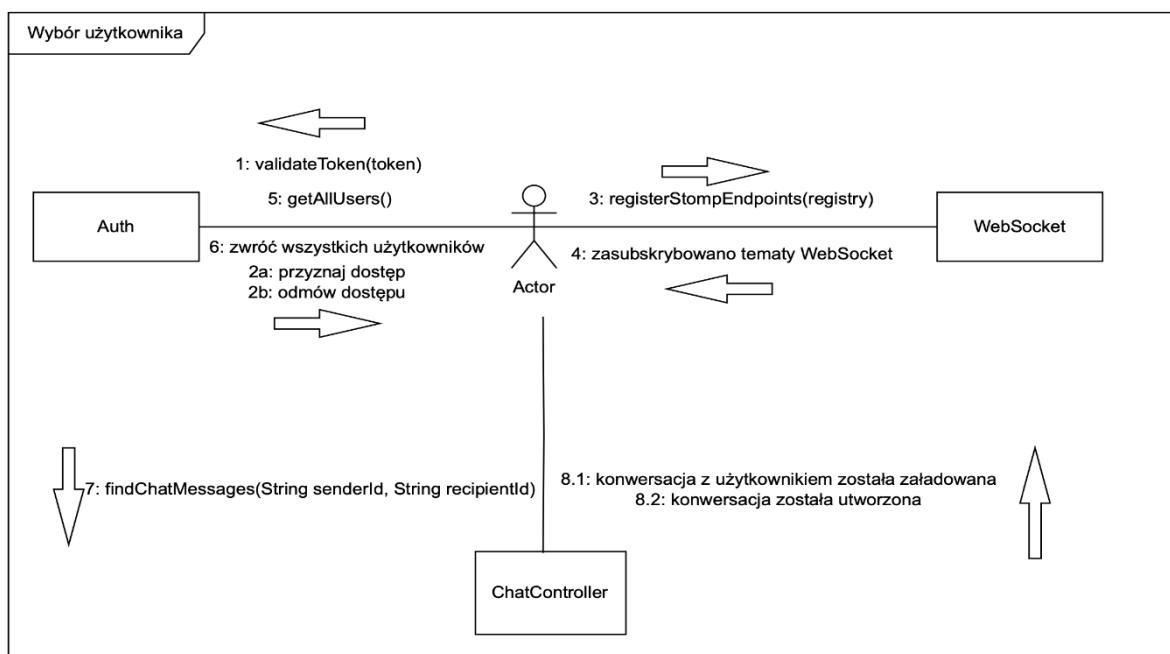
Powyższy diagram przedstawia strukturę systemu jako zestaw komponentów, ich interfejsów oraz relacji między nimi. Nasz moduł pobiera informacje o sesji i listę zarejestrowanych użytkowników z modułu Auth.

Scenariusz – Wybór użytkownika:

Nazwa	Wybór użytkownika
Poziom ważności	Średni
Typ przypadków użycia	Wybór użytkownika do konwersacji.
Aktorzy	Poszkodowany, Wolontariusz, Przedstawiciel władz, Organizacja pomocowa, Darczyńca.
Krótki opis	Użytkownik przy pomocy aplikacji wybiera użytkownika.
Warunki wstępne	Użytkownik musi być zalogowany.

Warunki końcowe	System tworzy konwersacje między użytkownikiem który ją utworzył a dodanym.
Główny przepływ zdarzeń	<ol style="list-style-type: none"> 1. Walidacja użytkownika. 2a. Przyznanie dostępu. 3. Rejestracja w Websocket. 4. Subskrypcja Websocket. 5. Pobranie wszystkich użytkowników. 6. Lista użytkowników zostaje wyświetlona. 7. Wybór użytkownika. 8a. Załadowanie konwersacji.
Alternatywny przepływ zdarzeń	<ol style="list-style-type: none"> 2b. Odmowa dostępu 8b. Utworzenie konwersacji

2.4.4.3 Diagram komunikacji – Wybór użytkownika:



Powyższy diagram pokazuje, jak obiekty w systemie wymieniają komunikaty w celu realizacji wyboru użytkownika.

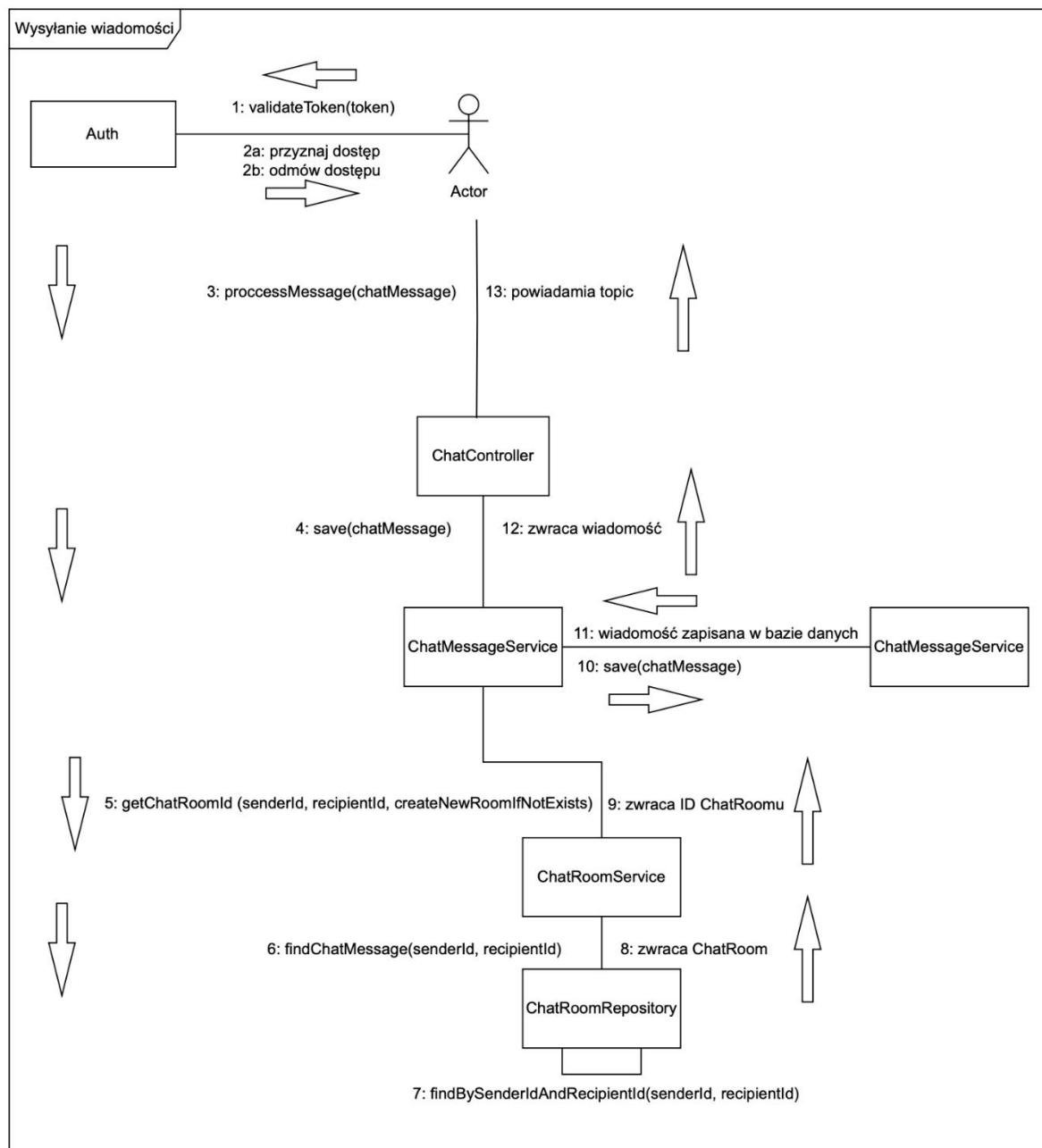
2.4.4.4 Scenariusz – Wysyłanie wiadomości:

Nazwa	Wysyłanie wiadomości
Poziom ważności	Wysoki
Typ przypadków użycia	Wysyłanie wiadomości, komunikacja w ramach konwersacji.
Aktorzy	Poszkodowany, Wolontariusz, Przedstawiciel władz,

Część I: Dokumentacja techniczna

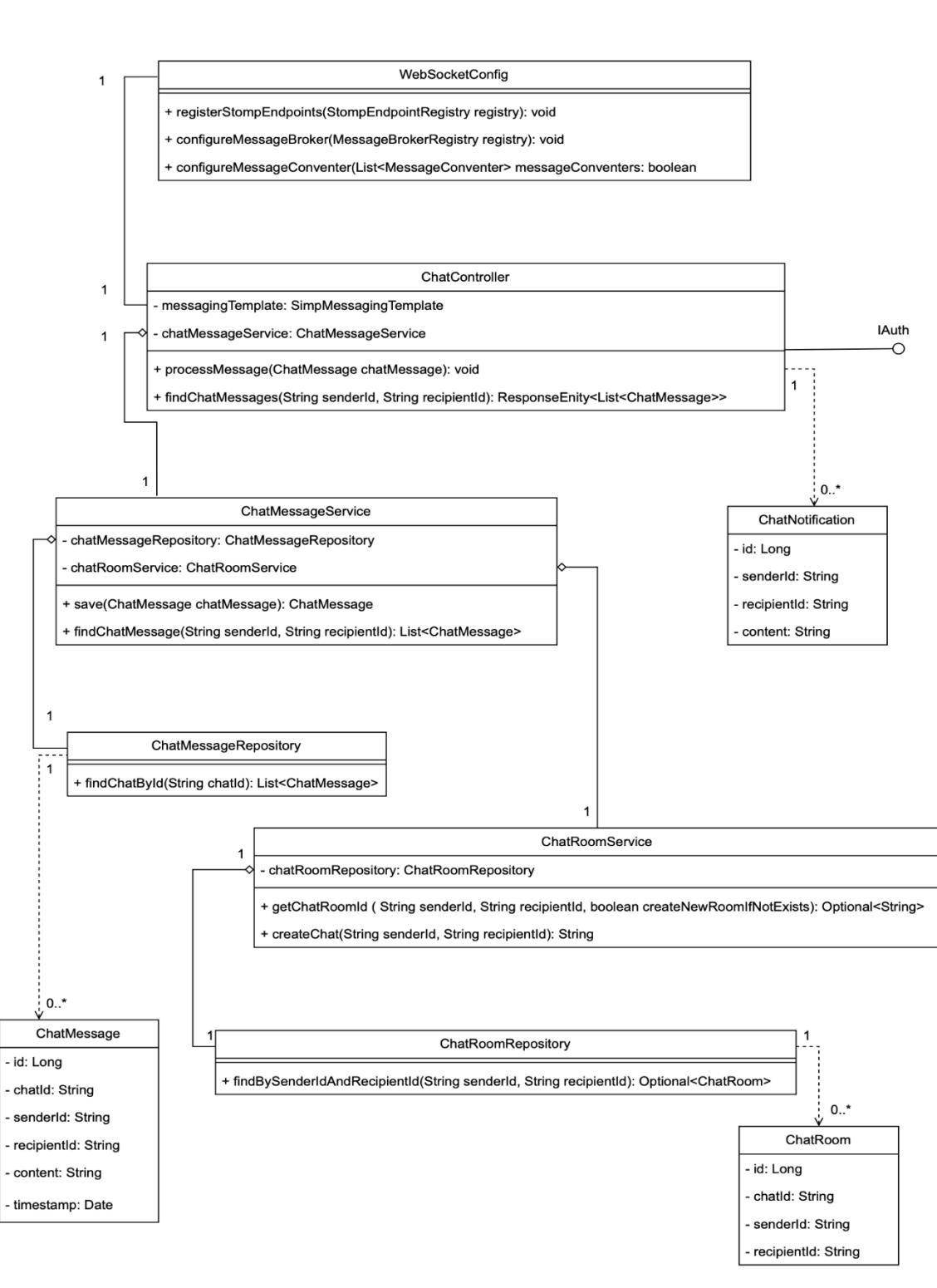
	Organizacja pomocowa, Darczyńca.
Krótki opis	Użytkownik przy pomocy aplikacji wysyła wiadomość widoczną dla wszystkich uczestników konwersacji.
Warunki wstępne	Musi istnieć użytkownik do którego użytkownik chce wysłać wiadomość.
Warunki końcowe	System wyświetla wiadomość na chacie dla wszystkich użytkowników danej konwersacji.
Główny przepływ zdarzeń	<ol style="list-style-type: none"> 1. Walidacja użytkownika 2a. Przyznanie dostępu 3. Wyślij wiadomość 4. System zapisuje wiadomość 5. System szuka id konwersacji 6. System znajduje wiadomość 7. System szuka konwersacji 8. System zwraca konwersacje 9. System zwraca id konwersacji 10. System zapisuje wiadomość do bazy danych 11. Wiadomość zapisana 12. Wiadomość zostaje zwrócona 13. Powiadomienie websocketu
Alternatywny przepływ zdarzeń	2b. Odmów dostępu

2.4.4.5 Diagram komunikacji – Wysyłanie wiadomości:



Powyższy diagram pokazuje, jak obiekty w systemie wymieniają komunikaty w celu realizacji wysłania wiadomości.

2.4.4.6 Diagram klas:



Powyższy diagram pokazuje strukturę modułu, definiując jego klasy, atrybuty, metody oraz relacje między nimi. Zawiera następujące elementy:

1. WebSocketConfig

→ **Opis:** Konfiguracja WebSocket w aplikacji.

→ **Metody:**

- registerStompEndpoints: Rejestruje punkty końcowe STOMP.
- configureMessageBroker: Konfiguruje broker wiadomości.
- configureMessageConverter: Konfiguruje konwertery wiadomości.

2. ChatController→ **Opis:** Kontroler obsługujący wiadomości czatu.→ **Atrybuty:**

- messagingTemplate: Służy do wysyłania wiadomości.
- chatMessageService: Serwis do obsługi wiadomości czatu.

→ **Metody:**

- processMessage: Przetwarza przychodzące wiadomości.
- findChatMessages: Znajduje wiadomości czatu.

3. ChatMessageService→ **Opis:** Serwis do zarządzania wiadomościami czatu.→ **Atrybuty:**

- chatMessageRepository: Repozytorium wiadomości czatu.
- chatRoomService: Serwis do obsługi pokojów czatu.

→ **Metody:**

- save: Zapisuje wiadomość czatu.
- findChatMessage: Znajduje wiadomość czatu.

4. ChatMessageRepository→ **Opis:** Repozytorium do zarządzania wiadomościami czatu.→ **Metody:**

- findChatById: Znajduje wiadomość czatu na podstawie ID.

5. ChatRoomService→ **Opis:** Serwis do zarządzania pokojami czatu.→ **Atrybuty:**

- chatRoomRepository: Repozytorium pokojów czatu.

→ **Metody:**

- getChatRoomId: Pobiera ID pokoju czatu.
- createChat: Tworzy nowy pokój czatu.

6. ChatRoomRepository→ **Opis:** Repozytorium do zarządzania pokojami czatu.→ **Metody:**

- findBySenderIdAndRecipientId: Znajduje pokój czatu na podstawie ID nadawcy i odbiorcy.

7. ChatNotification→ **Opis:** Powiadomienie czatu.

→ **Atrybuty:**

- id: ID powiadomienia.
- senderId: ID nadawcy.
- recipientId: ID odbiorcy.
- content: Zawartość powiadomienia.

8. ChatMessage→ **Opis:** Wiadomość czatu.→ **Atrybuty:**

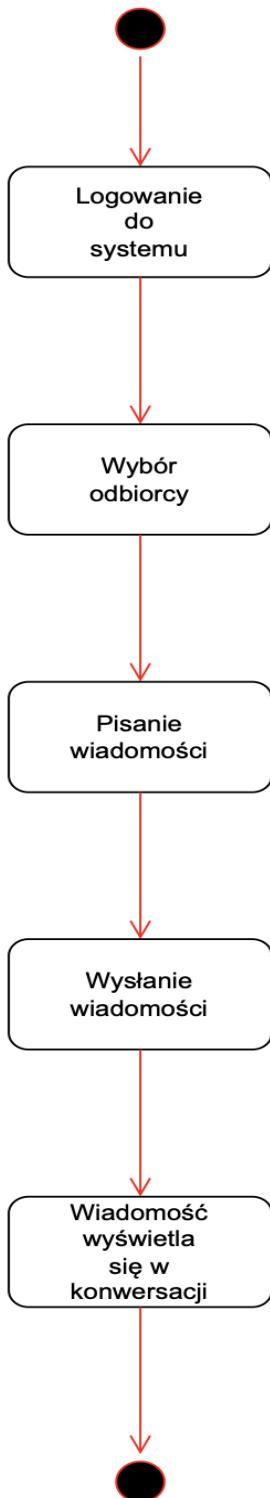
- id: ID wiadomości.
- chatId: ID czatu.
- senderId: ID nadawcy.
- recipientId: ID odbiorcy.
- content: Treść wiadomości.
- timestamp: Znacznik czasowy wiadomości.

9. ChatRoom→ **Opis:** Pokój czatu.→ **Atrybuty:**

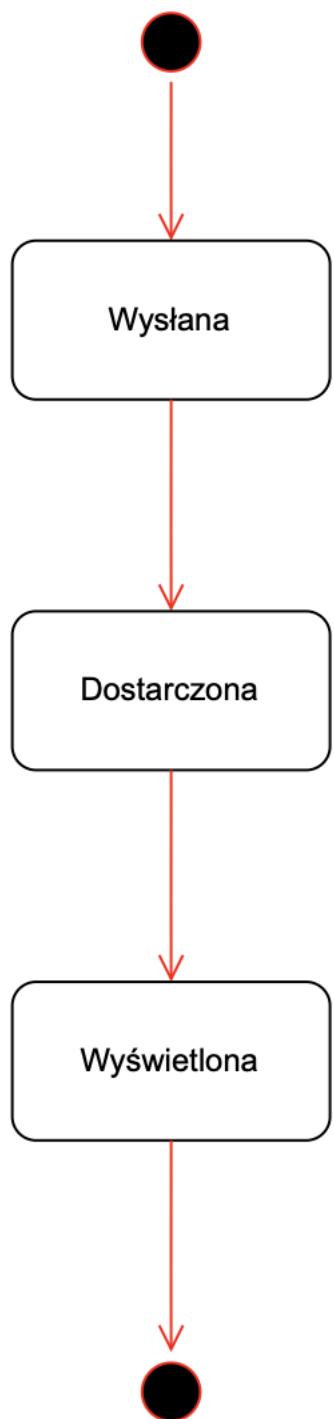
- id: ID pokoju czatu.
- chatId: ID czatu.
- senderId: ID nadawcy.
- recipientId: ID odbiorcy.

Część I: Dokumentacja techniczna

2.4.4.7 Diagram czynności – Wysyłanie wiadomości:

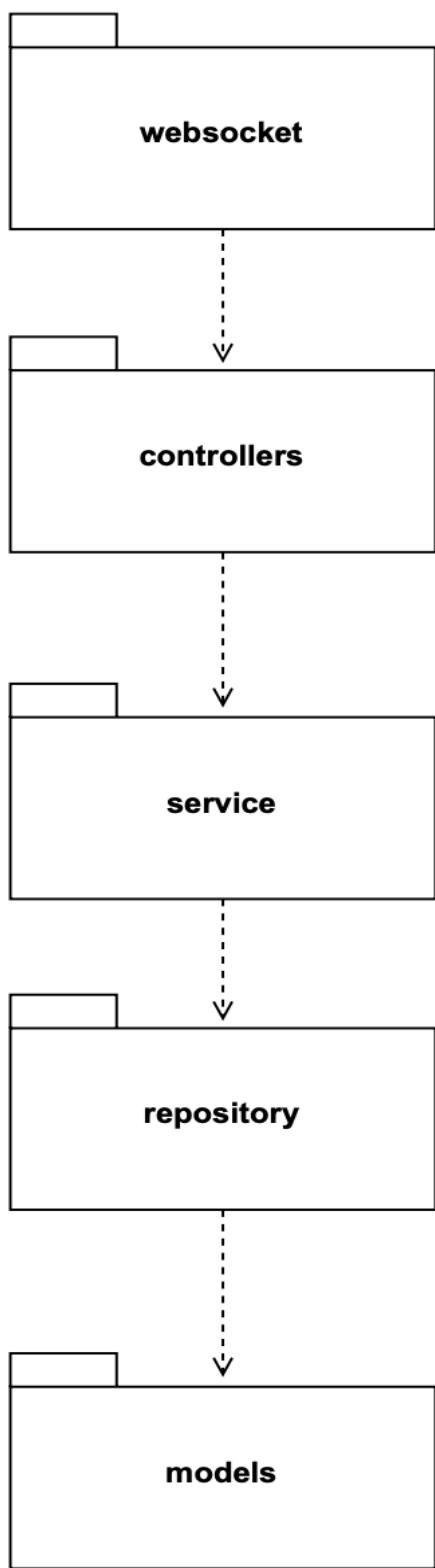


Powyższy diagram pokazuje przepływ pracy w systemie. Są to kolejne czynności, które musi podjąć użytkownik by wysłać wiadomość.

2.4.4.8 *Diagram maszyny stanowej:*

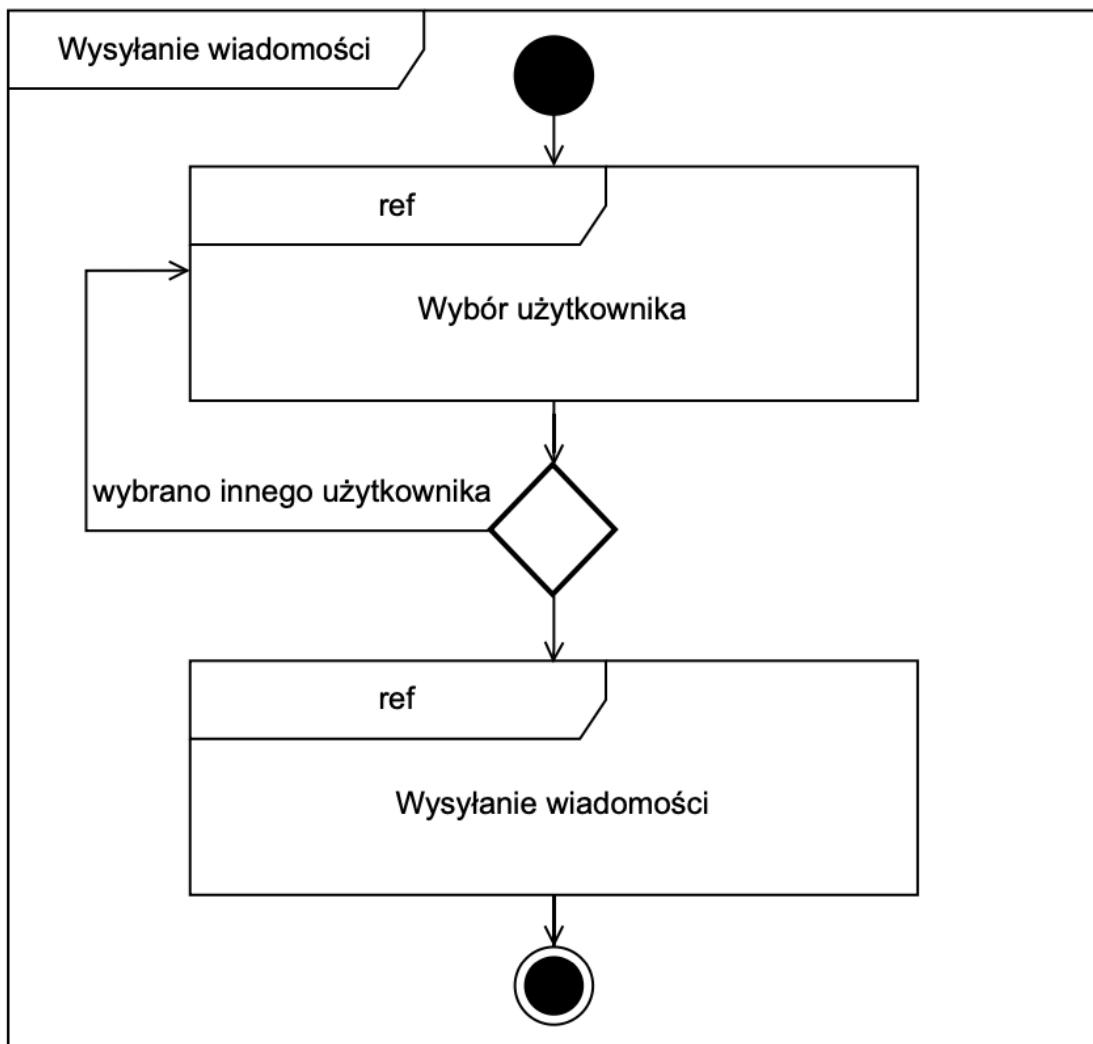
Powyższy diagram pokazuje możliwe stany obiektu wiadomości w systemie oraz przejścia między nimi w odpowiedzi na zdarzenia.

2.4.4.9 *Diagram pakietów:*

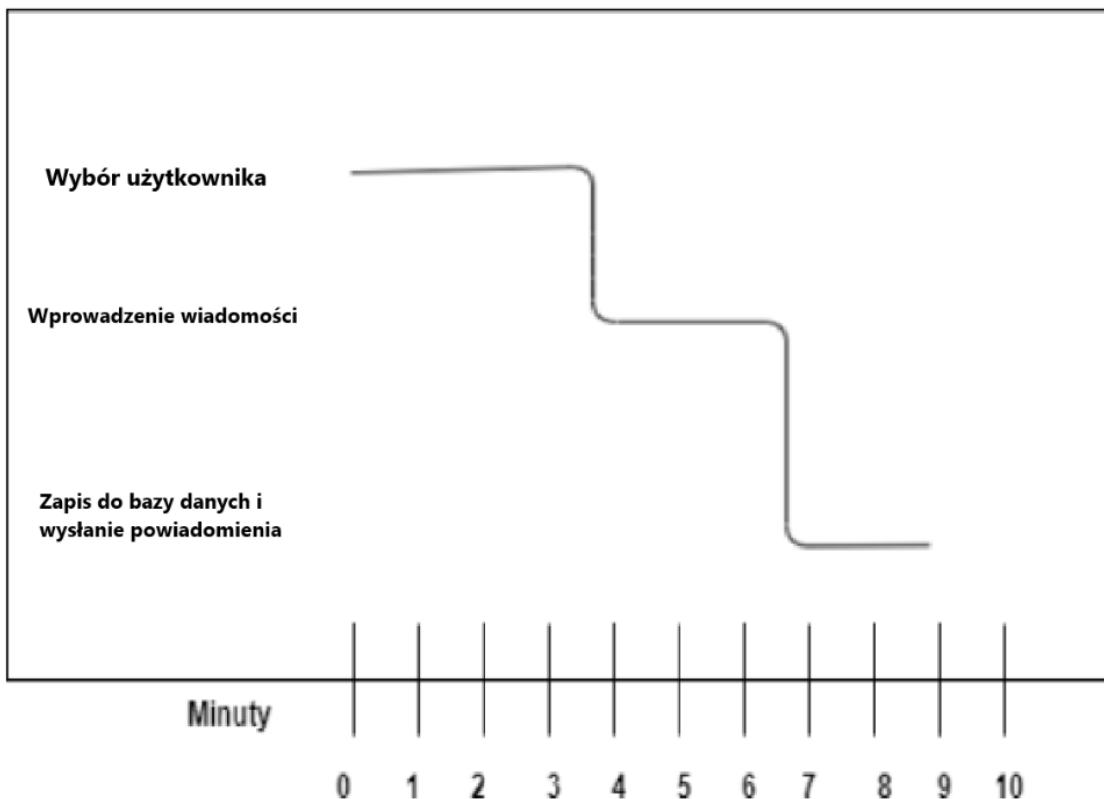


Powyższy diagram służy do pokazania logicznego podziału modułu na pakiety oraz relacji między nimi.

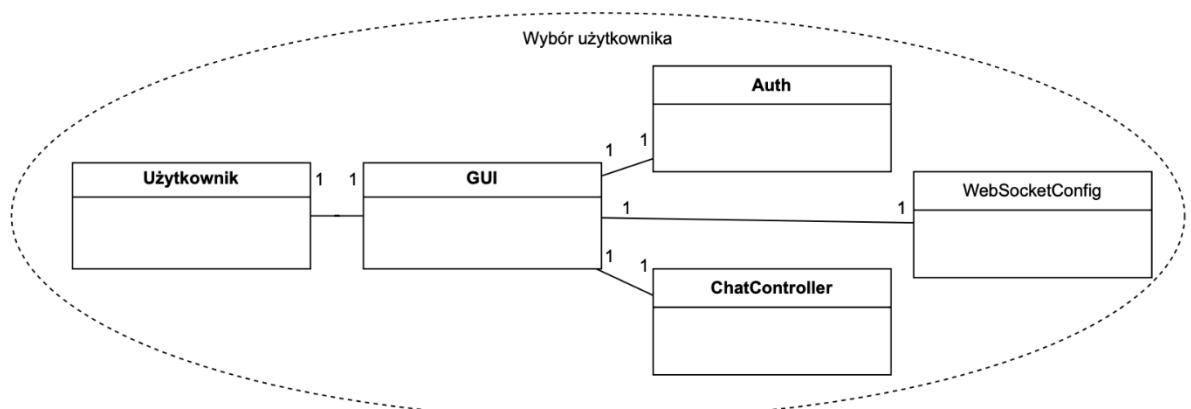
2.4.4.10 Diagram przeglądu interakcji:



Powyższy diagram łączy cechy diagramu czynności i diagramu sekwencji, aby pokazać interakcje między różnymi elementami systemu w kontekście wysyłania wiadomości.

2.4.4.11 *Diagram harmonogramowania:*

Powyższy diagram przedstawienia planu działań w czasie. Oś pozioma przedstawia czas trwania projektu, a zadania są reprezentowane jako poziome paski wskazujące daty rozpoczęcia i zakończenia.

2.4.4.12 *Diagram strukturalny:*

Powyższy diagram przedstawia układ i wzajemne powiązania poszczególnych obiektów podczas wykonywania zadania wyboru użytkownika.

2.5 Moduł Map

2.5.1 Cel

Umożliwienie wizualizacji lokalizacji poszkodowanych, punktów dystrybucji pomocy oraz tras dostaw, co ułatwi koordynację pomocy w czasie rzeczywistym.

2.5.2 Założenia

Moduł mapowy powinien być precyzyjny i łatwy w użyciu.

Powinien umożliwiać aktualizacje danych na bieżąco.

2.5.3 Wymagania

1. Funkcjonalne:

- Wyświetlanie lokalizacji wszystkich zgłoszeń: Na mapie zaznaczane będą miejsca, w których znajdują się poszkodowani wymagający pomocy, co pozwoli na szybsze reagowanie.
- Wyświetlanie własnych zgłoszeń: Na mapie zaznaczane będą miejsca, zgłoszone przez poszkodowanego, co zwiększy kontrolę nad zgłoszeniami.
- Śledzenie aktualizacji map w czasie rzeczywistym: Moduł będzie automatycznie odświeżać dane dotyczące położenia poszkodowanych, wolontariuszy i punktów dystrybucji.
- Wyświetlanie wszystkich dostępnych punktów dystrybucji pomocy: Mapa będzie pokazywać lokalizacje dostępnych punktów dystrybucji pomocy.
- Wyświetlanie własnych punktów dystrybucji pomocy: Mapa będzie pokazywać lokalizacje wszystkich punktów dystrybucji pomocy danej organizacji.
- Tworzenie zgłoszeń: Poszkodowany będzie miał możliwość utworzenia zgłoszenia z poziomu mapy.

2. Niefunkcjonalne:

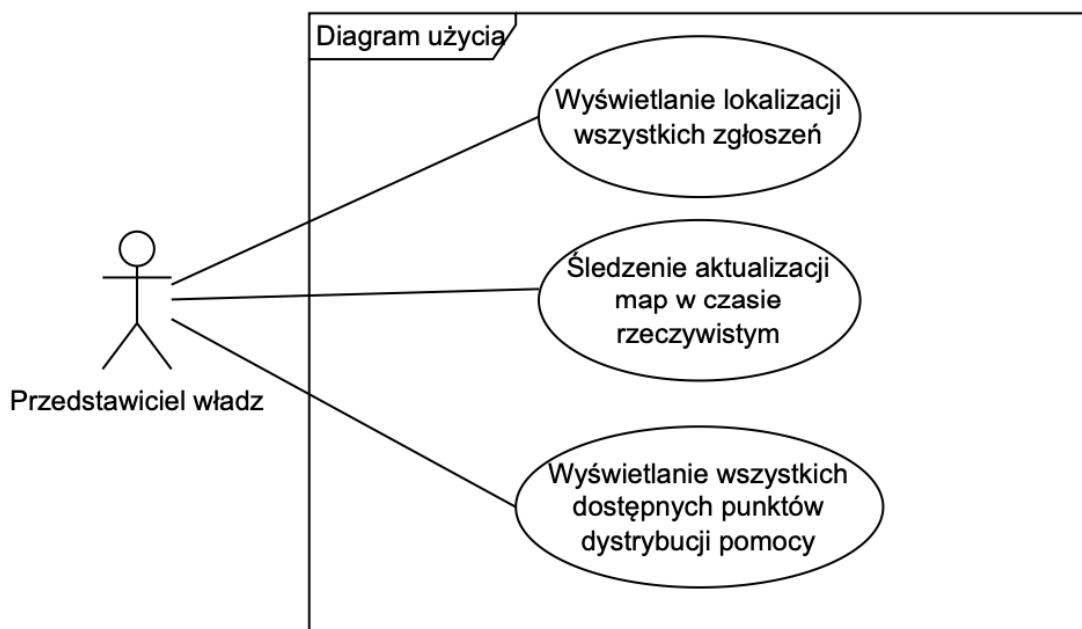
- Czas ładowania: Moduł mapy powinien ładować i wyświetlać dane w ciągu maksymalnie 2 sekund od żądania użytkownika.
- Plynność interakcji: Operacje przybliżania, oddalania i przesuwania mapy powinny odbywać się bez zauważalnych opóźnień.
- Obsługa danych: System powinien efektywnie obsługiwać wyświetlanie dużej liczby punktów danych (np. lokalizacje poszkodowanych, zasobów) bez utraty wydajności.
- Kompatybilność: Przeglądarki internetowe: Moduł mapy powinien być kompatybilny z najpopularniejszymi przeglądarkami (Chrome, Firefox, Safari, Edge) w ich aktualnych wersjach.
- Intuicyjny interfejs: Interfejs użytkownika powinien być prosty w obsłudze, z czytelnymi ikonami i opisami funkcji.

2.5.4 Diagramy

2.5.4.1 Diagram przypadków użycia:

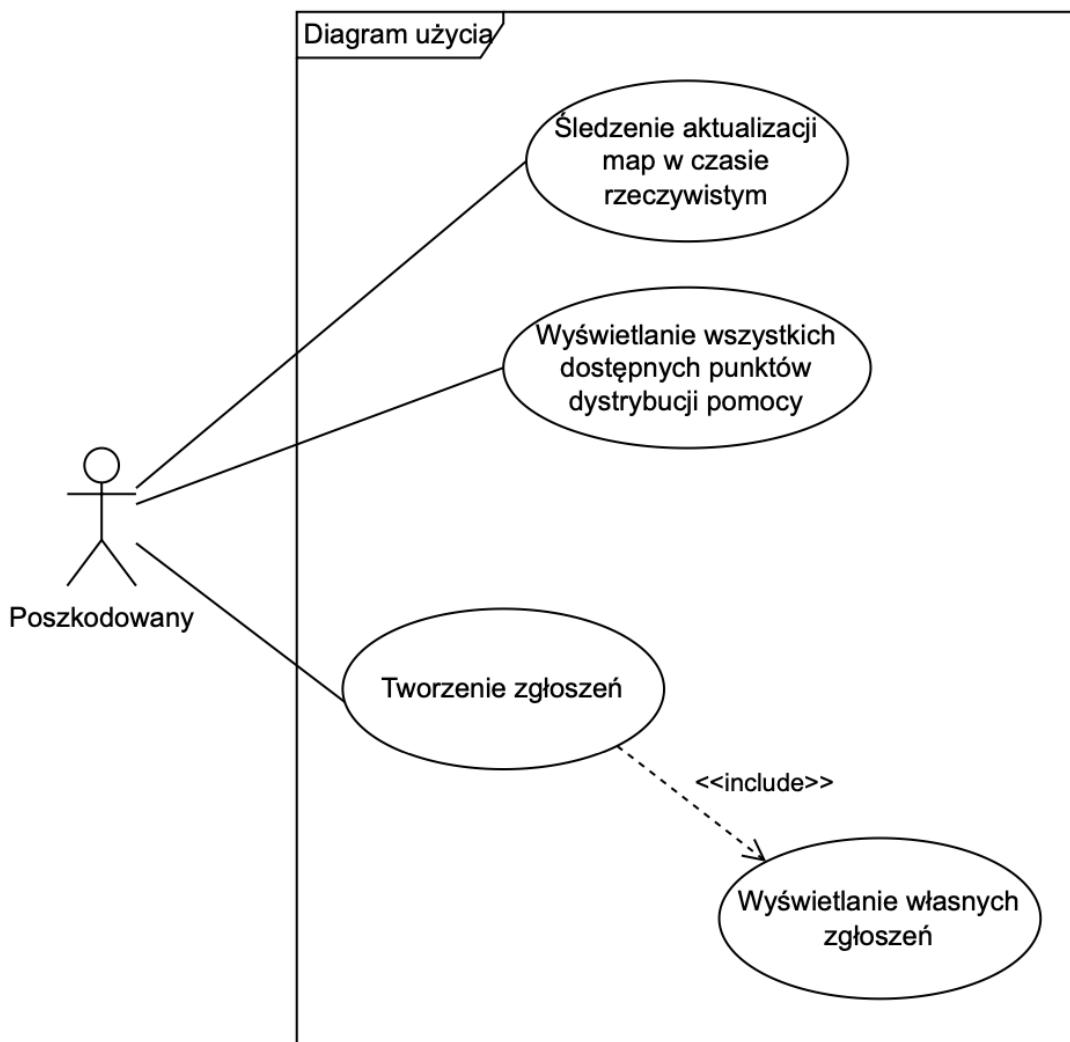
Część I: Dokumentacja techniczna

Poniższe diagramy ilustrują relacje między aktorami a przypadkami użycia, wskazując, kto i w jaki sposób korzysta z systemu.



Pierwszym aktorem jest przedstawiciel władz. Aktor ten ma trzy główne możliwości:

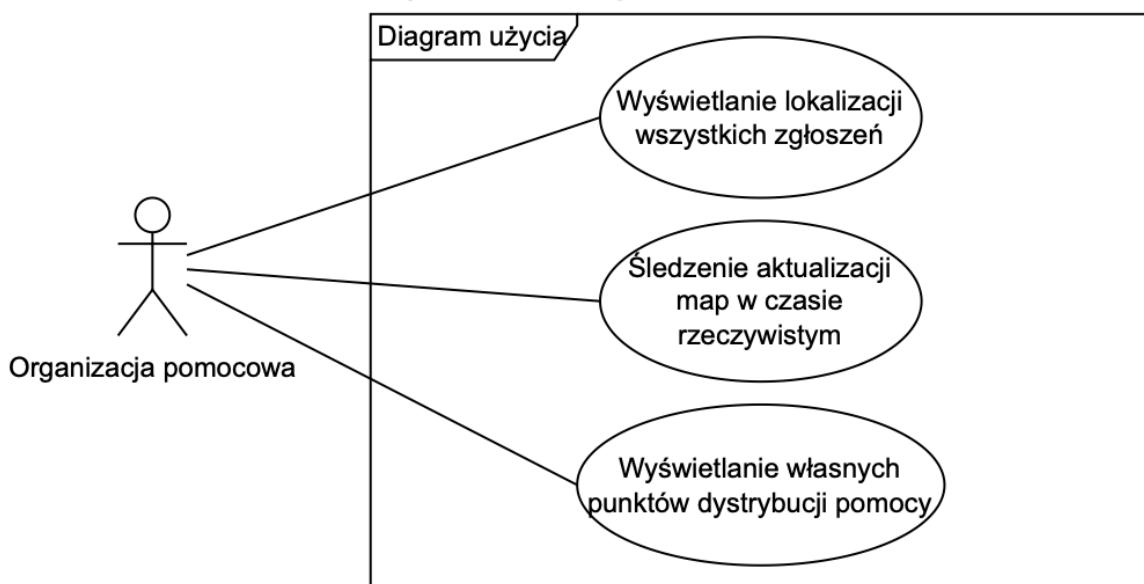
1. **Wyświetlanie lokalizacji wszystkich zgłoszeń:** Przedstawiciel władz może przeglądać lokalizacje wszystkich zgłoszeń, co umożliwia mu uzyskanie informacji na temat zgłoszonych incydentów.
2. **Śledzenie aktualizacji map w czasie rzeczywistym:** Przedstawiciel władz ma możliwość śledzenia aktualizacji map, które dzieją się w czasie rzeczywistym. Dzięki temu może monitorować zmieniającą się sytuację i podejmować decyzje na podstawie najnowszych danych.
3. **Wyświetlanie wszystkich dostępnych punktów dystrybucji pomocy:** Przedstawiciel władz może również wyświetlać wszystkie dostępne punkty dystrybucji pomocy. To pozwala na lepsze zarządzanie zasobami i koordynację pomocy na obszarze objętym kryzysem.



Drugim aktorem jest poszkodowany. Aktor ten ma trzy główne możliwości:

1. **Śledzenia aktualizacji map w czasie rzeczywistym:** Aktor może monitorować zmiany i aktualizacje na mapie w czasie rzeczywistym, co pozwala na szybkie reagowanie na zmieniające się sytuacje.
2. **Wyświetlania wszystkich dostępnych punktów dystrybucji pomocy:** Aktor może przeglądać wszystkie dostępne punkty dystrybucji pomocy, co zwiększa szansę na znalezienie potrzebnej pomocy.
3. **Tworzenia zgłoszeń:** Aktor ma także możliwość tworzenia nowych zgłoszeń, co jest kluczowe dla komunikowania potrzeb i problemów.
4. **Wyświetlania własnych zgłoszeń:** Aktor może przeglądać swoje wcześniejsze zgłoszenia.

Część I: Dokumentacja techniczna

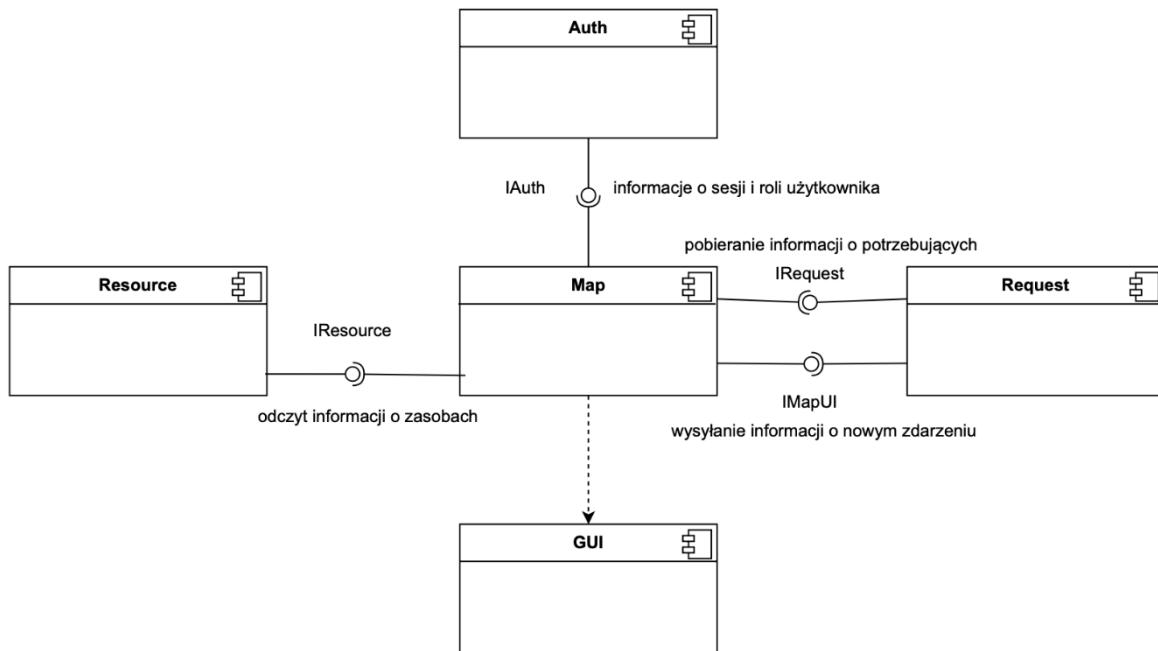


Trzecim aktorem jest organizacja pomocowa. Aktor ten ma trzy główne możliwości:

- Wyświetlanie lokalizacji wszystkich zgłoszeń:** Organizacja pomocowa może przeglądać lokalizacje wszystkich zgłoszeń, co umożliwia mu uzyskanie informacji na temat zgłoszonych incydentów.
- Śledzenie aktualizacji map w czasie rzeczywistym:** Przedstawiciel władz ma możliwość śledzenia aktualizacji map, które dzieją się w czasie rzeczywistym. Dzięki temu może monitorować zmieniającą się sytuację i podejmować decyzje na podstawie najnowszych danych.
- Wyświetlanie własnych punktów dystrybucji pomocy:** Organizacja pomocowa może również wyświetlać wszystkie własne punkty dystrybucji pomocy. To pozwala na lepsze zarządzanie zasobami i koordynację pomocy na obszarze objętym kryzysem.

Część I: Dokumentacja techniczna

2.5.4.2 Diagram komponentów:



Powyższy diagram przedstawia strukturę systemu jako zestaw komponentów, ich interfejsów oraz relacji między nimi. Moduł Map współpracuje z następującymi modułami:

1. Moduł Auth

- **Komunikacja:** Moduł Map komunikuje się z modułem Auth poprzez interfejs `IAuth`, który dostarcza informacje o sesji i roli użytkownika.
- **Funkcja:** Dzięki temu moduł Map może dostosować wyświetlane dane na mapie w zależności od uprawnień użytkownika.

2. Moduł Request

- **Komunikacja:** Moduł Map pobiera informacje o potrzebujących z modułu Request poprzez interfejs `IRequest`.
- **Funkcja:** To pozwala modułowi Map na wyświetlanie zgłoszeń na mapie, co umożliwia szybkie zlokalizowanie i reakcję na incydenty.

3. Moduł Resource

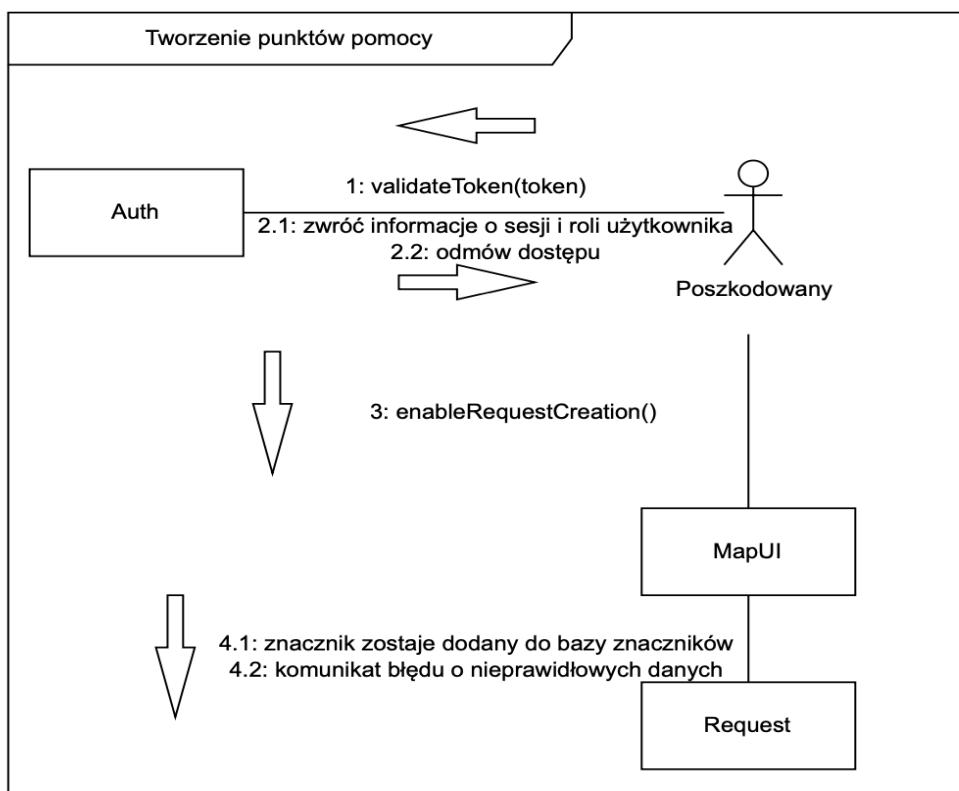
- **Komunikacja:** Moduł Map odczytuje informacje o zasobach z modułu Resource poprzez interfejs `IResource`.
- **Funkcja:** Zapewnia to modułowi Map dokładne dane o dostępnych zasobach, takich jak punkty dystrybucji pomocy, co jest istotne dla właściwego zarządzania zasobami.

2.5.4.3 Scenariusz – Tworzenie zgłoszeń:

Nazwa	Tworzenie zgłoszeń
Poziom ważności	Bardzo wysoki
Typ przypadków użycia	Oznaczenie lokalizacji poszkodowanego.
Aktorzy	Poszkodowany
Krótki opis	Poszkodowany przy pomocy aplikacji zaznacza miejsce gdzie potrzebna jest pomoc.
Warunki wstępne	Użytkownik (poszkodowany) musi mieć dostęp do aplikacji oraz możliwość zaznaczenia lokalizacji na mapie.
Warunki końcowe	System tworzy na mapie lokalizację poszkodowanego.
Główny przepływ zdarzeń	1. Walidacja użytkownika. 2a. Przyznanie dostępu. 3. Zaznaczenie lokalizacji i wypełnienie informacji. 4a. Dodanie znacznika.
Alternatywny przepływ zdarzeń	2b. Odmowa dostępu 4b. Komunikat o błędzie dodawania znacznika

Część I: Dokumentacja techniczna

2.5.4.4 Diagram komunikacji – Tworzenie zgłoszeń:

**Sekwencja Interakcji**

- Poszkodowany** wysyła prośbę do **Auth** o walidację tokenu używając metody `validateToken(token)`.
- Auth** zwraca informacje dotyczące sesji oraz roli użytkownika do **Poszkodowanego**.
- Poszkodowany** wywołuje metodę `enableRequestCreation()`, aby włączyć tworzenie żądania.
- Następnie, do bazy markerów jest dodawany znacznik, a **Request** zostaje stworzony.

Diagram pokazuje proces tworzenia punktów pomocy w systemie, ilustrując interakcje pomiędzy użytkownikami a różnymi komponentami systemu.

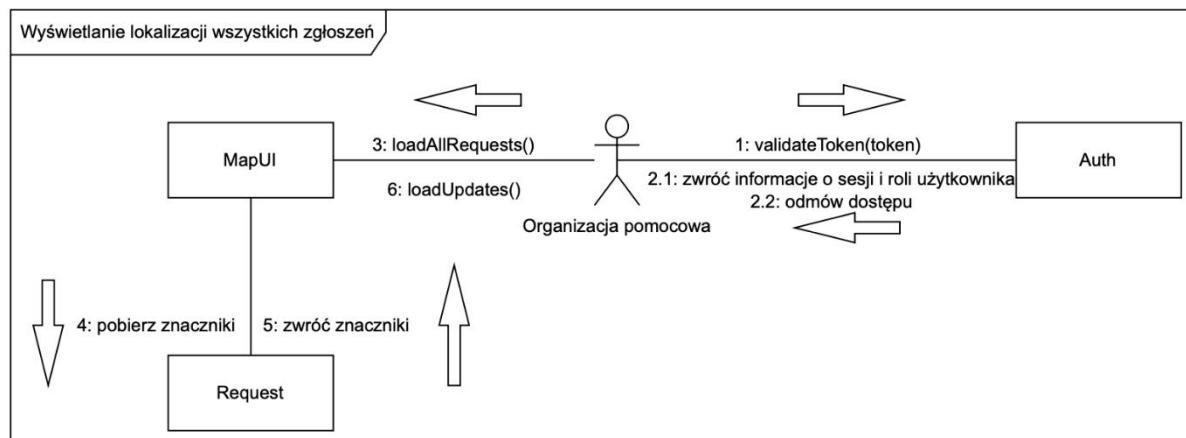
2.5.4.5 Scenariusz – Wyświetlenie lokalizacji wszystkich zgłoszeń:

Nazwa	Wyświetlenie lokalizacji wszystkich zgłoszeń
Poziom ważności	Bardzo wysoki
Typ przypadków użycia	Wyświetlanie lokalizacji osób potrzebujących pomocy, monitorowanie

Część I: Dokumentacja techniczna

	sytuacji kryzysowej.
Aktorzy	Organizacja pomocowa, Przedstawiciel władz
Krótki opis	Organizacja pomocowa i przedstawiciel władz przy pomocy aplikacji wyświetla miejsca gdzie potrzebna jest pomoc.
Warunki wstępne	Musi istnieć co najmniej jedna lokalizacja poszkodowanego.
Warunki końcowe	System wyświetla na mapie wszystkie punkty wymagające pomocy.
Główny przepływ zdarzeń	1. Walidacja użytkownika. 2a. Przyznanie dostępu. 3. Zapytanie o lokalizacje 4. Pobranie lokalizacji 5. Lokalizacje zostają zwrócone 6. Załadowanie lokalizacji
Alternatywny przepływ zdarzeń	2b. Odmowa dostępu

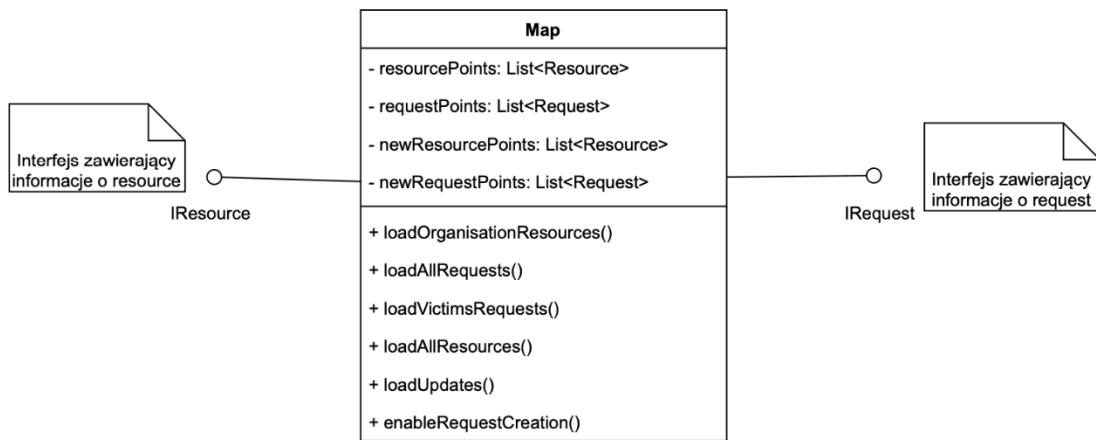
2.5.4.6 Diagram komunikacji – Wyświetlenie lokalizacji wszystkich zgłoszeń:

**Sekwencja działań**

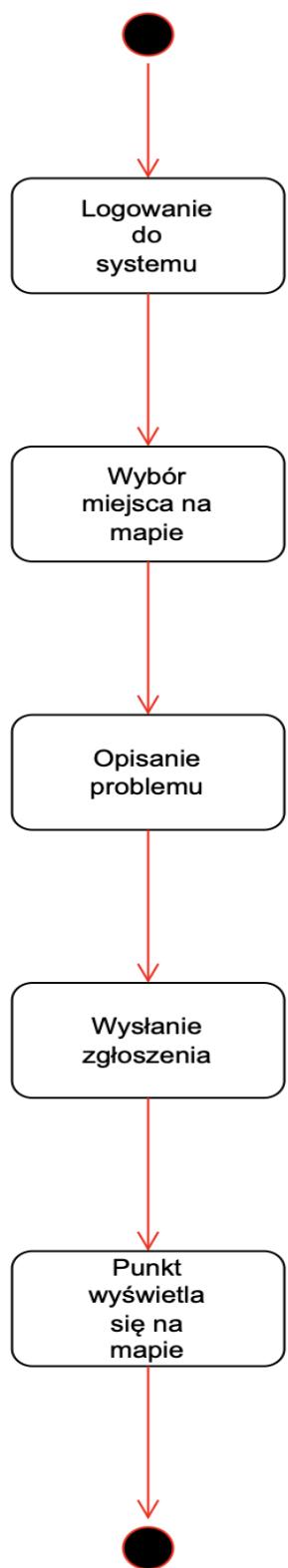
1. **Organizacja pomocowa** prosi **Auth** o walidację tokenu.
2. **Auth** zwraca informacje o sesji i roli użytkownika do **Organizacja pomocowa**.
3. **Organizacja pomocowa** wywołuje metodę `loadAllRequests()` w **MapUI**, aby załadować wszystkie zgłoszenia.
4. **MapUI** pobiera znaczniki zgłoszeń od **Request** wywołując metodę `pobierz znaczniki`.
5. **Request** zwraca znaczniki zgłoszeń do **MapUI**.
6. **MapUI** ładuje aktualizacje i dostarcza je do **Organizacja pomocowa**.

Część I: Dokumentacja techniczna

Diagram pokazuje proces tworzenia punktów pomocy w systemie, ilustrując interakcje pomiędzy użytkownikami a różnymi komponentami systemu.

2.5.4.7 Diagram klas:

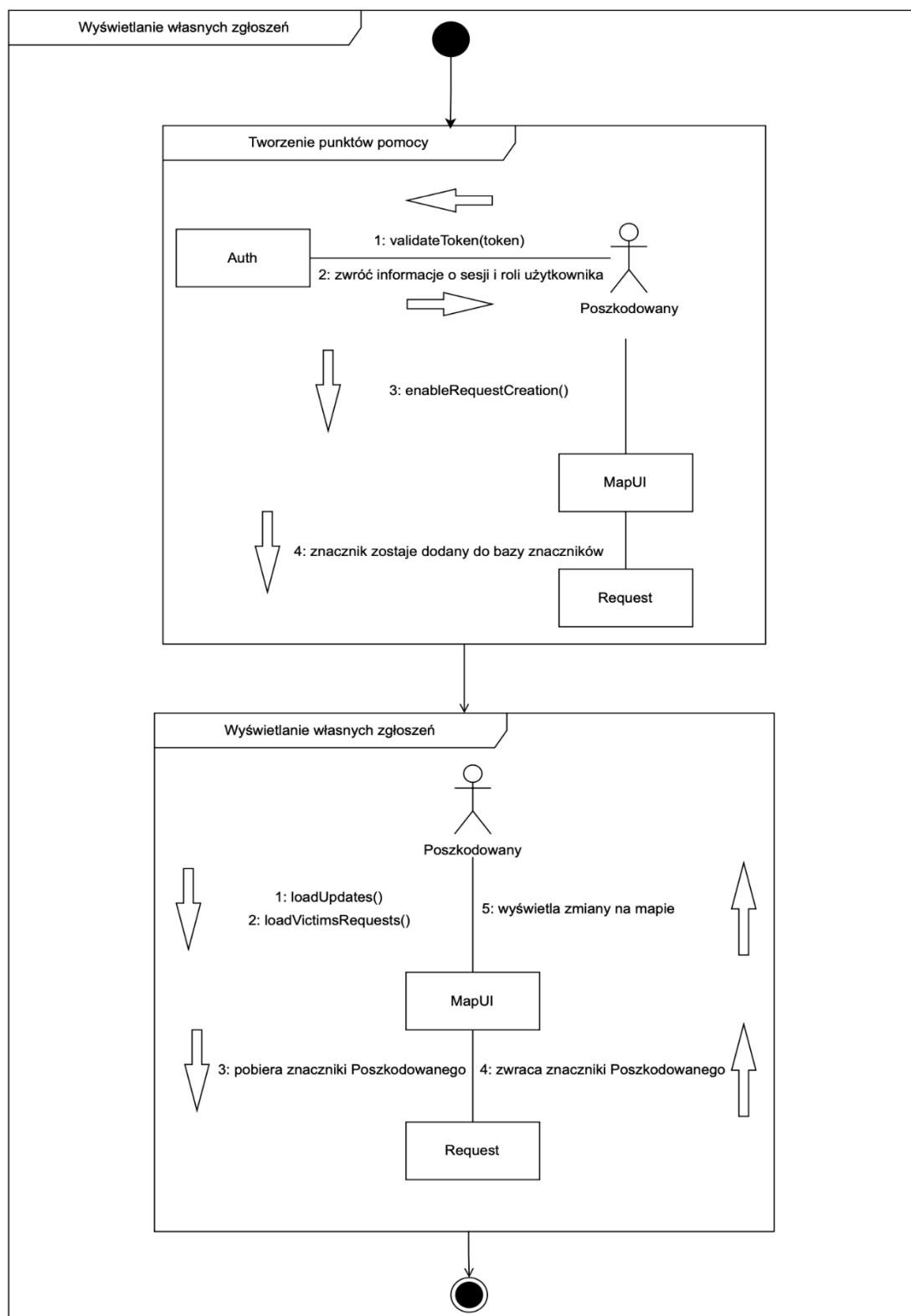
Część I: Dokumentacja techniczna

2.5.4.8 *Diagram czynności – Tworzenie zgłoszeń:*

Powyższy diagram pokazuje przepływ pracy w systemie. Są to kolejne czynności, które musi podjąć użytkownik by dodać zgłoszenie.

Część I: Dokumentacja techniczna

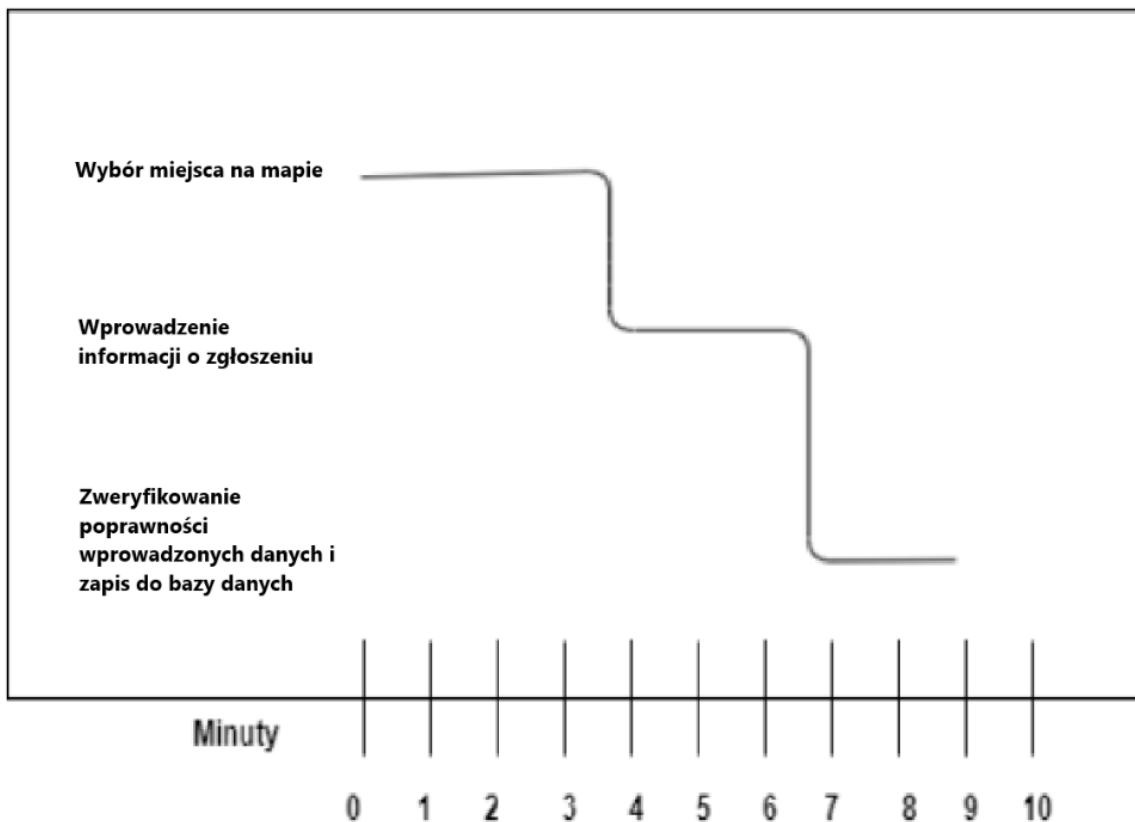
2.5.4.9 Diagram przeglądu interakcji:



Powyższy diagram łączy cechy diagramu czynności i diagramu sekwencji, aby pokazać interakcje między różnymi elementami systemu w kontekście wyświetlania własnych zgłoszeń.

Część I: Dokumentacja techniczna

2.5.4.10 Diagram harmonogramowania:



Powyższy diagram przedstawienia proces tworzenia nowego zgłoszenia podzielony jest na trzy etapy, które są przedstawione jako odcinki na wykresie krokowym, z osią X oznaczoną jako "Minuty".

2.5.4.11 Diagram strukturalny:

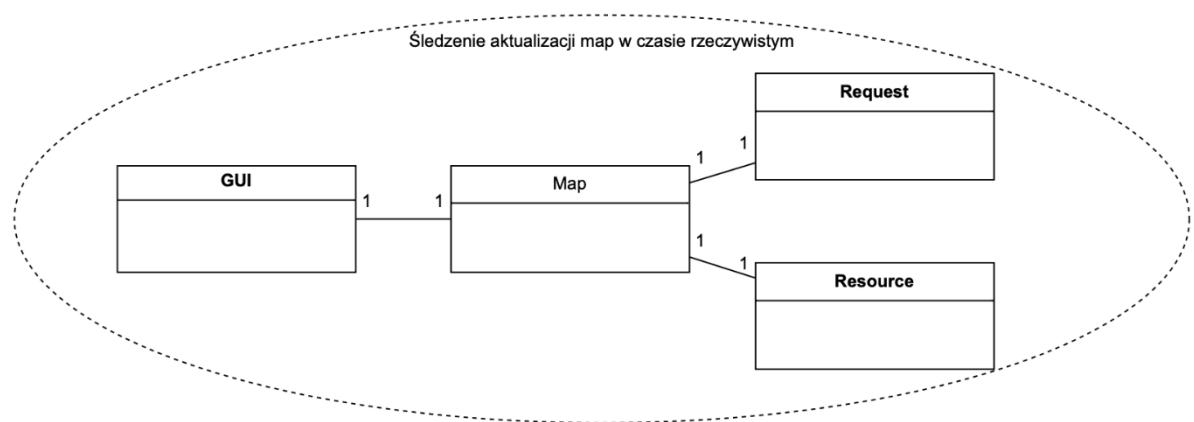


Diagram przedstawia strukturę systemu śledzenia aktualizacji map w czasie rzeczywistym.

2.6 Moduł Lang

2.6.1 Cel

System ma wspierać język polski i angielski.

2.6.2 Założenia

Możliwość trwałej zmiany języka dla całego systemu.

2.6.3 Wymagania

1. Funkcjonalne

- Możliwość zmiany języka za pomocą jednego przycisku
- Możliwość zmiany języka w każdym miejscu systemu

2. Niefunkcjonalne

- System powinien pamiętać wybrany język
- Zmiana nie powinna trwać dłużej niż 2 sekundy

2.6.4 Diagramy

2.6.4.1 Diagram przypadków użycia

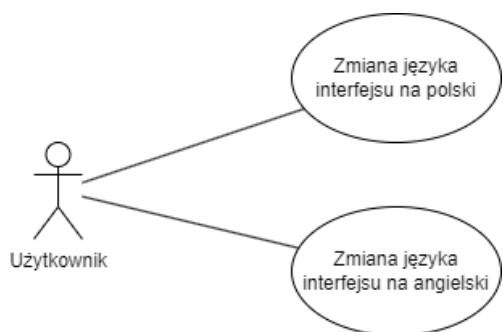


Diagram przypadków użycia przedstawia interakcje między użytkownikiem a modułem *Language* w kontekście realizowanych funkcji.

Każdy użytkownik systemu dzięki modułowi języków ma możliwość zmiany języka na polski oraz angielski.

2.6.4.2 Diagram komponentów

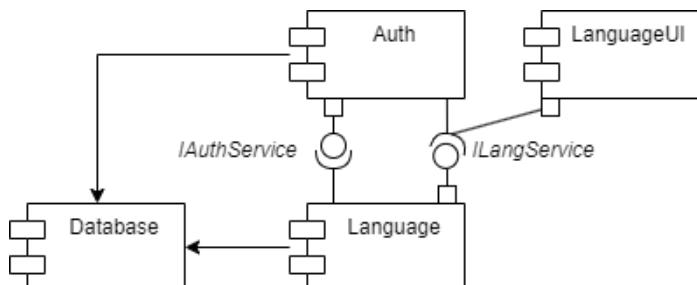


Diagram przedstawia strukturę komponentów w kontekście modułu *Language*. Moduł udostępnia interfejs *ILangService* w celu zapisania rekordu odnośnie języka dla nowego użytkownika oraz korzysta z *IAuthService* dzięki którym możliwe jest zapisanie preferowanego przez użytkownika języka do bazy danych.

2.6.4.3 Diagram pakietów

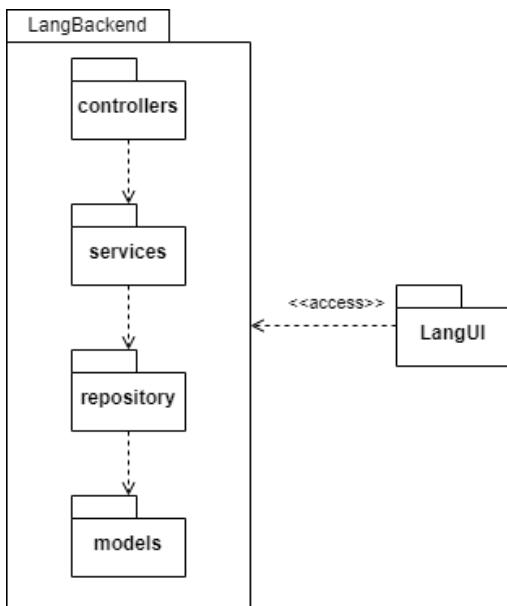


Diagram przedstawia strukturę pakietów wewnętrznych modułu języka *Lang*. Każdy pakiet grupuje klasy odpowiedzialne za określoną funkcjonalność systemu. Pakiet *models* jest odzwierciedleniem tabel w bazie danych, *repository* odpowiada za wysyłanie zapytań do bazy danych, *service* odpowiada za logikę a *controllers* odpowiada za udostępnianie endpointów. Pakiet *LangUI* jest reprezentacją graficzną dla użytkownika.

2.6.4.4 Diagram klas

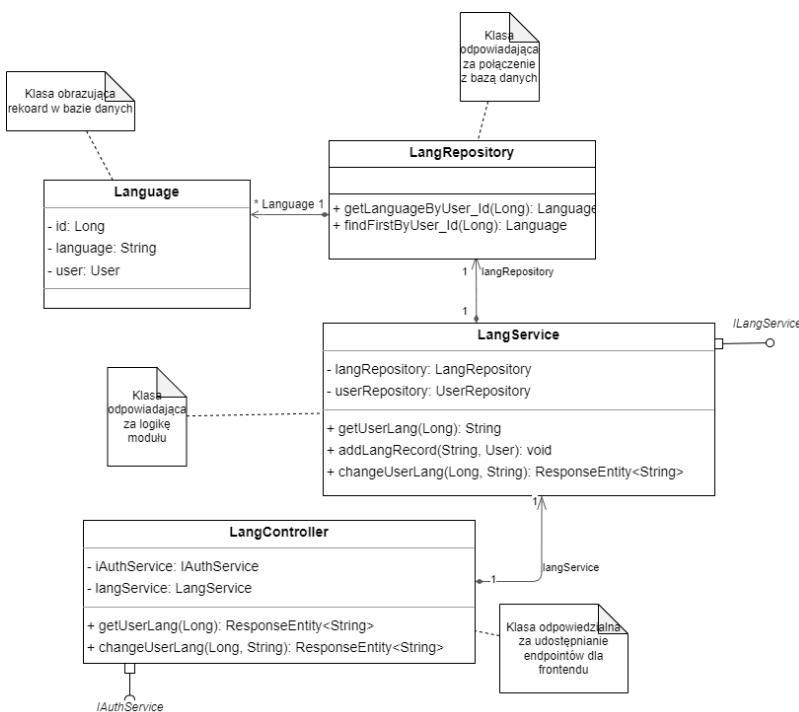
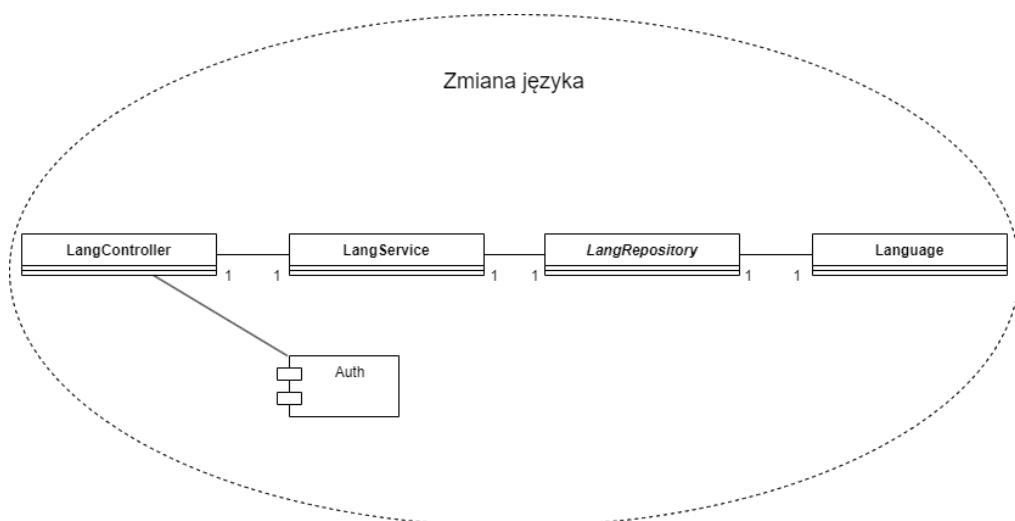


Diagram klas przedstawia strukturę modułu *Lang* w postaci klas, ich atrybutów, metod oraz relacji. Głównym elementem jest klasa *LangController*, która udostępnia endpointy oraz korzysta z interfejsu *IAuthService*. *LangService* odpowiada za całą logikę i zmianę języka oraz udostępnia interfejs *ILangService*, *LangRepository* odpowiada za komunikację z bazą danych, a *Language* jest odwzorowaniem rekordu z bazy danych.

Część I: Dokumentacja techniczna

2.6.4.5 Diagram strukturalny (struktur połączonych)



Po przejściu na endpoint odpowiedzialny za zmianę języka, wybrany przez użytkownika język jest zapisywany do bazy danych. Dzięki temu, gdy użytkownik zaloguje się na swoje konto z innego systemu, jego wybór zostanie zapamiętany.

2.6.4.6 Diagram sekwencji - zmiana języka interfejsu na polski

Scenariusz przypadku użycia - zmiana języka interfejsu na polski

Nazwa	Zmiana języka interfejsu na polski
Poziom ważności	Średni
Typ przypadku użycia	Ogólny
Aktorzy	Użytkownik
Krótki opis	Zmiana języka na polski oraz zapisanie tego w bazie danych
Warunki wstępne	Brak
Warunki końcowe	Zmiana języka
Główny przepływ zdarzeń	1. Użytkownik zmienia język poprzez naciśnięcie na flagę polski 2. Język systemu zmienia się na polski, a informacja o tym zostaje zapisana w bazie danych
Alternatywne przepływy zdarzeń	2a. Gdy użytkownik nie jest zalogowany, zmienia się język aplikacji natomiast zmiana nie jest zapisana w bazie danych
Specjalne wymagania	1. System nie powinien przetwarzać żądania dłużej niż 2 sekundy

Część I: Dokumentacja techniczna

2.6.4.7

Diagram sekwencji - zmiana języka interfejsu na polski

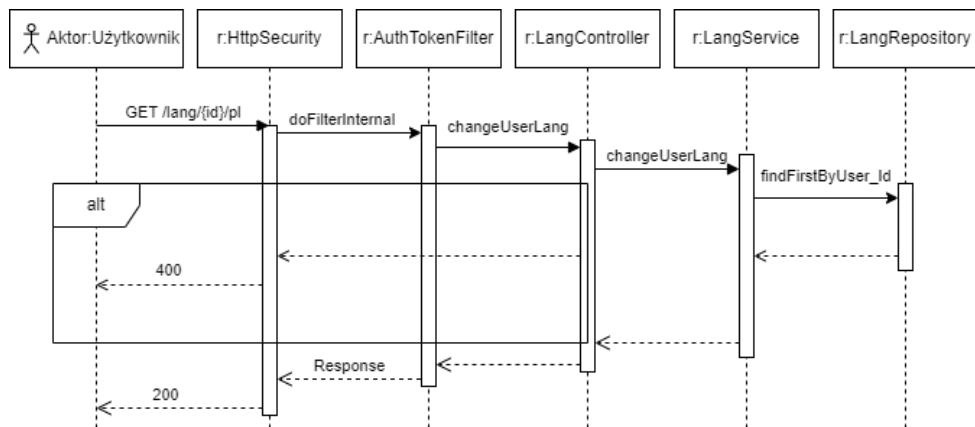


Diagram przedstawia interakcję pomiędzy obiektami *LangController*, *LangService* i *LangRepository* (oraz *HttpSecurity* oraz *AuthTokenFilter* z modułu *Auth*). Obiekty wspólnie realizują proces zmiany języka interfejsu na polski.

Obiekt *LangRepository* odpowiada za komunikację z bazą danych, *LangController* za udostępnienie endpointu a *LangService* odpowiada za logikę całego procesu.

2.6.4.8 Diagram sekwencji - zmiana języka interfejsu na angielski

Scenariusz przypadku użycia - zmiana języka interfejsu na angielski

Nazwa	Zmiana języka interfejsu na angielski
Poziom ważności	Średni
Typ przypadku użycia	Ogólny
Aktorzy	Użytkownik
Krótki opis	Zmiana języka na angielski oraz zapisanie tego w bazie danych
Warunki wstępne	Brak
Warunki końcowe	Zmiana języka
Główny przepływ zdarzeń	1. Użytkownik zmienia język poprzez naciśnięcie na flagę wielkiej Brytanii 2. Język systemu zmienia się na angielski, a informacja o tym zostaje zapisana w bazie danych
Alternatywne przepływy zdarzeń	2a. Gdy użytkownik nie jest zalogowany, zmienia się język aplikacji natomiast zmiana nie jest zapisana w bazie danych
Specjalne wymagania	1. System nie powinien przetwarzać żądania dłużej niż 2 sekundy

2.6.4.9 Diagram sekwencji - zmiana języka interfejsu na angielski

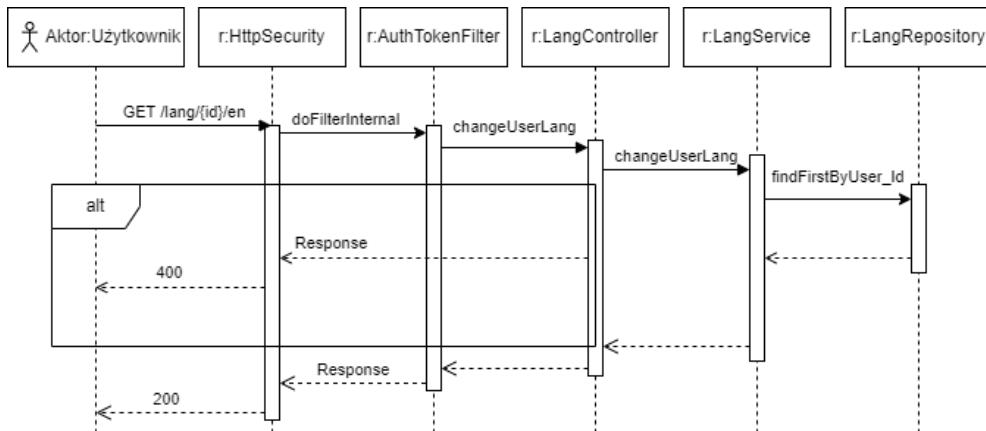


Diagram przedstawia interakcję pomiędzy obiektami *LangController*, *LangService* i *LangRepository* (oraz *HttpSecurity* oraz *AuthTokenFilter* z modułu *Auth*). Obiekty wspólnie realizują proces zmiany języka interfejsu na angielski.

Obiekt *LangRepository* odpowiada za komunikację z bazą danych, *LangController* za udostępnienie endpointu a *LangService* odpowiada za logikę całego procesu.

2.6.4.10 Diagram przeglądu interakcji

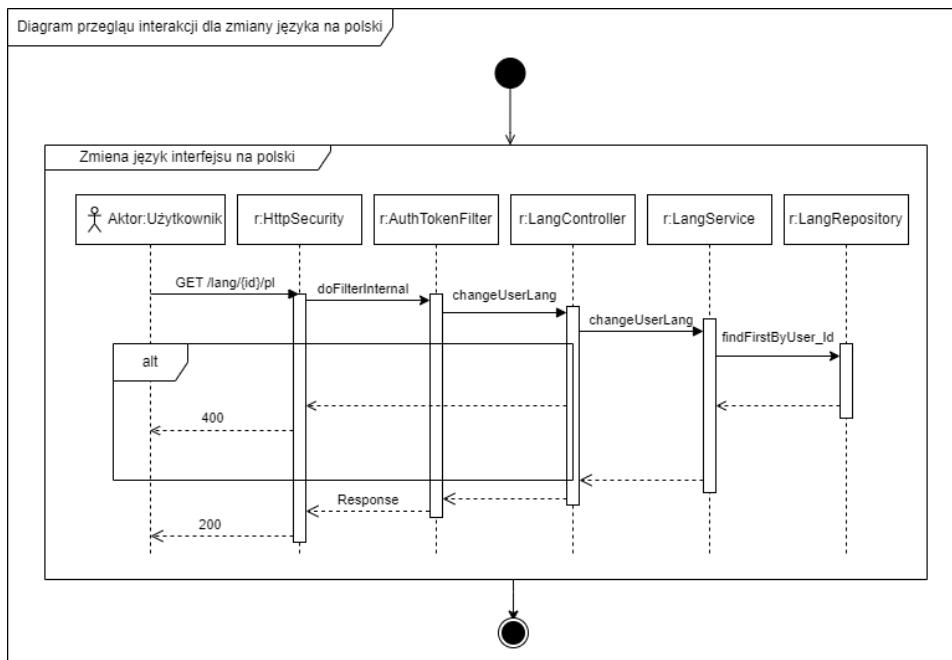


Diagram przedstawia przepływ sterowania w procesie zmiany języka interfejsu na polski. Ponieważ zmiana języka jest prostą funkcjonalnością, diagram przedstawia jeden diagram sekwencji prezentujący zmianę języka interfejsu na polski.

2.6.4.11 Diagram czynności

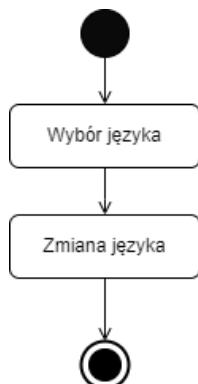


Diagram czynności przedstawia proces zmiany języka. Cały proces obejmuje wybranie przez użytkownika żądanego języka oraz zmianę języka w obrębie całego systemu. W przypadku, gdy użytkownik jest zalogowany zmiana zapisywana jest również w bazie danych.

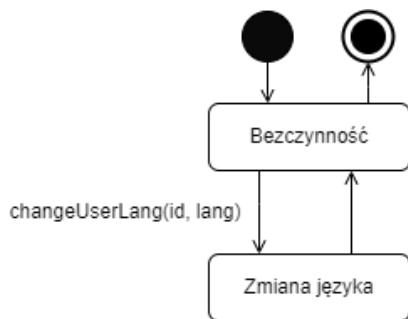
2.6.4.12 Diagram maszyny stanowej (*klasa LangController*)

Diagram przedstawia zmianę stanu obiektu *LangControl*. Obiekt domyślnie jest w stanie bezczynności. Gdy użytkownik zmieni język, wykonywana jest metoda *changeUserLang*, po której uruchomieniu stan obiektu zmienia się na „Zmiana języka”. Po wykonaniu całego procesu, obiekt wraca do stanu bezczynności.

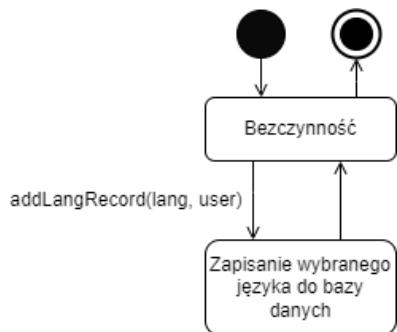
2.6.4.13 Diagram maszyny stanowej (*klasa LangService*)

Diagram przedstawia zmianę stanu obiektu *LangService*. Obiekt domyślnie jest w stanie bezczynności. Gdy zalogowany użytkownik zmieni język, wykonywana jest metoda *addLangRecord*, a obiekt zmienia stan na „Zapisanie wybranego języka do bazy danych”. Po zakończeniu operacji związanych z zapisywaniem do bazy danych obiekt wraca do stanu bezczynności.

2.6.4.14 Diagram harmonogramowania

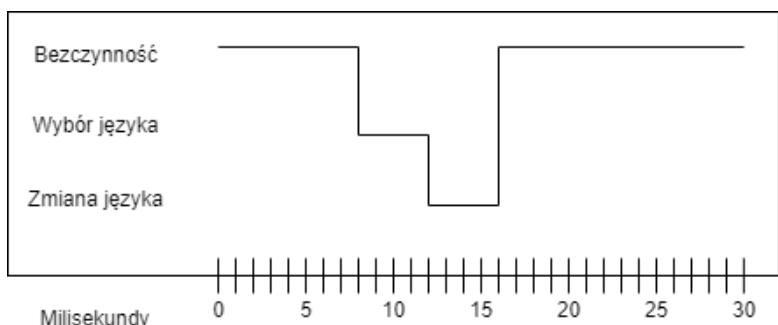


Diagram przedstawia zmiany w systemie w czasie podczas zmiany języka. Po wybraniu języka dane przesyłane są do backendowego modułu *Lang*, który to zapisuje zmianę w bazie danych, a następnie wysyła odpowiedź.

2.7 Moduł Report

2.7.1 Cel

Celem jest stworzenie modułu umożliwiającego generowanie raportów dotyczących Systemu Koordynacji Pomocy Humanitarnej (SKPH).

2.7.2 Założenia

Moduł umożliwia wygenerowanie raportów oraz ich podgląd przez uprawnionych użytkowników.

Moduł musi współpracować z innymi interfejsami w celu zapewnienia wygenerowania raportów z aktualnymi danymi oraz kontrolowanym dostępem do nich.

2.7.3 Wymagania

1. Wymagania funkcjonalne:

- Pobranie raportu w formie PDF o działaniach pomocowych
- Pobranie raportu w formie PDF o stanie zasobów
- Pobranie raportu w formie PDF o zgłoszeniach
- Pobranie raportu w formie PDF potwierdzenia darowizny
- Pobranie raportu w formie PDF o uldze podatkowej
- Podgląd raportu o działaniach pomocowych
- Podgląd raportu o stanie zasobów
- Podgląd raportu o zgłoszeniach
- Podgląd raportu potwierdzenia darowizny
- Podgląd raportu o uldze podatkowej

2. Wymagania niefunkcjonalne:

- Wydajność – system powinien reagować w czasie nie dłuższym niż 2 sekundy.
- Bezpieczeństwo – system powinien udostępniać możliwość generowania oraz podglądu raportów tylko odpowiednim użytkownikom
- Użyteczność – raporty powinny być czytelne
- Trwałość – dane są zapisywane w bazie danych

2.7.4 Diagramy

2.7.4.1 Diagram przypadków użycia dla Organizacji pomocowej

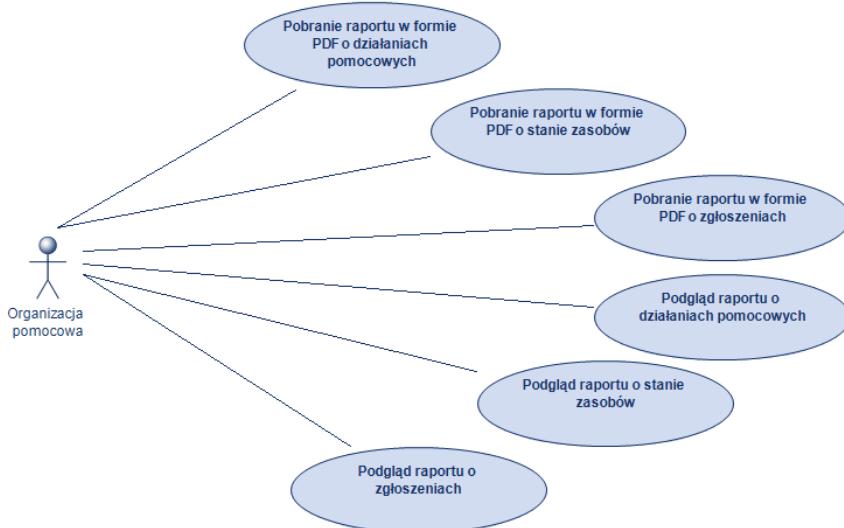


Diagram przypadków użycia organizacji pomocowej, który przedstawia możliwe działania zalogowanego użytkownika który posiada rolę „Organizacja pomocowa”. Do dyspozycji ma 3 typy raportu: działania pomocowe, stan zasobów oraz zgłoszenia. Wszystkie raporty może uzyskać w dwóch wersjach: pliku PDF i Wyświetlone na stronie pod wyborem raportów.

2.7.4.2 Diagram przypadków użycia dla Przedstawiciela władz

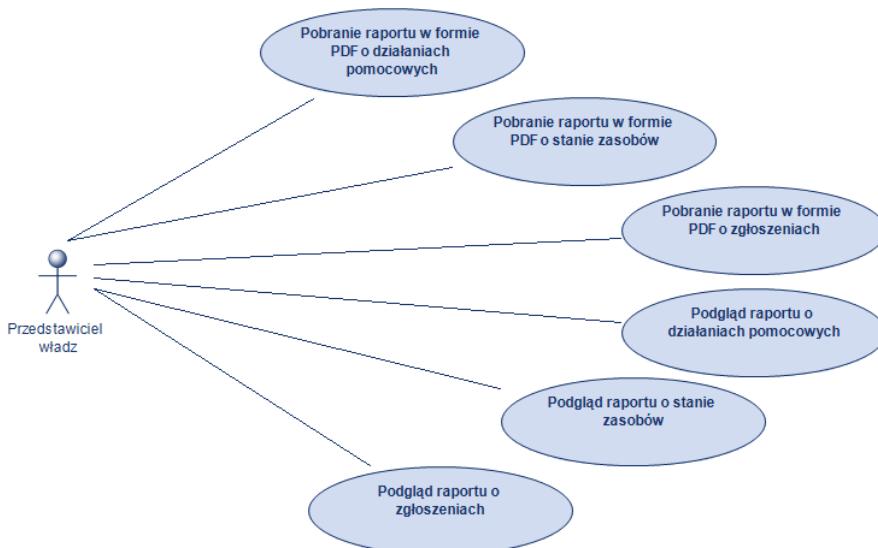


Diagram przypadków użycia Przedstawiciela władz, który przedstawia możliwe działania zalogowanego użytkownika który posiada rolę „Przedstawiciel władz”. Do dyspozycji ma 3 typy raportu: działania pomocowe, stan zasobów oraz zgłoszenia. Wszystkie raporty może uzyskać w dwóch wersjach: pliku PDF i Wyświetlone na stronie pod wyborem raportów.

Część I: Dokumentacja techniczna

2.7.4.3 Diagram przypadków użycia dla Darczyńcy

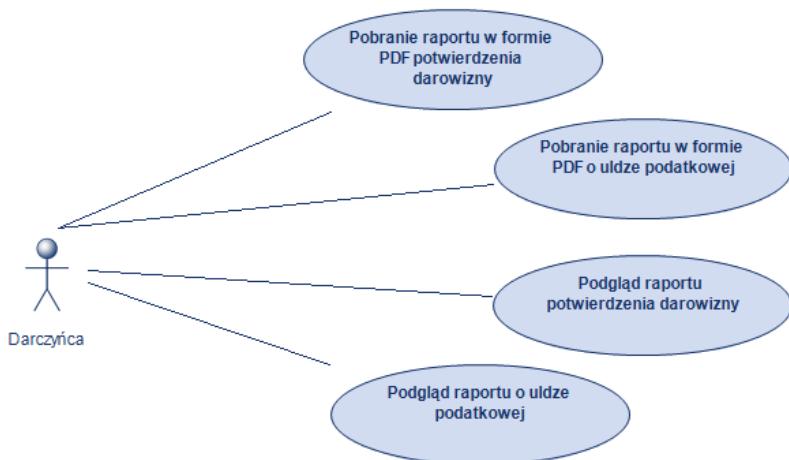


Diagram przypadków użycia Darczyńcy, który przedstawia możliwe działania załogowanego użytkownika który posiada rolę „Darczynca”. Do dyspozycji ma 2 typy raportu: potwierdzenie darowizny oraz ulga podatkowa. Wszystkie raporty może uzyskać w dwóch wersjach: pliku PDF

i Wyświetlone na stronie pod wyborem raportów.

2.7.4.4 Diagram klas

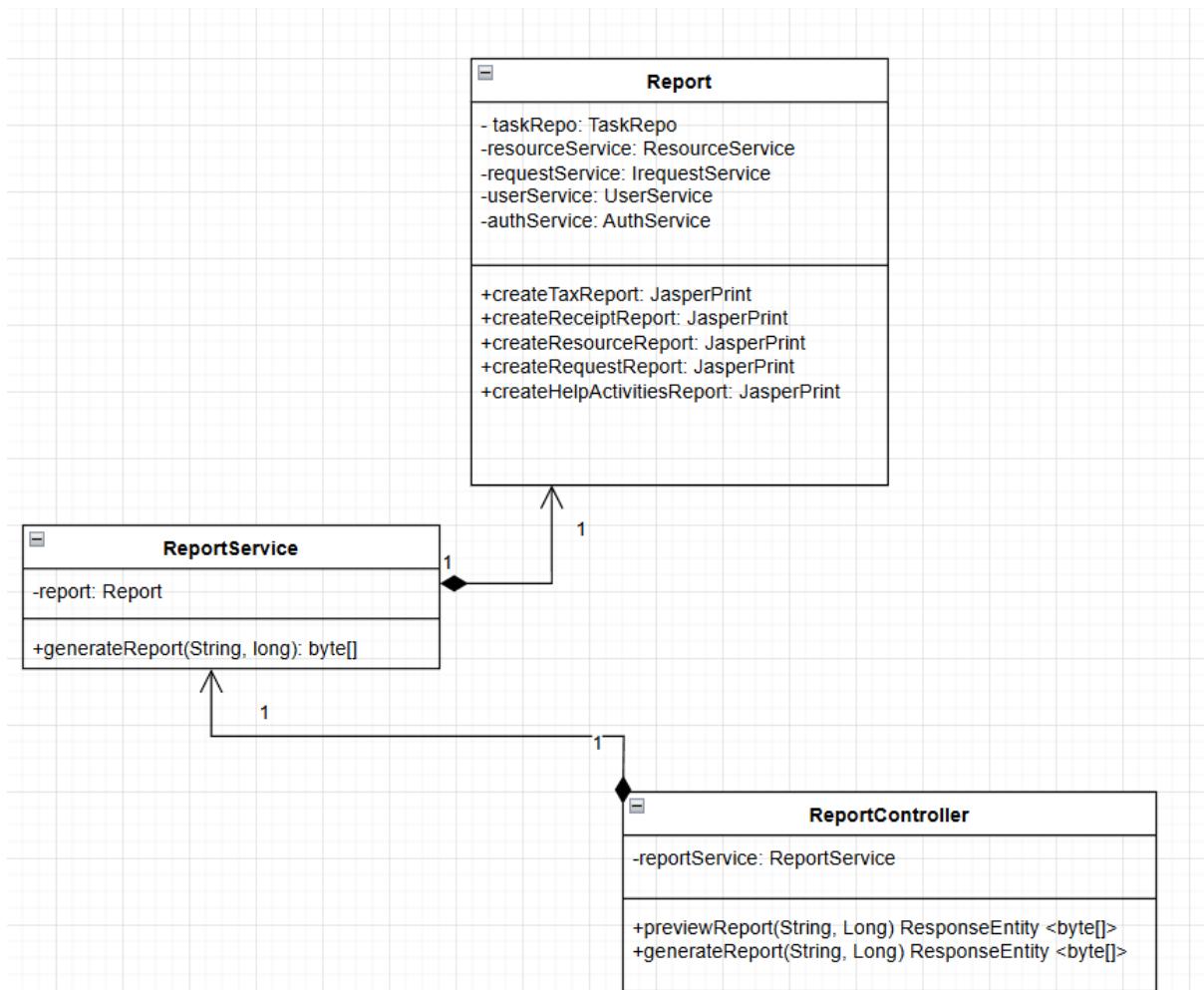


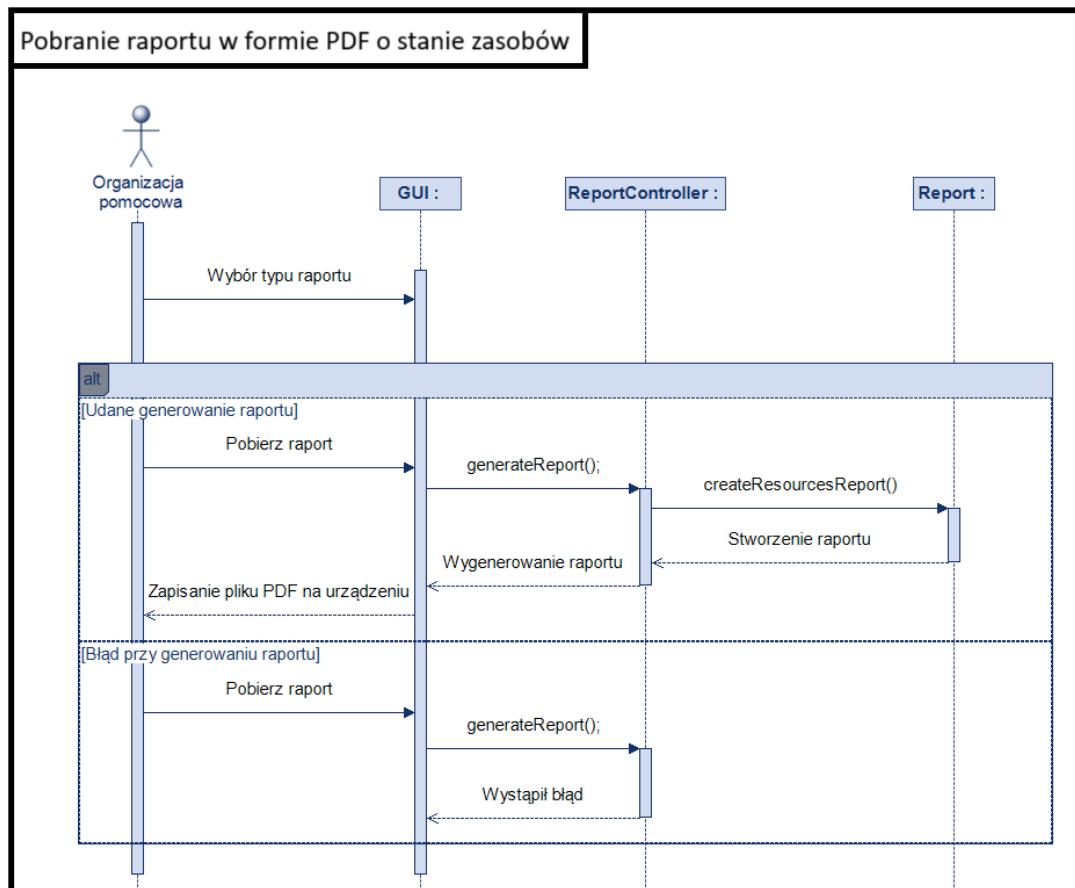
Diagram klas przedstawia klasy z ich atrybutami oraz metodami oraz relacje między nimi.

Część I: Dokumentacja techniczna

2.7.4.5 Scenariusz – Pobranie raportu w formie PDF o stanie zasobów

Nazwa:	Pobranie raportu w formie PDF o stanie zasobów
Poziom ważności:	Średni
Aktorzy:	Organizacja pomocowa
Krótki opis:	Organizacja pomocowa generuje raport o stanie zasobów i zapisuje go w wybranym miejscu na swoim urządzeniu
Warunki wstępne:	W systemie musi być zalogowany użytkownik jako Organizacja pomocowa oraz w bazie danych musi znajdować się zasób
Warunki końcowe:	System generuje raport na temat stanu zasobów oraz pobiera go w formacie PDF
Główny przepływ zdarzeń:	<ol style="list-style-type: none"> 1. Wybór raportu o stanie zasobów 2. Zatwierdzenie generowania raportu 3. Zapisanie raportu w wybranym miejscu na urządzeniu
Alternatywny przepływ zdarzeń:	2a. Wystąpienie błędu przy generowaniu raportu

2.7.4.6 Diagram sekwencji – pobranie raportu w formie PDF o stanie zasobów



Część I: Dokumentacja techniczna

Diagram sekwencji dotyczący generowania raportu. przedstawia realizację scenariusza generowania raportu w formie PDF o stanie zasobów, opisując interakcję między obiektami w postaci sekwencji komunikatów wymienianych między nimi. Użytkownik wybiera rodzaj raportu oraz jedną z dwóch opcji (Pobranie raportu/Podgląd raportu) w tym wypadku „Pobranie raportu”. Żądanie z odpowiednimi parametrami zostają przekazane do obiektu :ReportController, wykorzystujący obiekt :Report , który komunikuj się z bazą danych w celu pobrania potrzebnych danych do wygenerowania oczekiwanej raportu. Następnie do użytkownika zostaje przekazany komunikat o powodzeniu operacji i wybrany wcześniej rodzaj raportu w odpowiedniej formie (PDF). Przebieg alternatywny przedstawia sytuację, w której wystąpił błąd wygenerowania raportu, na przykład z powodu braku wymaganych danych.

2.7.4.7 Scenariusz – Podglqd raportu potwierdzenia darowizny

Nazwa:	Podgląd raportu potwierdzenia darowizny
Poziom ważności:	Średni
Aktorzy:	Darczyńca
Krótki opis:	Darczyńca generuje potwierdzenie dotyczące przekazanych przez niego darowizn oraz ma możliwość jego podglądu na stronie
Warunki wstępne:	W systemie musi być zalogowany użytkownik jako Darczyńca w bazie danych musi znajdować się darowizna przekazana przez niego
Warunki końcowe:	System generuje potwierdzenie dotyczące przekazanych darowizn oraz umożliwia jego podgląd
Główny przepływ zdarzeń:	<ol style="list-style-type: none"> 1. Wybór typu raportu 2. Zatwierdzenie generowania raportu potwierdzenia darowizny 3. Podgląd raportu potwierdzenia darowizny
Alternatywny przepływ zdarzeń:	3a. Wystąpienie błędu przy generowaniu potwierdzenia

2.7.4.8 Diagram sekwencji – podgląd raportu potwierdzenia darowizny

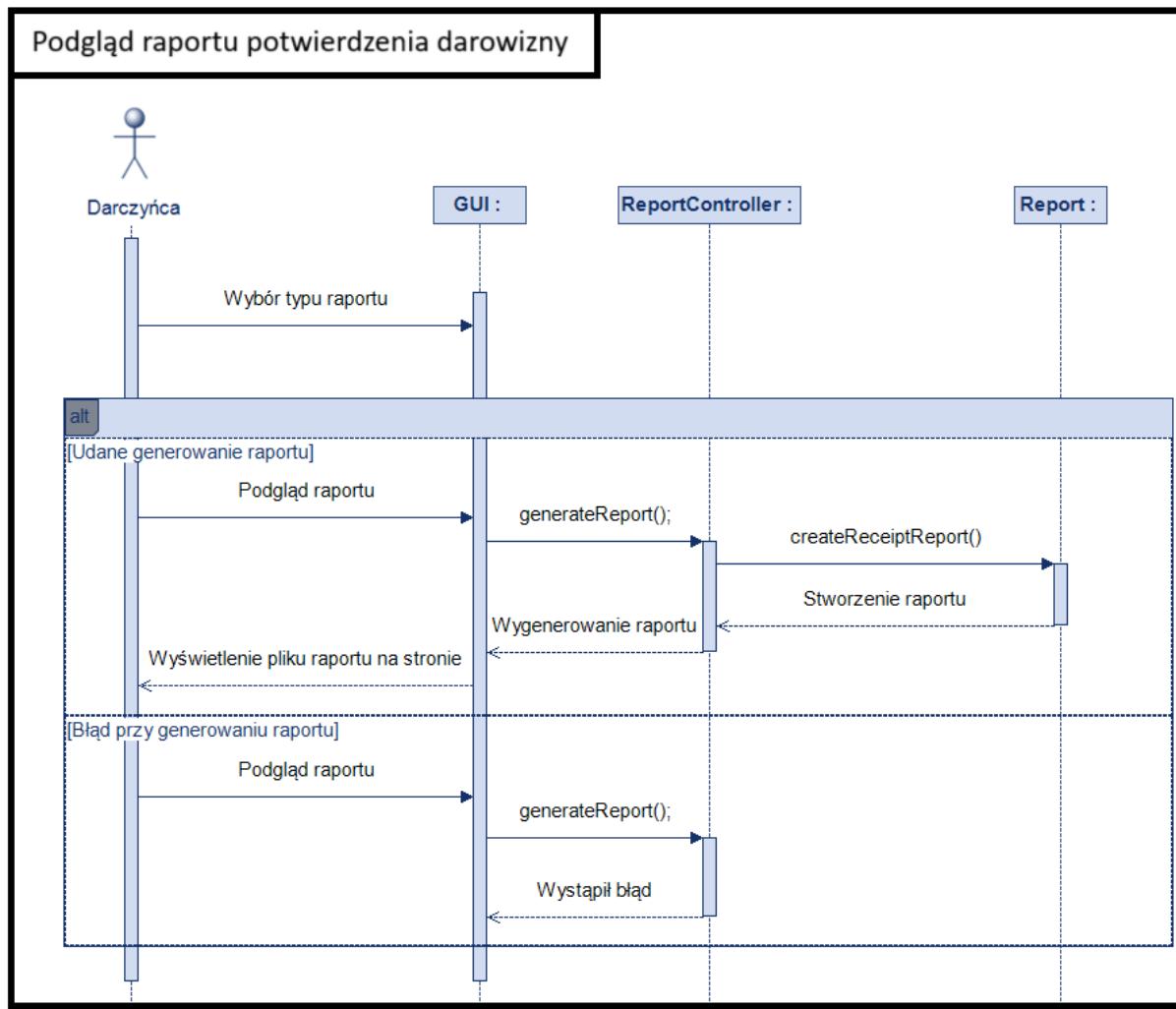
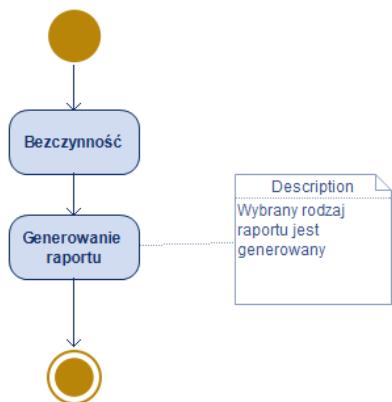


Diagram sekwencji dotyczący generowania raportu. przedstawia realizację scenariusza generowania raportu w formie podglądu na stronie o potwierdzeniu darowizny, opisując interakcję między obiektami w postaci sekwencji komunikatów wymienianych między nimi. Użytkownik wybiera rodzaj raportu oraz jedną z dwóch opcji (Pobranie raportu/Podgląd raportu) w tym wypadku „Podgląd raportu”. Żądanie z odpowiednimi parametrami zostają przekazane do obiektu :ReportController, wykorzystujący obiekt :Report , który komunikuje się z bazą danych w celu pobrania potrzebnych danych do wygenerowania oczekiwanej raportu. Następnie do użytkownika zostaje przekazany komunikat o powodzeniu operacji i wybrany wcześniej rodzaj raportu w odpowiedniej formie (Podgląd na stronie). Przebieg alternatywny przedstawia sytuację, w której wystąpił błąd wygenerowania raportu, powodem może być na przykład brak darowizn przekazanych przez tego użytkownika.

2.7.4.9 *Diagram maszyny stanowej*

Description
Wybrany rodzaj raportu jest generowany

Diagram maszyny stanowej przedstawia串行流程, jakie przyjmuje obiekt w odpowiedzi na zdarzenia zachodzące w czasie jego życia oraz reakcje obiektu na te zdarzenia. Obiekt zajmujący się generowaniem raportu na początku znajduje się w stanie Bezczynności. Po wybraniu typu raportu i formie w jakiej chcę się go otrzymać obiekt przechodzi w stan Generowania raportu.

2.7.4.10 *Diagram czynności - podgląd rapportu o stanie zasobów*

Część I: Dokumentacja techniczna

Diagram czynności przedstawia czynności wykonywane podczas generowania raportu. Użytkownik najpierw musi zalogować się do systemu na konto z odpowiednią rolą (Organizacja pomocowa/Przedstawiciel władz), następnie przejść w zakładkę „Reports”. Użytkownik wybiera odpowiedni typ raportu oraz decyduje w jakiej formie chce go otrzymać (Pobrany plik PDF/Podgląd raportu na stronie). Wynikiem poprawnego działania będzie komunikat o udanym generowaniu i raport w wybranym formacie.

2.7.4.11 Diagram komponentów

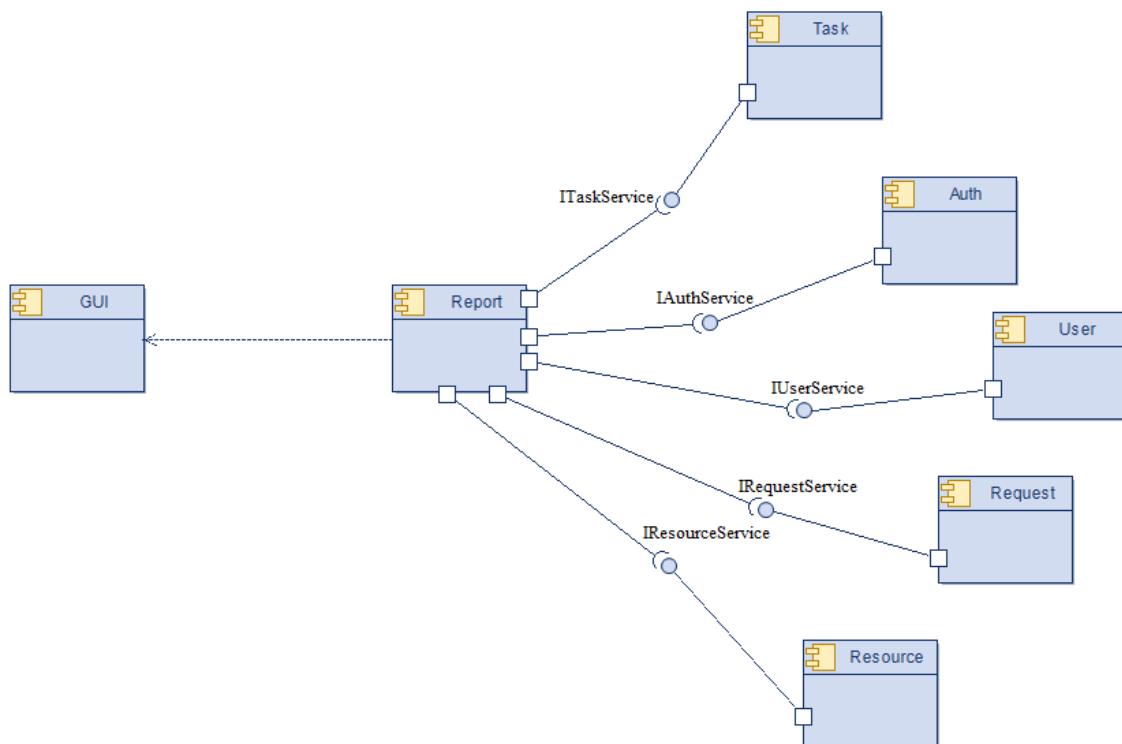
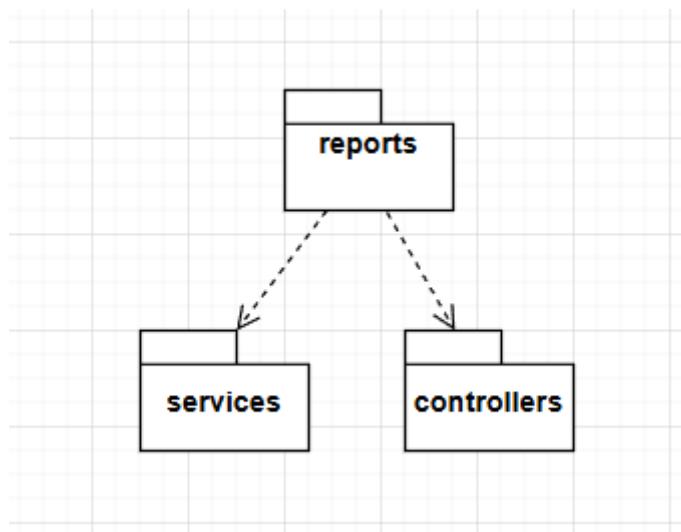


Diagram przedstawia strukturę komponentów w kontekście modułu raportowania (Report). Moduł Report wykorzystuje dane zapewnione przez moduły *Resource*, *Request*, *User* oraz *Task*. Autoryzację potrzebną przy odpowiednim filtrowaniu dostępnych czynności dla użytkowników udostępnia mu moduł *Auth*.

2.7.4.12 Diagram pakietów



Część I: Dokumentacja techniczna

Diagram przedstawia strukturę pakietów wewnętrz modułu raportowania Report. Każdy pakiet grupuje klasy odpowiedzialne za konkretne funkcjonalność systemu. Controllers zawiera endpointy, services metody pośredniczące pomiędzy endpointem a klasą Report.

2.7.4.13 Diagram przeglądu interakcji

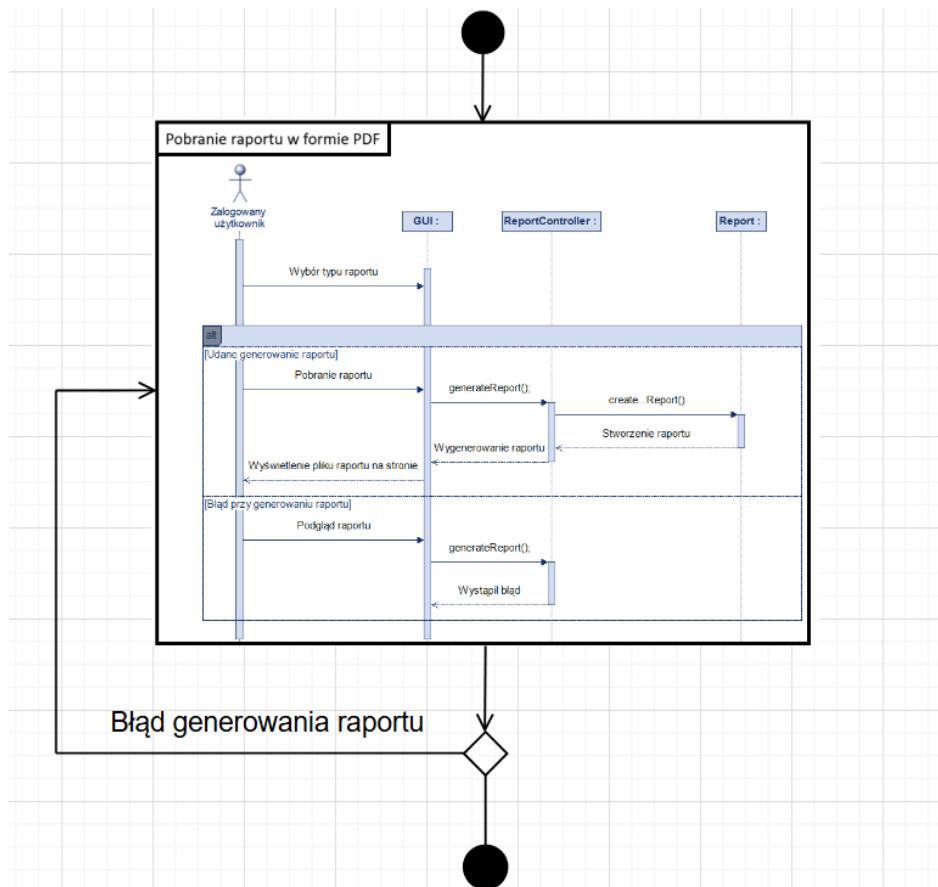
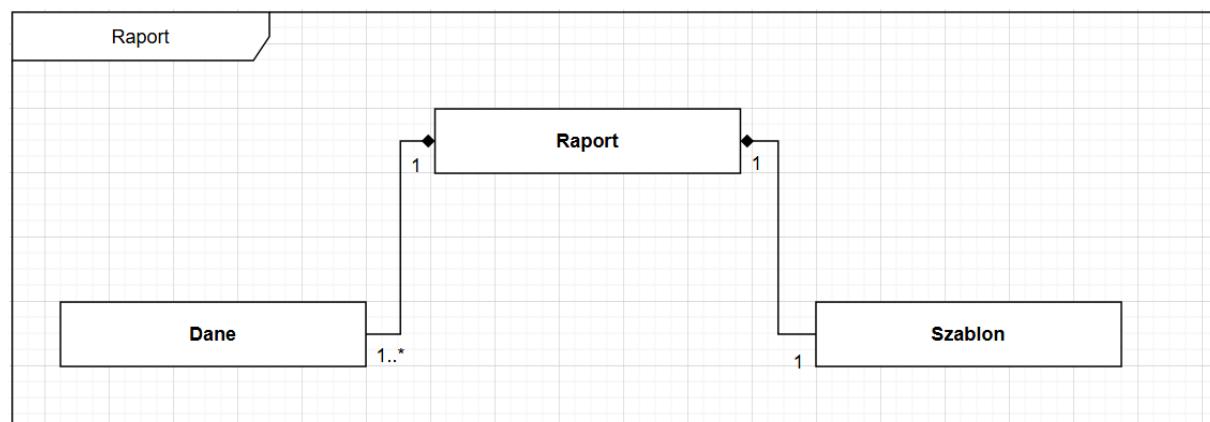


Diagram przeglądu interakcji przedstawia diagram interakcji oraz możliwy błąd. W momencie, gdy zalogowany użytkownik wybiera typ i sposób przekazania (PDF/wyświetlenie na stronie) raportu, następuje wymiana komunikatów między obiektami w celu udostępnienia odpowiednich metod. Jeżeli parametry są poprawne i błąd nie występuje, następuje wymiana komunikatów między obiektem ReportController: i Report: a bazą danych w celu pobrania danych potrzebnych do wygenerowania raportu.

2.7.4.14 Diagram strukturalny



Część I: Dokumentacja techniczna

Diagram strukturalny przedstawia strukturę raportu, czyli potrzebne części, które są niezbędne aby raport miał wizualny sens. W tym wypadku jest to przypisany szablon oraz dane które określają jego zawartość.

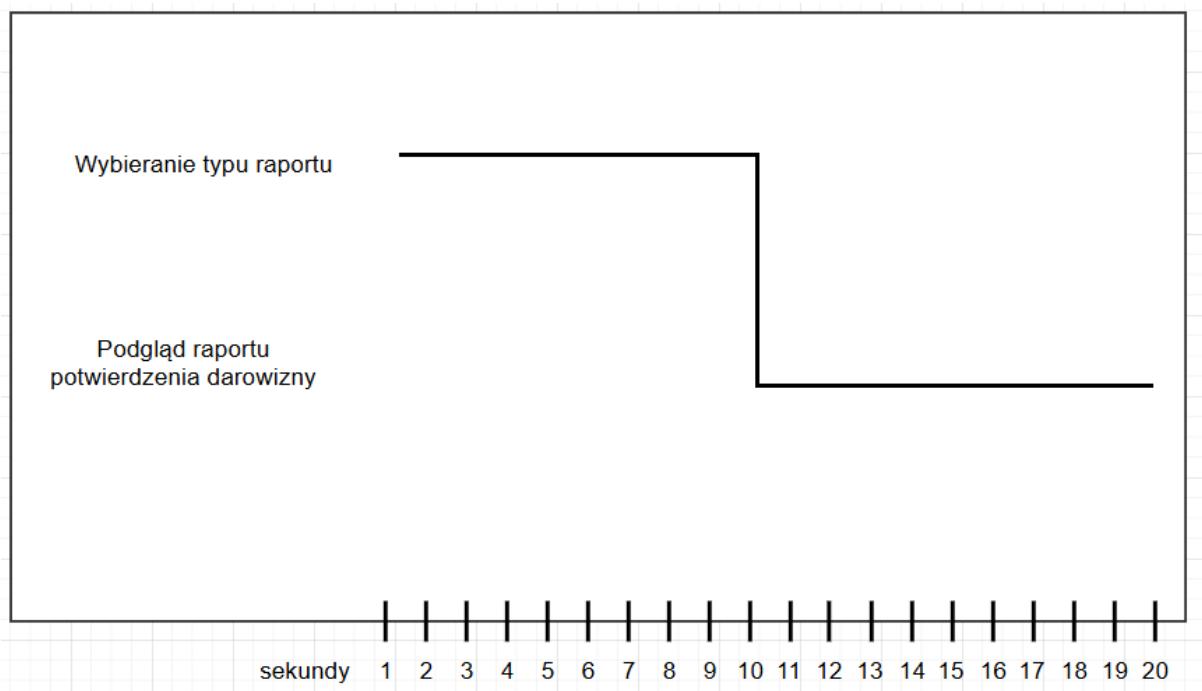
2.7.4.15 Diagram harmonogramowania – Podgląd rapportu darowizny

Diagram harmonogramowania reprezentuje na osi czasu zmiany dopuszczalnych stanów jakie może przyjmować obiekt lub aktor uczestniczący w procesie. W naszym przypadku określamy czas podglądu raportu przez aktora. Sam proces generowania i wyświetlania raportu jest znikomy, dlatego najdłużej i zajmuje wybór i podgląd samego raportu.

2.8 Moduł Request

2.8.1 Cel

Celem modułu jest rejestracja zgłoszeń poszkodowanych i zarządzanie nimi.

2.8.2 Założenia

1. Założenia realizowane przez moduł:

- Umożliwienie osobom potrzebującym rejestracji zgłoszeń
- Przechowywanie informacji o potrzebach zgłoszonych przez poszkodowanych
- Umożliwienie poszkodowanym aktualizacji wcześniej zgłoszonych potrzeb, o ile nie zostało ono przyjęte do realizacji albo zakończone
- Umożliwienie usunięcia zgłoszenia przez poszkodowanego oraz przedstawicieli władz
- Umożliwienie poszkodowanym wyświetlenie listy ich zgłoszeń oraz śledzenie statusu zgłoszeń
- Umożliwienie przedstawicielom władz wyświetlenie listy wszystkich zgłoszeń

2.8.3 Wymagania

1. Wymagania funkcjonalne

- Umożliwienie osobom potrzebującym rejestracji zgłoszeń
 - Każde zgłoszenie będzie zawierało informacje o potrzebnych zasobach oraz ich ilości, lokalizację oraz opcjonalnie dodatkowe informacje
- Przechowywanie informacji o potrzebach zgłoszonych przez poszkodowanych
 - Każde zgłoszenie będzie zapisywane do bazy danych
- Umożliwienie aktualizacji wcześniej zgłoszonych potrzeb
 - Aktualizacja wcześniej zgłoszonej potrzeby może być wykonana tylko, jeśli dane zgłoszenie nie zostało przyjęte do realizacji ani zakończone
 - Aktualizować zgłoszenie może tylko osoba, która jest jego autorem
- Umożliwienie usunięcia zgłoszenia
 - Poszkodowany może usunąć tylko swoje zgłoszenia
 - Przedstawiciel władz może usuwać zgłoszenia poszkodowanych
 - Mogą być usunięte tylko zgłoszenia, które nie zostały oznaczone jako wykonane
- Umożliwienie poszkodowanym wyświetlenie listy ich zgłoszeń oraz śledzenia statusu zgłoszenia
 - Poszkodowany na liście widzi tylko swoje zgłoszenia
- Umożliwienie przedstawicielom władz wyświetlania listy wszystkich zgłoszeń
 - Przedstawiciel władz na liście widzi wszystkie zgłoszenia

2. Wymagania niefunkcjonalne

- Wydajność
 - Reakcja na akcje użytkownika w czasie nie dłuższym niż 2 sekundy
- Użyteczność

Część I: Dokumentacja techniczna

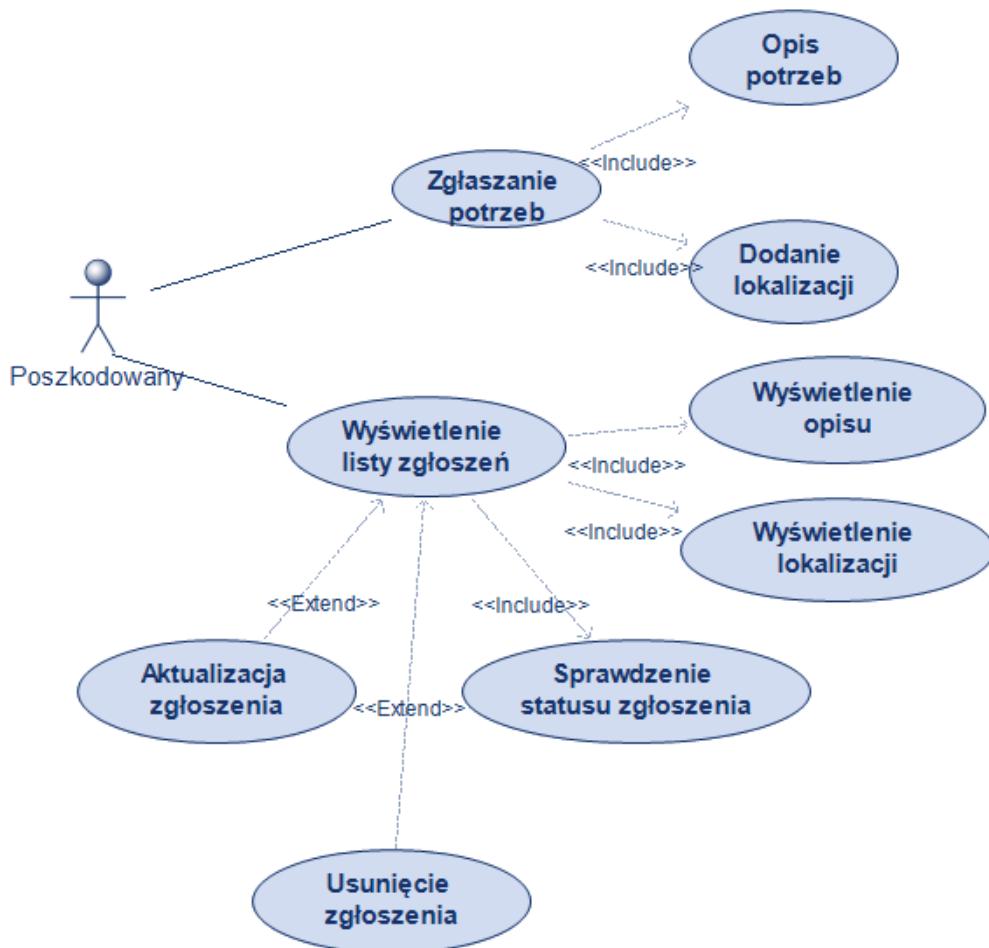
- Intuicyjny i przyjazny interfejs użytkownika
- Niezawodność
- Dostępność przez 99% czasu
- Kompatybilność
- wsparcie dla najnowszych wersji przeglądarek internetowych oraz dostęp na urządzeniach mobilnych (przez przeglądarkę)

2.8.4 Diagramy

W naszym module jest dwóch aktorów – poszkodowany oraz przedstawiciel władz.

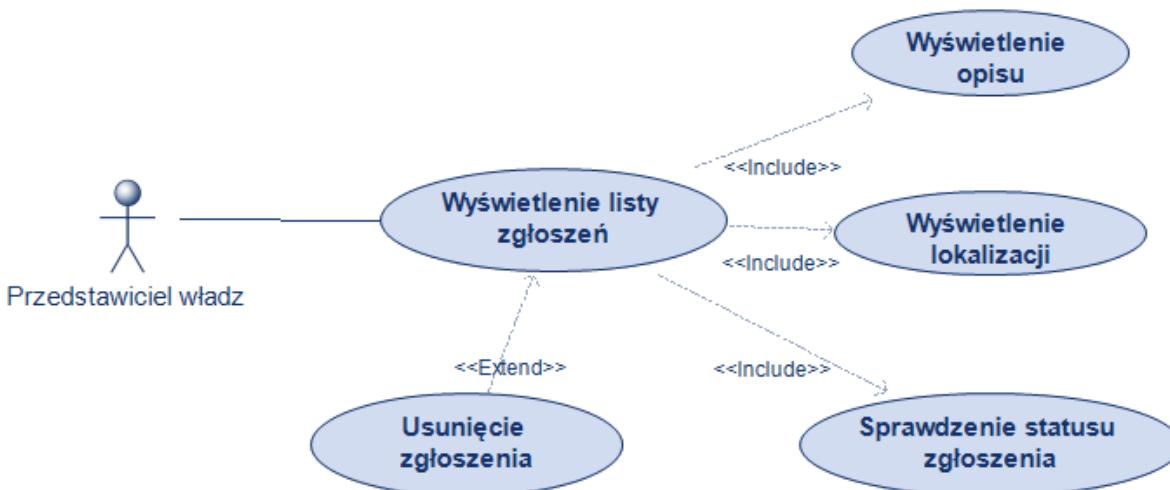
2.8.4.1 Diagram przypadków użycia dla poszkodowanego

Pierwszym aktorem jest poszkodowany, który ma możliwość zgłoszania potrzeb. Podczas zgłoszenia potrzeby musi podać informację dotyczące potrzeby oraz dodać lokalizację zgłoszenia. Może on również zobaczyć listę wszystkich swoich zgłoszeń wraz z informacjami o każdym z nich. Podczas przeglądania zgłoszeń poszkodowany może edytować wcześniej zgłoszoną potrzebę. Istnieje dodatkowo możliwość usunięcia zgłoszenia.

**2.8.4.2 Diagram przypadków użycia dla przedstawiciela władz**

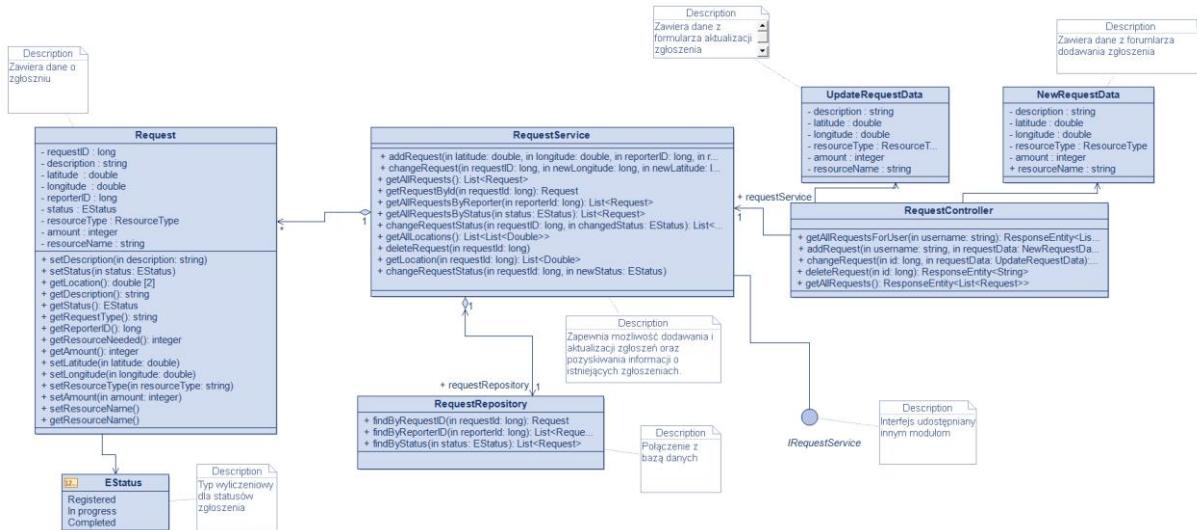
Kolejnym aktorem w naszym module jest przedstawiciel władz, który ma możliwość przeglądania listy wszystkich zgłoszeń wraz z informacjami o nich. Podczas wyświetlania listy istnieje możliwość usunięcia zgłoszenia.

Część I: Dokumentacja techniczna



2.8.4.3 Diagram klas

Poniższy diagram przedstawia klasy wchodzące w skład modułu Request. Klasa Request zawiera dane o zgłoszeniu, które później są zapisywane do bazy danych. Typ wyliczeniowy EStatus zawiera statusy, jakie może przyjąć dane zgłoszenie. Klasa RequestService implementuje interfejs IRequestService, który jest udostępniany innym modułom. Klasa RequestController pełni rolę kontrolera, który jest niezbędny do zaimplementowania graficznego interfejsu użytkownika. NewRequestData oraz UpdateRequestData to modele, które zawierają dane z formularzy dodawania i aktualizacji zgłoszenia.



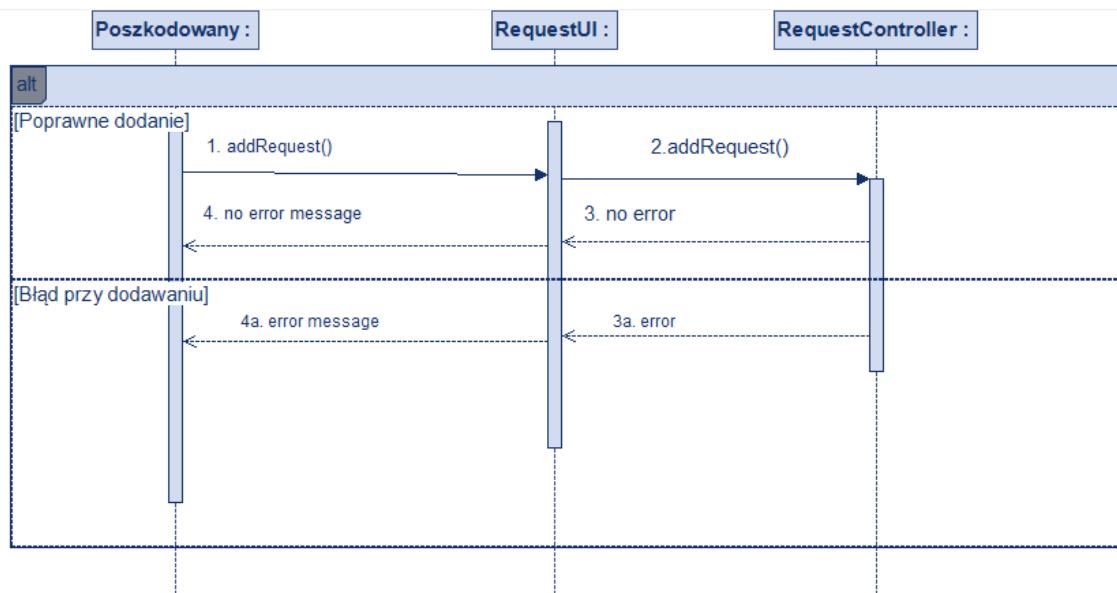
2.8.4.4 Scenariusz – Zgłaszczenie potrzeb

Nazwa	Zgłaszczenie potrzeb
Poziom ważności	Wysoki
Typ przypadków użycia	Niezbędny
Aktorzy	Poszkodowany
Krótki opis	Poszkodowany przy pomocy formularza dodaje zgłoszenie
Warunki wstępne	Musi istnieć przynajmniej 1 poszkodowany

Warunki końcowe	Nowe zgłoszenie jest tworzone i zapisywane w bazie danych
Główny przepływ zdarzeń	<ol style="list-style-type: none"> 1. Poszkodowany podaje w formularzu dane o zgłoszeniu i potwierdza dodanie zgłoszenia 2. Dane zostają wysłane do backendu 3. Przesyłany jest komunikat o poprawnym dodaniu zgłoszenia 4. System wyświetla komunikat potwierdzający dodanie zgłoszenia
Alternatywne przepływy zdarzeń	<ol style="list-style-type: none"> 3a. Przesyłany jest komunikat o błędzie 4a. System wyświetla komunikat o błędzie
Specjalne wymagania	Połączenie z bazą danych

2.8.4.5 Diagram sekwencji – Zgłaszanie potrzeb

Diagram przedstawia komunikacje pomiędzy poszczególnymi obiektami podczas zgłaszania potrzeby.



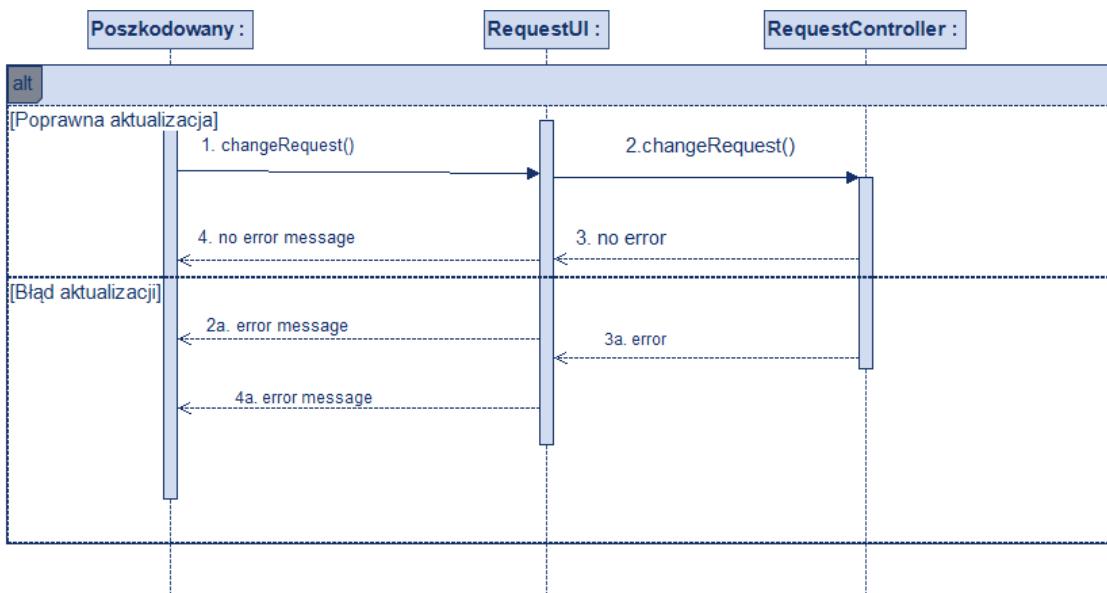
2.8.4.6 Scenariusz – Aktualizacja zgłoszenia

Nazwa	Aktualizacja zgłoszenia
Poziom ważności	Średni
Typ przypadków użycia	Pomocnicze
Aktorzy	Poszkodowany
Krótki opis	Poszkodowany przy pomocy formularza aktualizuje wcześniejsze zgłoszenie
Warunki wstępne	Musi istnieć przynajmniej 1 poszkodowany, który zgłosił potrzebę i ta potrzeba musi mieć status zarejestrowana (registered)

Warunki końcowe	Zaktualizowanie opisu wcześniej zgłoszonej potrzeby
Główny przepływ zdarzeń	<ol style="list-style-type: none"> 1. Poszkodowany wybiera zgłoszenie do aktualizacji i aktualizuje dane 2. Dane zostają wysłane do backendu 3. Przesyłany jest komunikat o poprawnej aktualizacji 4. System wyświetla komunikat potwierdzający aktualizację
Alternatywne przepływy zdarzeń	<ol style="list-style-type: none"> 2a. System wyświetla komunikat o braku możliwości aktualizacji zgłoszenia 3a. Przesyłany jest komunikat o błędzie 4a. System wyświetla komunikat o błędzie
Specjalne wymagania	Połączenie z bazą danych

2.8.4.7 Diagram sekwencji – Aktualizacja zgłoszenia

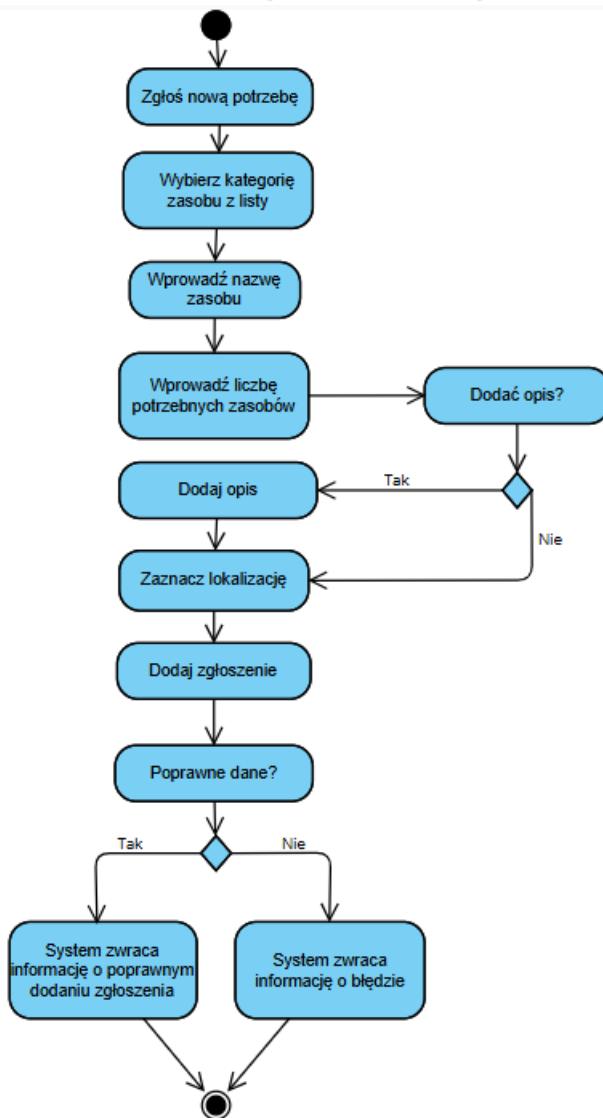
Diagram przedstawia komunikacje pomiędzy poszczególnymi obiektami podczas aktualizacji zgłoszenia.



2.8.4.8 Diagram czynności

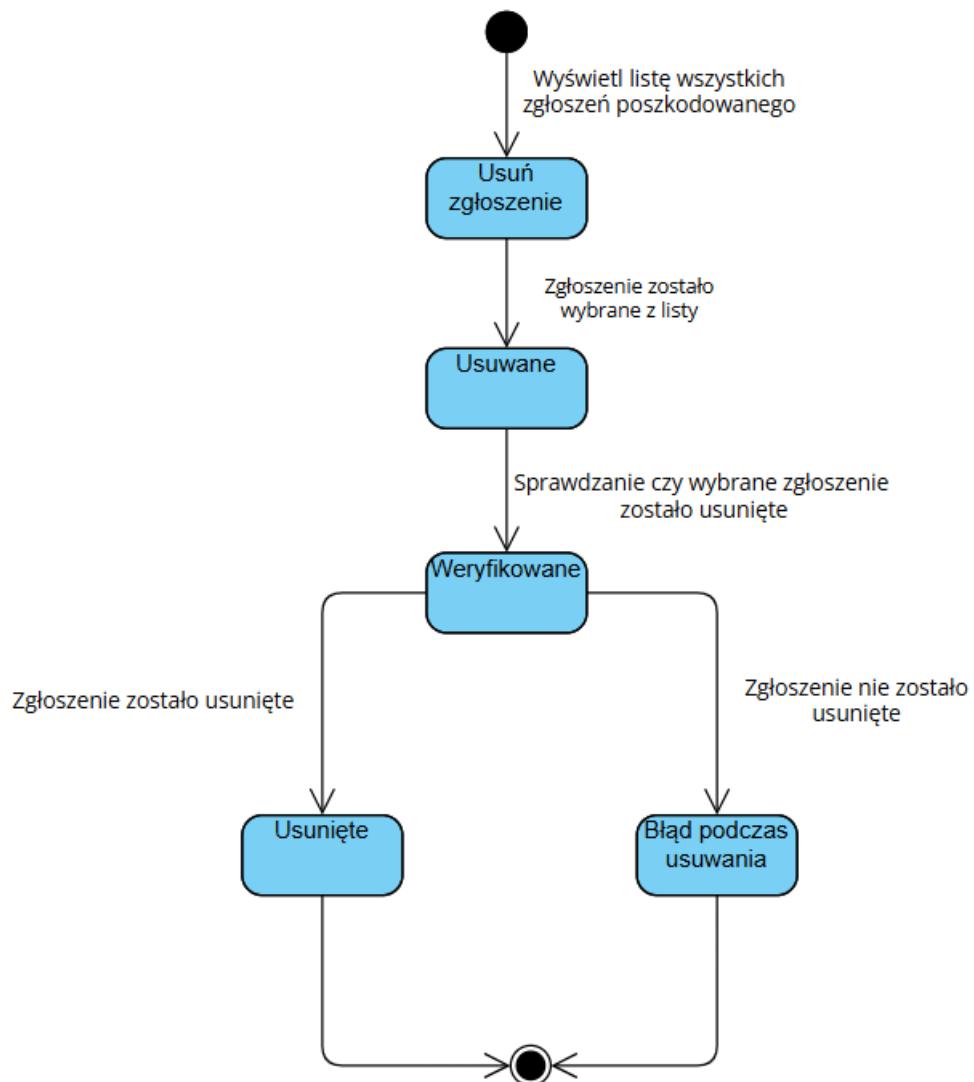
Diagram przedstawia czynności, które są wykonywane podczas dodawania nowego zgłoszenia przez poszkodowanego. Użytkownik musi najpierw nacisnąć przycisk dodawania nowego zgłoszenia. Następnie musi wybrać kategorię potrzebnego zasobu, wprowadzić jego nazwę oraz liczbę potrzebnych zasobów, później może dodać dodatkowy opis oraz koniecznie zaznaczyć lokalizację zgłoszenia. Na koniec musi potwierdzić dodanie zgłoszenia poprzez kliknięcie przycisku. Potem dane są sprawdzane i, w zależności od poprawności podanych danych, wyświetlany jest komunikat o powodzeniu lub niepowodzeniu dodania zgłoszenia.

Część I: Dokumentacja techniczna



2.8.4.9 Diagram maszyny stanowej

Diagram przedstawia stany, które może przyjąć zgłoszenie podczas usuwania zgłoszenia.



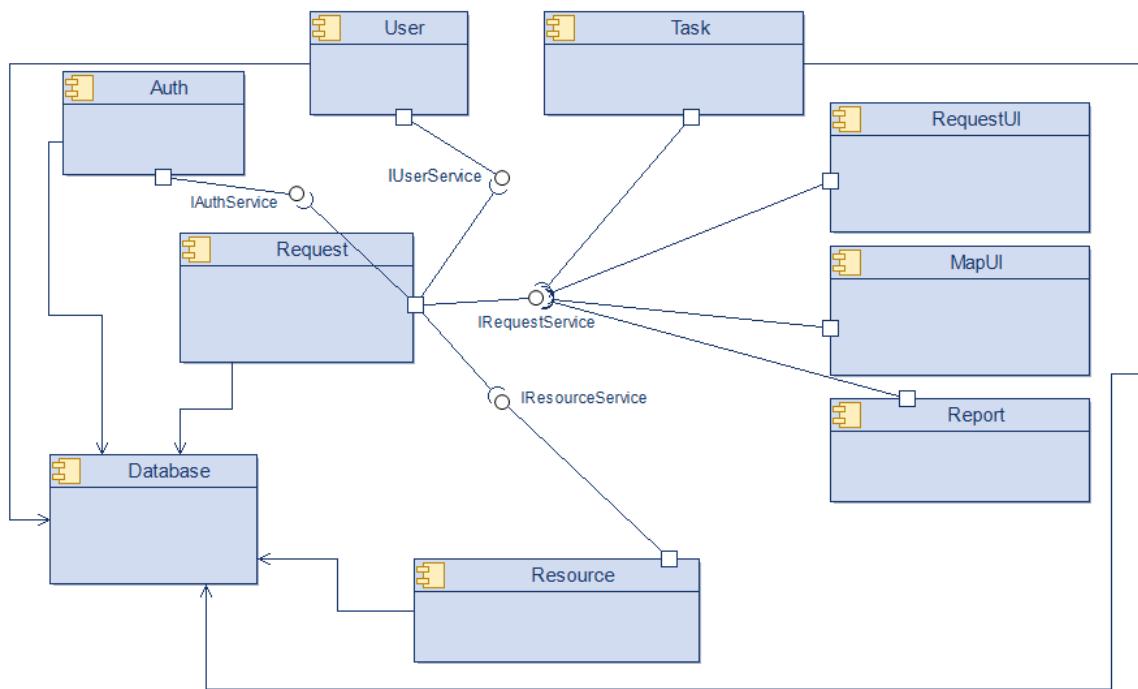
2.8.4.10 Diagram komponentów

Diagram komponentów przedstawia komponenty zawarte w naszym module oraz pozostałe komponenty systemu, które mają powiązanie z modułem Request.

Zrealizowany przez nas komponent Request korzysta z funkcjonalności udostępnianych przez komponenty Auth, User oraz Resource, a dodatkowo połączony jest z bazą danych.

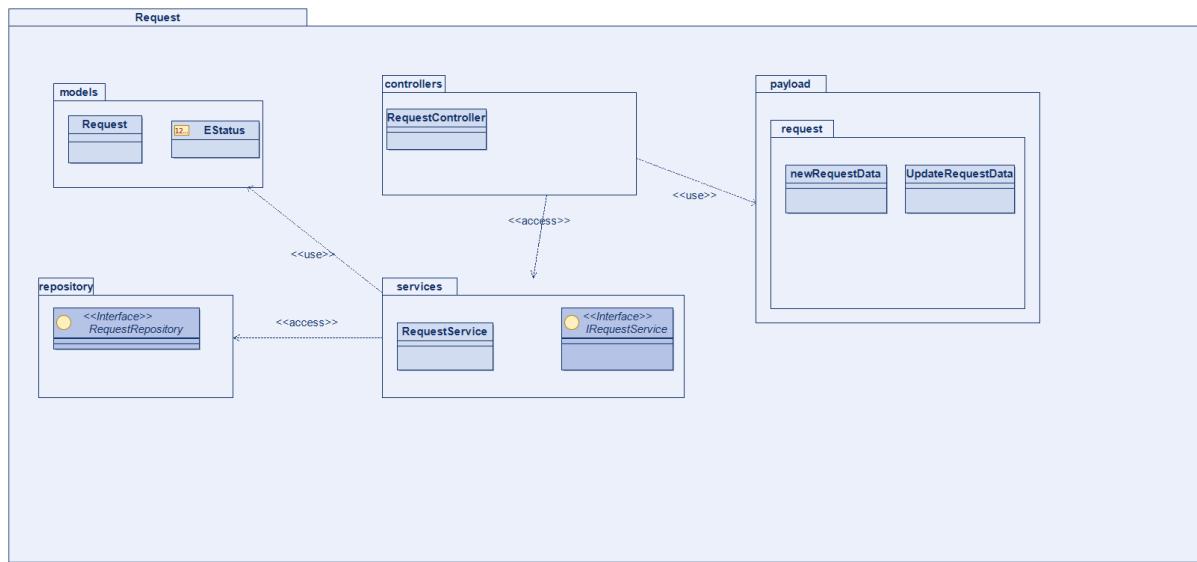
Z funkcjonalności udostępnianych przez nasz komponent korzystają Task, Report MapUI oraz zrealizowany przez nas komponent RequestUI.

Część I: Dokumentacja techniczna



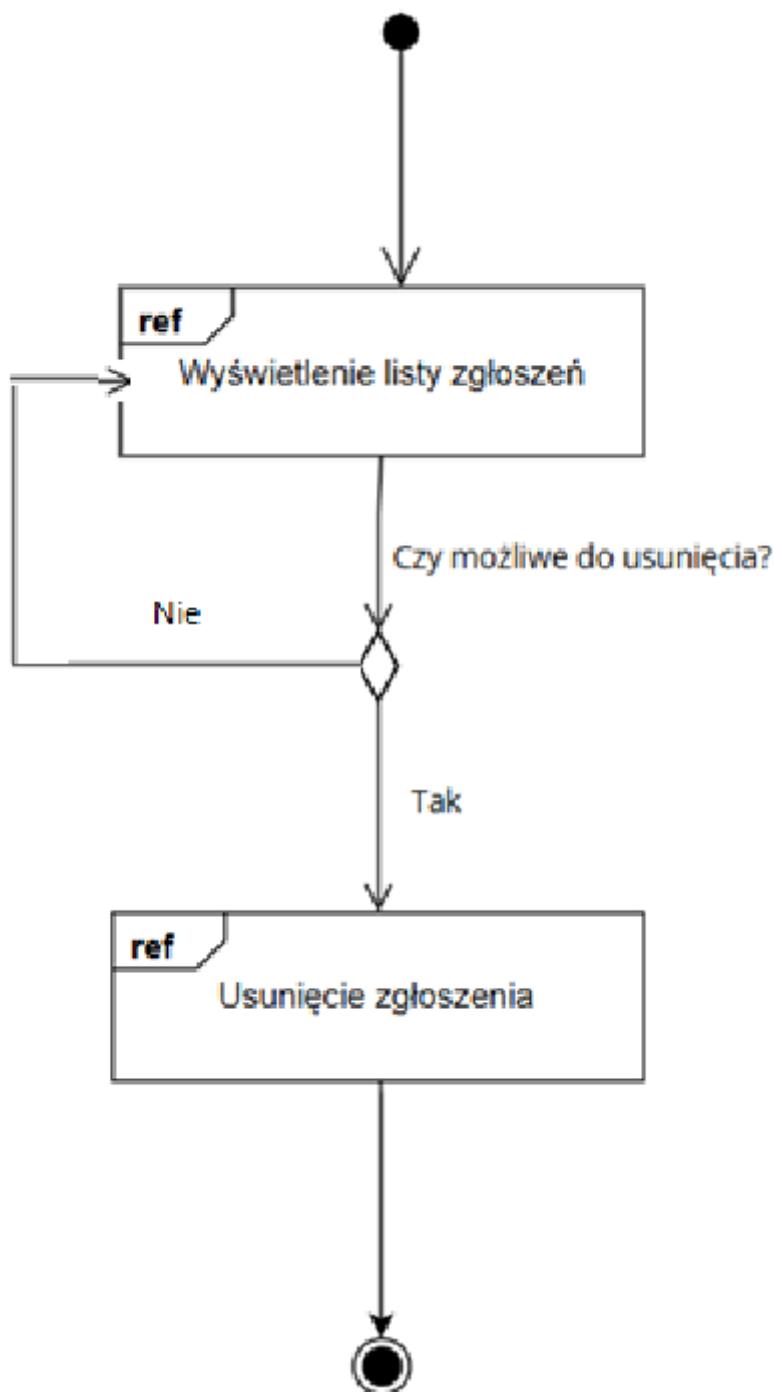
2.8.4.11 Diagram pakietów

Poniższy diagram przedstawia podział logiczny modułu na pakiety oraz zależności między tymi pakietami.

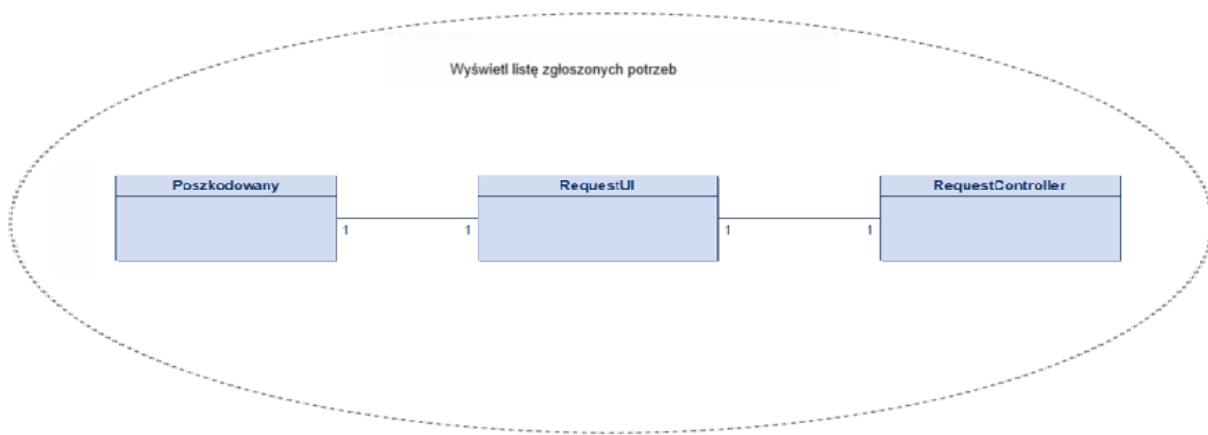


2.8.4.12 Diagram przeglądu interakcji

Diagram przeglądu interakcji przedstawia współpracę diagramów interakcji. Po wyświetleniu listy zgłoszeń użytkownik wybiera zgłoszenie, które chce usunąć. Jeśli wybrane zostało zgłoszenie, które może zostać usunięte, to jest ono usuwane. Jeśli nie – ponownie wyświetla się lista zgłoszeń.

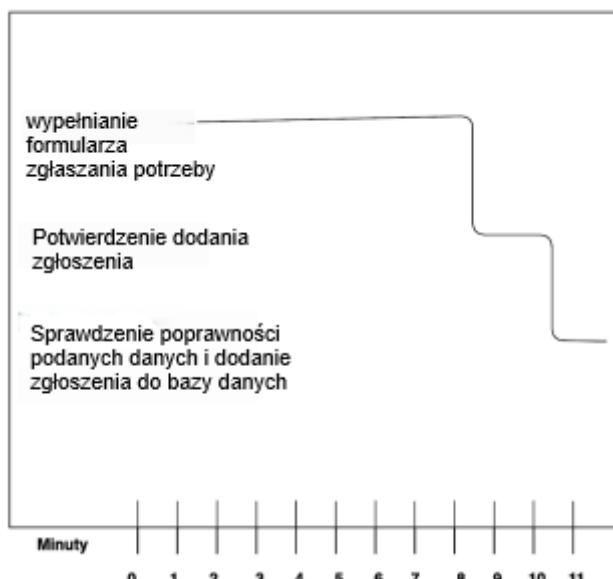


2.8.4.13 Diagram strukturalny



2.8.4.14 Diagram harmonogramowania

Poniższy diagram obrazuje zachowanie wraz z określeniem czasu, w którym następują kolejne działania. Jednostką czasu jest minuta.



2.9 Moduł Resource

2.9.1 Cel

1. Zadaniem modułu Resource jest śledzenie i zarządzanie zasobami w tym darowiznami rzeczowymi oraz finansowymi.
 - Usprawnienie procesów rejestracji, lokalizacji, monitorowania oraz przydzielania zasobów.
 - Umożliwienie przeglądania historii darowizn.
 - Wsparcie w planowaniu działań pomocowych, dzięki szczegółowych informacji dotyczących zasobów.

2.9.2 Założenia

Moduł zapewni przechowywanie różnych typów zasobów, w tym środki transportu, dary rzeczowe oraz środki finansowe.

Różne role i uprawnienia - wykonanie określonych funkcjonalności oraz dostęp do określonych zasobów, będzie możliwy tylko dla uprawnionych użytkowników.

Możliwość oznaczania zasobów jako przeterminowane, przydzielone, dostępne lub zepsute.

Każdy zasób będzie posiadał przypisaną lokalizację, tak żeby później można było go wyświetlić na mapie.

Intuicyjny i prosty w obsłudze interfejs, tak aby można było łatwo nawigować po funkcjach modułu i korzystać z jego możliwości.

2.9.3 Wymagania

1. Wymagania funkcjonalne

→ **Rejestracja zasobu**

- Organizacja może dodać nowy zasób ze szczegółowymi informacjami jak nazwa, opis, typ, ilość, miara, lokalizacja oraz data ważności.

→ **Wyświetlenie poziomu zasobów i darów**

- Organizacja oraz władze mają możliwość przeglądania zasobów i darów z podziałem na kategorie lub stan.

→ **Sprawdzenie stanu zasobu**

- Pozwala organizacji na sprawdzenie czy dany zasób jest dostępny do udostępnienia.

→ **Przydzielanie zasobu**

- Organizacja może przydziełać zasoby do zgłoszeń lub punktów dystrybucji pomocy.

→ **Odebranie zasobu**

- Organizacja może odbierać zasoby konkretnym zgłoszeniom lub punktom dystrybucji pomocy.

→ **Aktualizacja informacji zasobu**

- Organizacja ma możliwość zmiany opisu, lokalizacji, ilości oraz oznaczenia zasobu jako zniszczony.

→ **Przekazanie daru**

Część I: Dokumentacja techniczna

- Darczyńcy mogą rejestrować nowe darowizny rzeczowe lub finansowe.
- System umożliwia przypisywanie darowizn do konkretnych organizacji pomocowych.

→ **Wyświetlenie historii darowizn**

- Darczyńcy mogą przeglądać historię darowizn z podziałem na daty i rodzaje zasobów.

→ **Śledzenie wykorzystania darów.**

- Darczyńcy mają możliwość przeglądania jak ich dary są wykorzystywane.

→ **Filtrowanie zasobów**

- Organizacja i władze mogą filtrować zasoby na podstawie ich dostępności lub kategorii.
- Możliwość filtrowania nazwy za pomocą słów kluczowych.

→ **Sortowanie zasobów**

- Organizacja i władze mogą sortować zasoby na podstawie ich nazwy, ilości, daty dodania czy terminu ważności.

2. Wymagania niefunkcjonalne→ **Bezpieczeństwo**

- Mechanizmy uwierzytelniania i autoryzacji oparte na rolach użytkowników.

→ **Wydajność**

- System musi reagować na akcje użytkownika w czasie krótszym niż 2 sekundy.

→ **Dostępność**

- Moduł musi być dostępny przez minimum 99,9% czasu.

→ **Skalowalność**

- Moduł musi obsługiwać rosnącą liczbę użytkowników i zasobów bez utraty wydajności.
- Łatwość w dodawaniu nowych funkcji lub rozbudowie już istniejących.

→ **Intuicyjność**

- Moduł ma być łatwy w obsłudze i w zrozumieniu.

2.9.4 Diagramy**2.9.4.1 Diagram przypadków użycia**

Część I: Dokumentacja techniczna



Diagram przypadków użycia przedstawia interakcje jakie Darczyńca, Organizacja oraz Władze mogą wykonywać. Wlicza się w to wyświetlenie darowizn, zasobów, ich poziom, przekazanie daru, rejestrację, aktualizację, odebranie, przydzielenie zasobu oraz filtrowanie i sortowanie.

Część I: Dokumentacja techniczna

Diagram klas

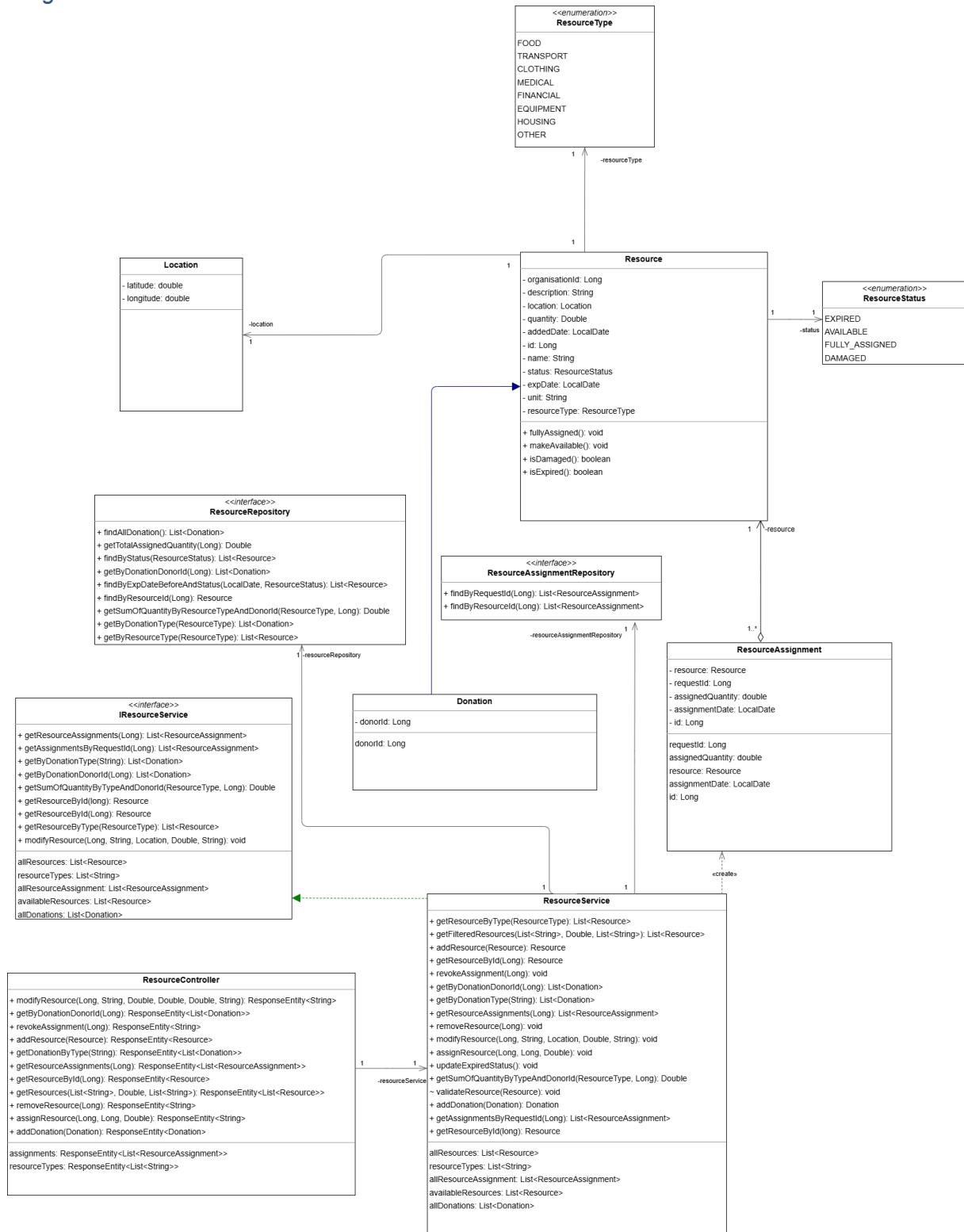


Diagram klas przedstawia strukturę modułu Resource w postaci klas, ich atrybutów, metod, relacji oraz interfejsów.

Część I: Dokumentacja techniczna

Resource Controller jest głównym elementem modułu, dzięki któremu możemy udostępnić endpointy, dzięki którym można się połączyć z interfejsem graficznym. Moduł Resource udostępnia interfejs IResourceService innym modułom dzięki czemu mogą korzystać z wybranych metod naszych klas. Klasa Location pozwala zapisać lokalizację zasobu za pomocą szerokości oraz długości geograficznej. W Enum ResourceStatus są zawierane wszystkie stany jakie zasób może przyjąć, a w Enum ResourceType wszystkie typy. Dodatkowo klasa ResourceAssignment odpowiada za przypisanie zasobu do konkretnego zgłoszenia (Request).

2.9.4.2 Scenariusz przypadku użycia - przydzielenie zasobu

Nazwa	Przydzielenie zasobu
Twórcy	Jakub Cendalski, Kacper Przybylski - Projektanci
Poziom ważności	Bardzo Ważny
Typ przypadku użycia	Ogólny
Aktorzy	Organizacja
Krótki opis	Przydzielenie zasobu do zgłoszeń lub punktów dystrybucji pomocy
Warunki wstępne	Musi istnieć zasób
Warunki końcowe	Zasób zostanie poprawnie przydzielony
Główny przepływ zdarzeń	Organizacja próbuje przydzielić zasób. System weryfikuje uprawnienia sesji użytkownika. System wyświetla listę dostępnych zasobów. Organizacja wybiera zasób. Organizacja przydziela zasoby zgłoszeniu. Przypisanie zasobu jest zapisywane w bazie danych.
Alternatywny przepływ zdarzeń	2a. Brak wystarczających uprawnień. 3a. Brak dostępnych zasobów
Specjalne wymagania	Wysoka niezawodność systemu Czas przetwarzania operacji przydzielenia zasobu nie przekracza 2 sekund.

Część I: Dokumentacja techniczna

2.9.4.3 Diagram sekwencji – przydzielanie zasobu

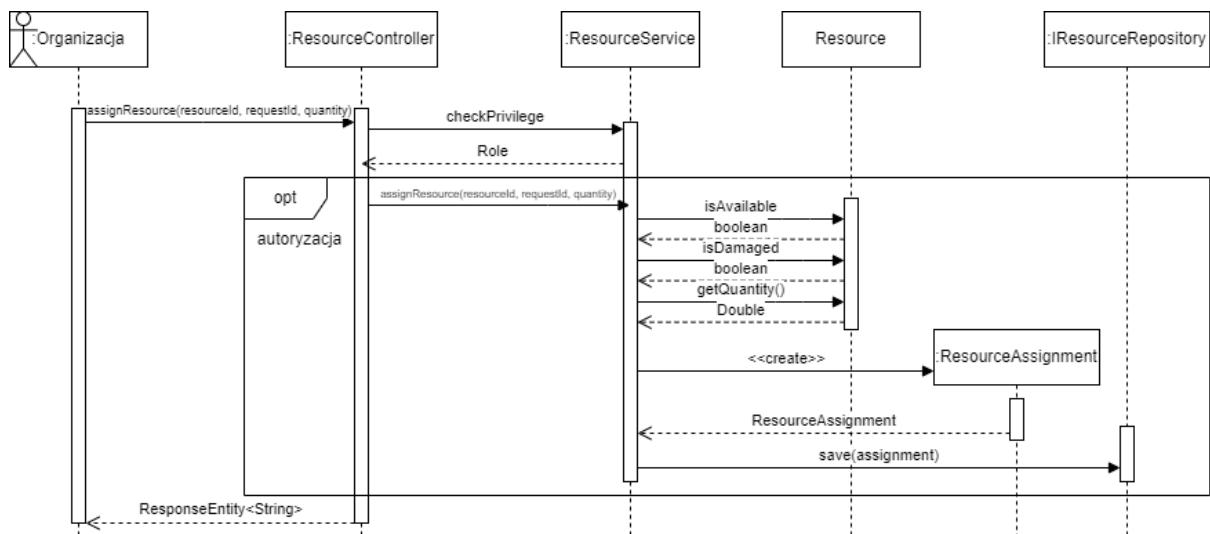


Diagram przedstawia realizację procesu przypisania zasobu korzystając ze sprawdzania autoryzacji użytkownika. Zasób jest przypisywany oraz jego przypisanie jest zapisane w bazie danych za pomocą klasy `IResourceRepository`.

2.9.4.4 Scenariusz przypadku użycia - przekazanie daru

Nazwa	Przekazanie daru
Twórcy	Jakub Cendalski, Kacper Przybylski - Projektanci
Poziom ważności	Bardzo wysoki
Typ przypadku użycia	Ogólny
Aktorzy	Darczyńca
Krótki opis	Przekazanie daru przez darczyńcę
Warunki wstępne	Zgłoszenie chęci przekazania daru
Warunki końcowe	Zarejestrowanie daru w repozytorium
Główny przepływ zdarzeń	<p>Darczyńca dokonuje przekazania daru klikając w przycisk na stronie.</p> <p>Darczyńca wpisuje dane darowizny w formularzu i wysyła dar.</p> <p>Weryfikacja uprawnień sesji użytkownika.</p> <p>Dodanie daru do bazy danych.</p>
Alternatywny przepływ zdarzeń	2a. Niepoprawne dane w formularzu 3a. Użytkownik nie ma wystarczających uprawnień
Specjalne wymagania	Kontrola dostępu do funkcji w zależności od sesji użytkownika

Część I: Dokumentacja techniczna

2.9.4.5 Diagram sekwencji – przekazanie daru

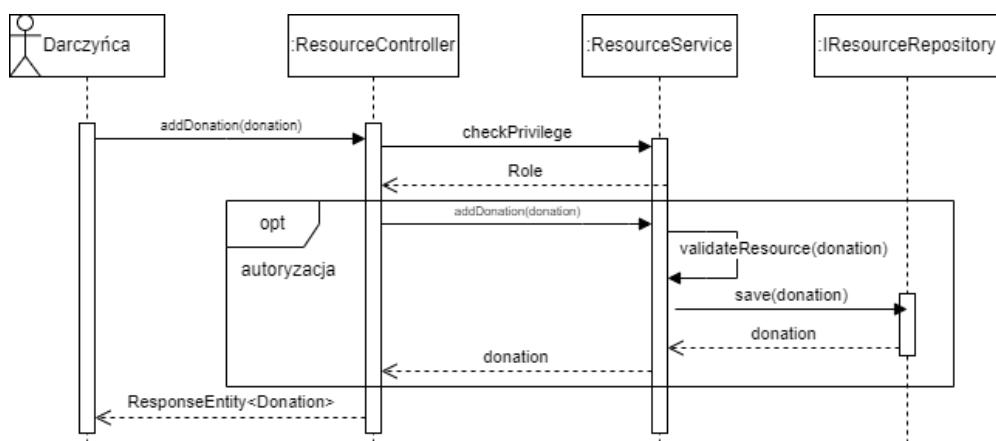


Diagram przedstawia realizację procesu przekazania daru. Odbywa się to przez użytkownika, który wypełnia formularz oraz klasy ResourceController, ResourceService oraz IResourceRepository. IResourceRepository odpowiada za komunikację z bazą danych a Controller za autoryzację użytkownika oraz komunikację z interfejsem graficznym.

2.9.4.6 Scenariusz przypadku użycia - aktualizacja zasobu

Nazwa	Aktualizacja zasobu
Twórcy	Jakub Cendalski, Kacper Przybylski - Projektanci
Poziom ważności	Ważny
Typ przypadku użycia	Ogólny
Aktorzy	Organizacja
Krótki opis	Aktualizacja zasobu z listy
Warunki wstępne	Musi istnieć zasób
Warunki końcowe	Dane zasobu zostaną zaktualizowane
Główny przepływ zdarzeń	Organizacja wybiera zasób do modyfikacji z listy. Organizacja wypełnia dane w formularzu z danymi do modyfikacji Weryfikacja uprawnień sesji użytkownika. Sprawdzenie poprawności wpisanych danych w formularzu. Aktualizacja zasobu w bazie danych
Alternatywny przepływ zdarzeń	3a. Brak wystarczających uprawnień. 4a. Źle wypełniony formularz
Specjalne wymagania	Wysoka niezawodność systemu. Czas przetwarzania operacji przydzielenia zasobu nie przekracza 2 sekund.

Część I: Dokumentacja techniczna

2.9.4.7 Diagram sekwencji – aktualizacja zasobu

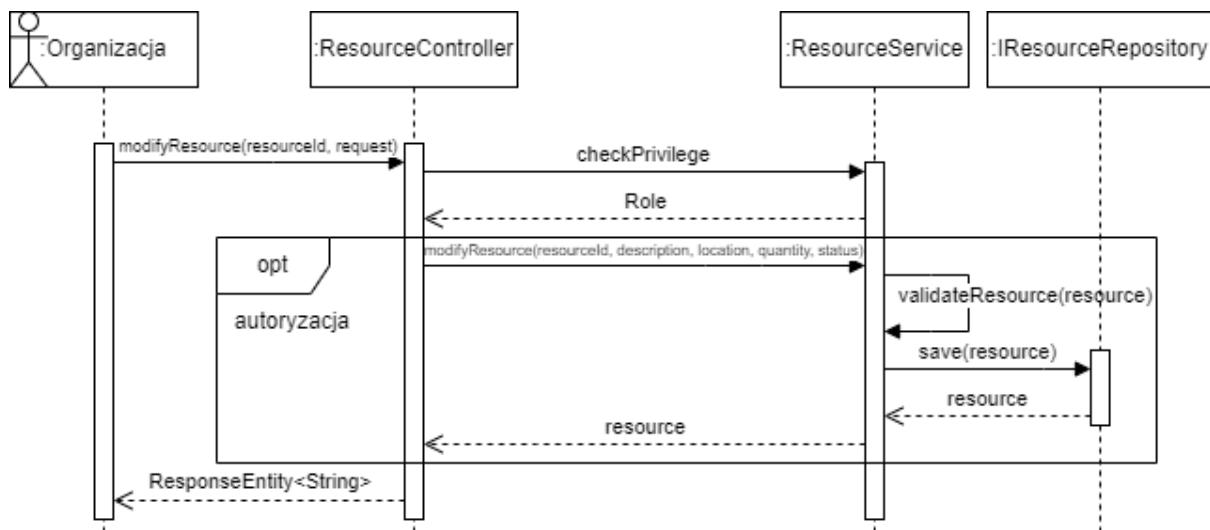


Diagram przedstawia realizację procesu aktualizacji zasobu Resource. ResourceRepository odpowiada za komunikację z bazą danych a Controller za autoryzacje użytkownika. W ResourceService odbywa się walidacja danych oraz zapis danych przez klasę IResourceRepository.

2.9.4.8 Diagram czynności – Przydzielenie zasobu

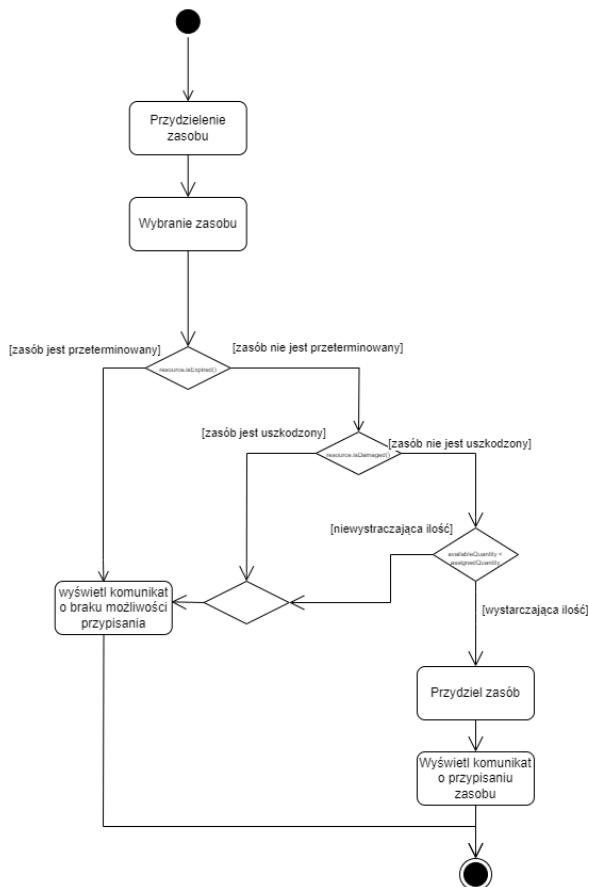


Diagram czynności przedstawia proces przydzielania zasobu, można tutaj wyróżnić takie czynności jak sprawdzanie czy zasób jest przeterminowany, uszkodzony lub czy jego ilość jest wystarczająca. Cały proces odbywa się poprzez formularz, w którym wybiera się zasób oraz zapotrzebowanie które jest do niego dobierane.

2.9.4.9 Diagram czynności - Rejestracja zasobu

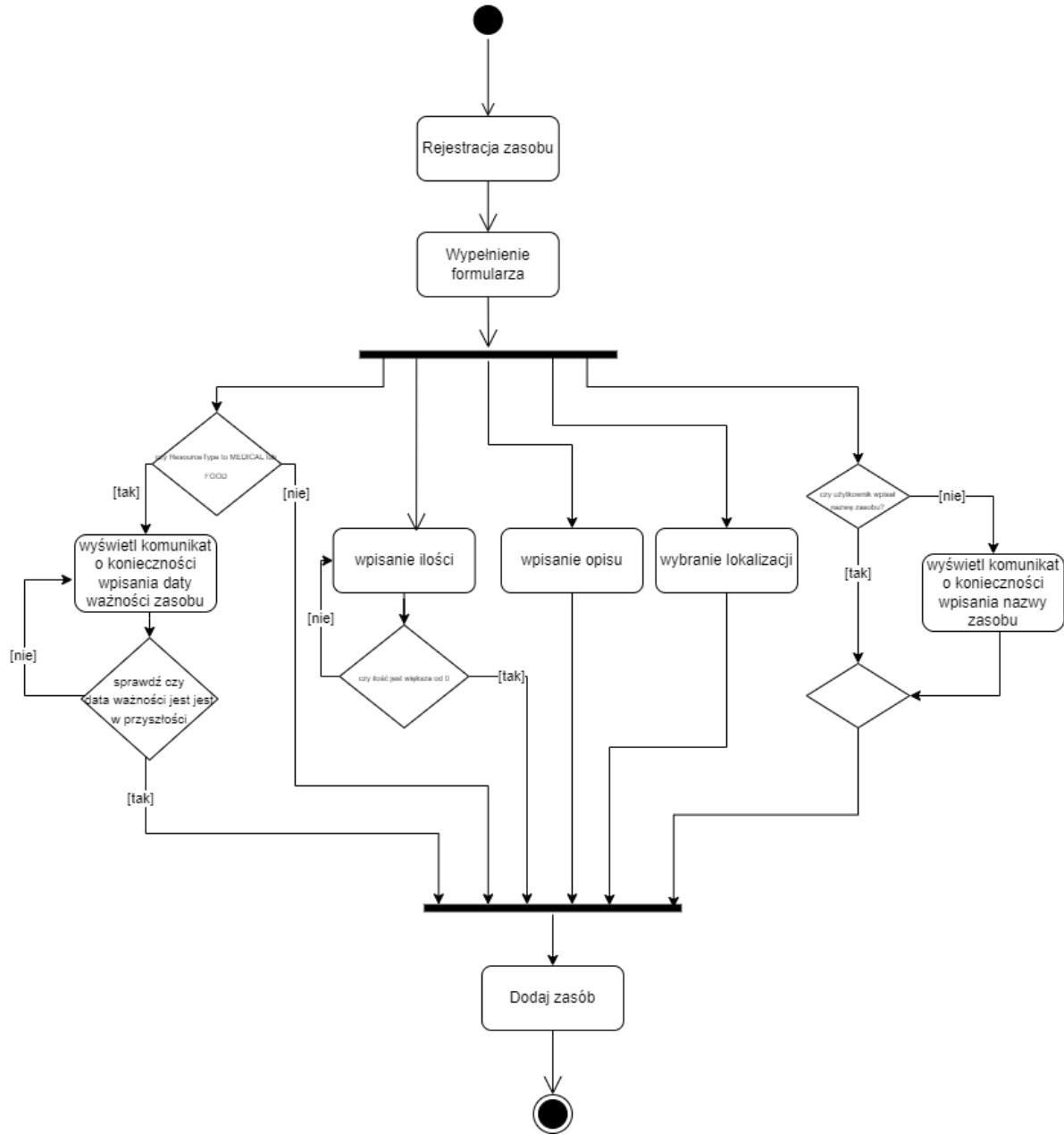
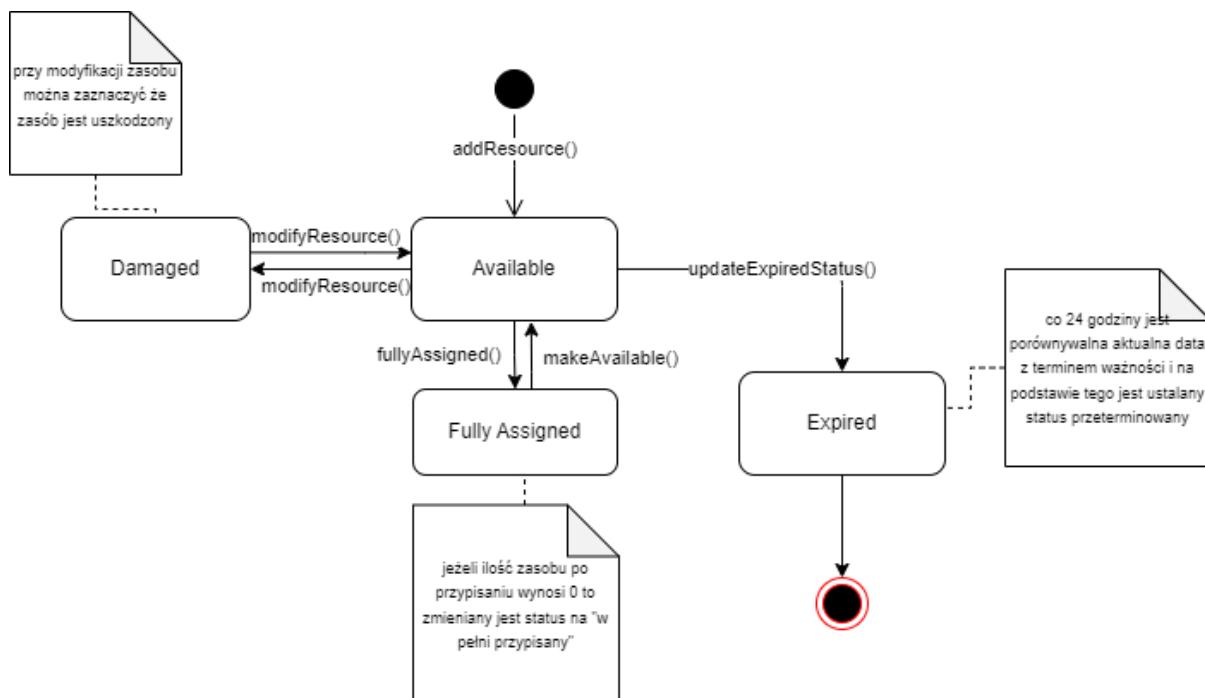


Diagram czynności przedstawia proces rejestracji zasobu, gdzie organizacja ma możliwość wyboru daty ważności (jeżeli dodaje jedzenie bądź rzeczy medyczne), lokalizacji, opisu, ilości oraz nazwy. Po pomyślnym wypełnieniu formularza, zasób zostaje dodany do listy zasobów.

Część I: Dokumentacja techniczna

2.9.4.10 Diagram maszyny stanowej (klasa Resource)



2.9.4.11 Diagram komponentów

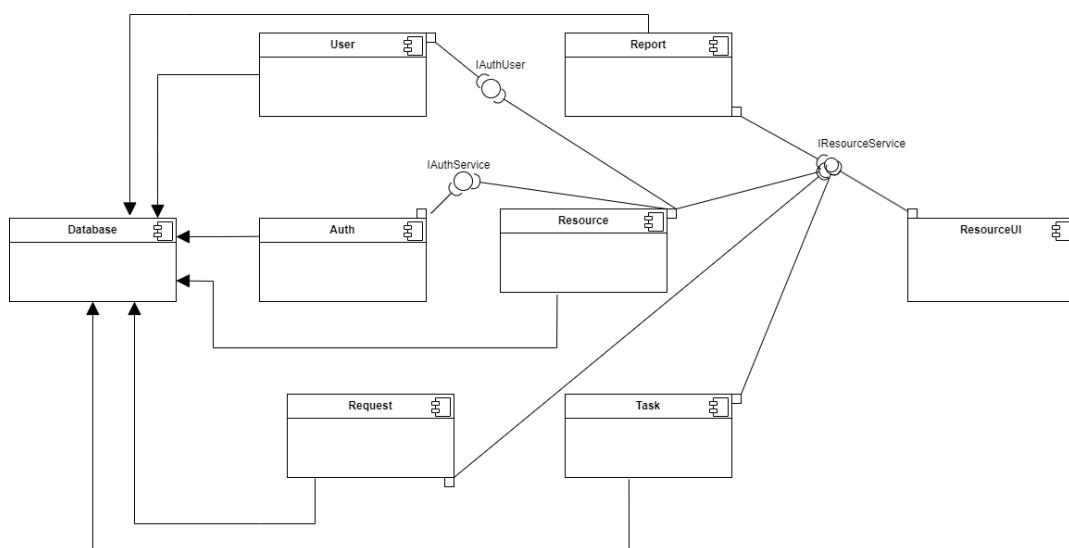


Diagram komponentów przedstawia strukturę komponentów, z których moduł Resource korzysta oraz którym udostępnia inne metody. IResourceService udostępnia API, dzięki któremu moduły Task, Request oraz Report mogą uzyskać informacje o aktualnych zasobach, darowiznach oraz ich użyciu. IAuthService pozwala na weryfikację autoryzacji, a IAuthUser udostępnia информацию o organizacjach (do wyboru organizacji przy przekazaniu daru) oraz użytkownikach (głównie ich identyfikator używany do weryfikacji, które zasoby zwrócić określonej organizacji/darczyńcy).

Część I: Dokumentacja techniczna

2.9.4.12 Diagram przeglądu interakcji

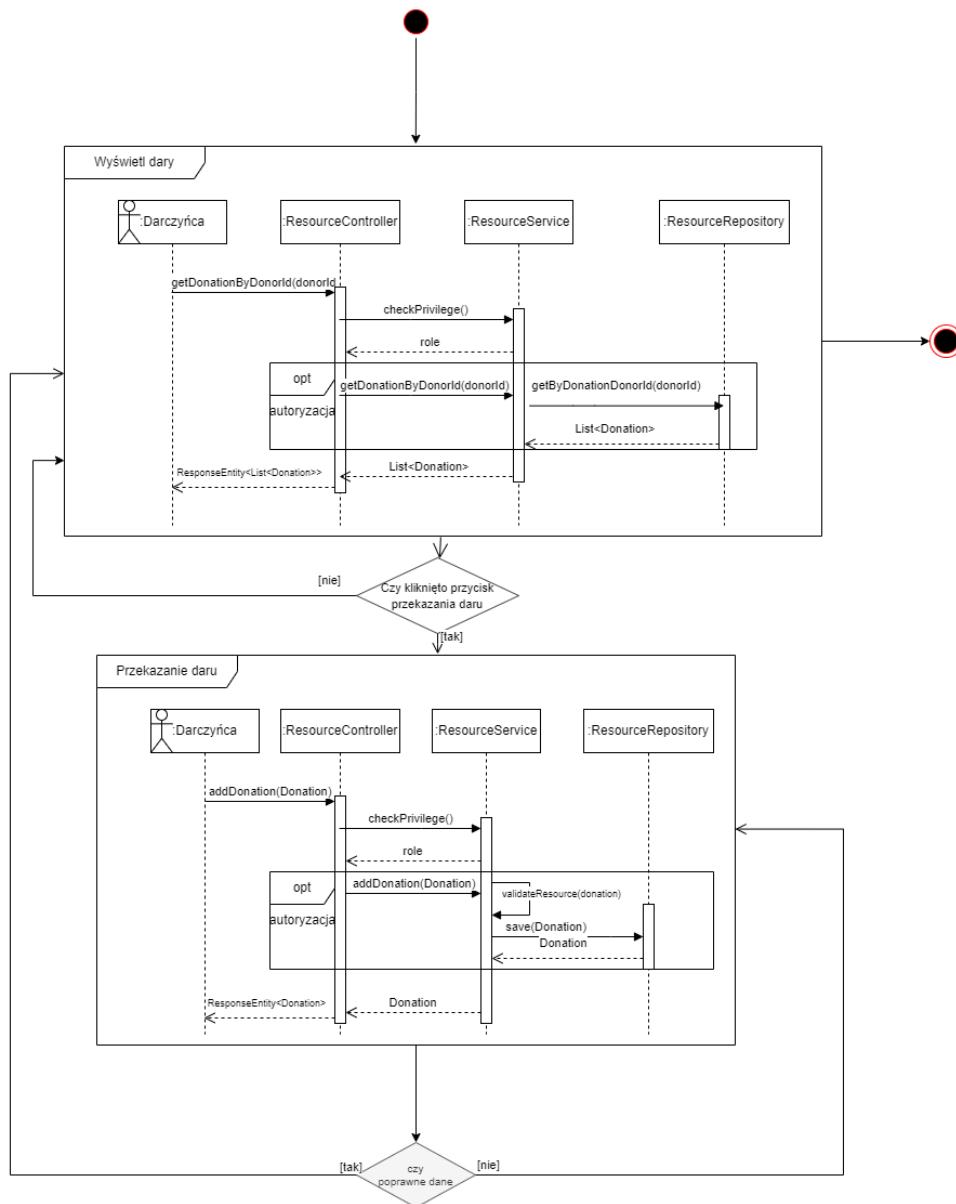


Diagram przedstawia sterowanie operacjami pomiędzy przekazywaniem daru oraz wyświetlaniem. Żeby przekazać dar, najpierw zostanie wyświetlona lista darów oraz przycisk po którego kliknięciu pojawi się formularz przekazywania daru. Po przesłaniu formularza jest sprawdzana poprawność danych, gdy zostanie spełniona to darowizna jest dodawana do listy darów oraz zostaje wyświetlona darczyńcy.

Część I: Dokumentacja techniczna

2.9.4.13 Diagram harmonogramowania

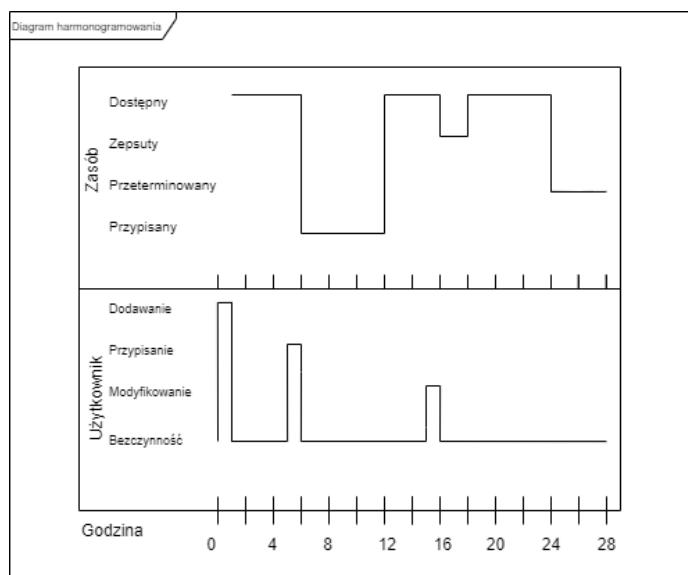


Diagram przedstawia zmiany stanów użytkownika i zasobu w czasie. Zasób może zmieniać swój stan na dostępny, zepsuty, przeterminowany zależnie od akcji użytkownika oraz terminu ważności zasobu.

2.9.4.14 Diagram pakietów

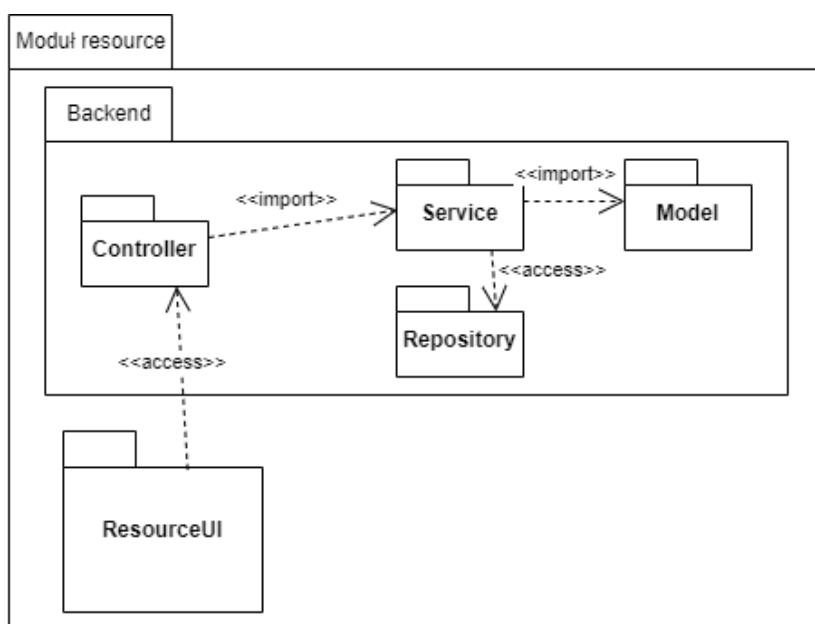


Diagram przedstawia strukturę pakietów wewnątrz modułu Resource. Każdy pakiet grupuje klasy, które są odpowiedzialne za pełnienie określonej funkcjonalności. Models zawiera obiekty z danymi na których będziemy operować. Repository pozwala zapisywać dane w bazie danych, Service umożliwia wykonywać operacje logiczne, a Controller udostępnia endpointy dzięki którym ResourceUI (graficzny interfejs) wyświetli i obsłuży dane podane przez użytkownika.

2.9.4.15 Diagram strukturalny

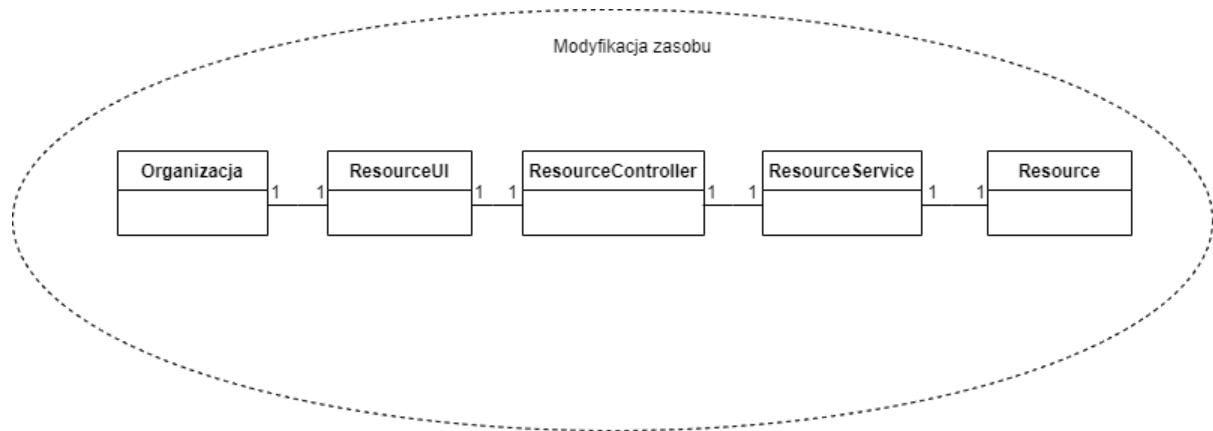


Diagram przedstawia kooperację aktora Organizacji, modułu ResourceUI oraz klas ResourceController, ResourceService oraz Resource. Pokazuje on modyfikację zasobu oraz w jaki sposób klasy oraz aktorzy mają relacje w trakcie tej kooperacji.

2.10 Moduł Task

2.10.1 Cel

1. Celem modułu Task jest skuteczne zarządzanie procesem realizacji zadań przez wolontariuszy w organizacji. Moduł ten ma na celu:
 - Ułatwienie organizacji i nadzoru pracy
 - Monitorowanie postępu
 - Usprawnienie komunikacji
 - Ocena efektywności

2.10.2 Założenia

Centralizacja zarządzania zadań: moduł umożliwia organizacjom wykonywanie operacji na zadaniach (tworzenie, przydzielanie, edycja, ocena) w jednym miejscu, co zapewnia spójność i uporządkowanie.

Różne role i uprawnienia: System będzie przewidywać różne poziomy dostępu i uprawnień.

Intuicyjny interfejs użytkownika: Interfejs powinien być prosty i intuicyjny zarówno dla organizatorów, jak i dla wolontariuszy, aby łatwo można było nawigować po funkcjach modułu i korzystać z jego możliwości.

2.10.3 Wymagania

1. Wymagania funkcjonalne:

- Przegląd zadań – organizacja ma możliwość wyświetlenia listy zadań, oraz informacji na ich temat, a wolontariusz ma możliwość przeglądania swoich zadań
- Tworzenie zadania – organizacje tworzą zadania, z możliwością dodawania szczegółowych elementów (tytuł, opis, organizacja zgłaszająca, lista wolontariuszy i zasobów, lokalizacja, status, ocena).
- Przydzielanie wolontariuszy zadaniom – organizacje mają możliwość przydzielania zadań poszczególnym wolontariuszom lub grupom.
- Monitorowanie zadań - organizacje mają dostęp do widoku podsumowującego postęp pracy
- Filtrowanie zadań - organizacja i wolontariusz ma możliwość filtrowania zadań na podstawie ich priorytetu i lokalizacji
- Edycja zadań - organizacja ma możliwość edytowania poszczególnych części zadania, a wolontariusz jest w stanie zmienić ich status
- Ocena zadań – organizacja mają możliwość oceniania wykonanych zadań oraz dodawania informacji zwrotnej

2. Wymagania niefunkcjonalne

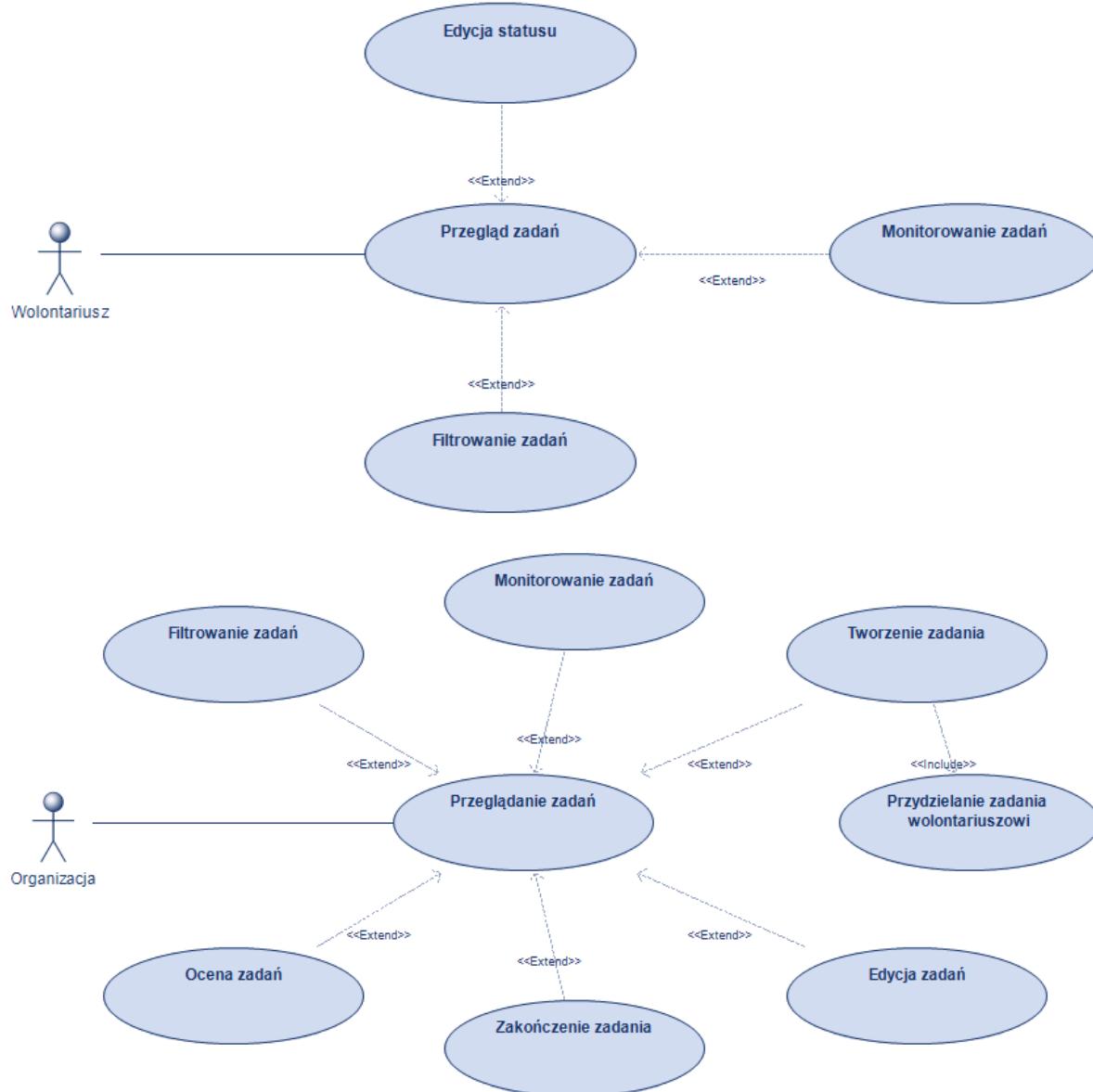
- Dostępność – moduł powinien być dostępny przez cały okres działania systemu
- Skalowalność - moduł powinien być zaprojektowany w sposób umożliwiający jego rozbudowę w przyszłości.
- Intuicyjność – czytelność i łatwość zrozumienia modułu
- Kontrola dostępu w zależności od roli - moduł musi zapewniać różnicowanie dostępu do funkcjonalności w zależności od wybranego aktora

Część I: Dokumentacja techniczna

- Wydajność - moduł powinien obsługiwać działania użytkowników bez zauważalnych opóźnień, nawet przy dużej liczbie aktywnych zadań.

2.10.4 Diagramy

2.10.4.1 Diagramy przypadków użycia



Diagramy przypadków użycia przedstawiają interakcje między użytkownikami systemu a funkcjonalnościami modułu Task w systemie koordynacji pomocy humanitarnej. Diagramy obrazują dwie główne role użytkowników: Organizacja oraz Wolontariusz, a także związane z nimi przypadki użycia.

Część I: Dokumentacja techniczna

2.10.4.2 Diagram klas

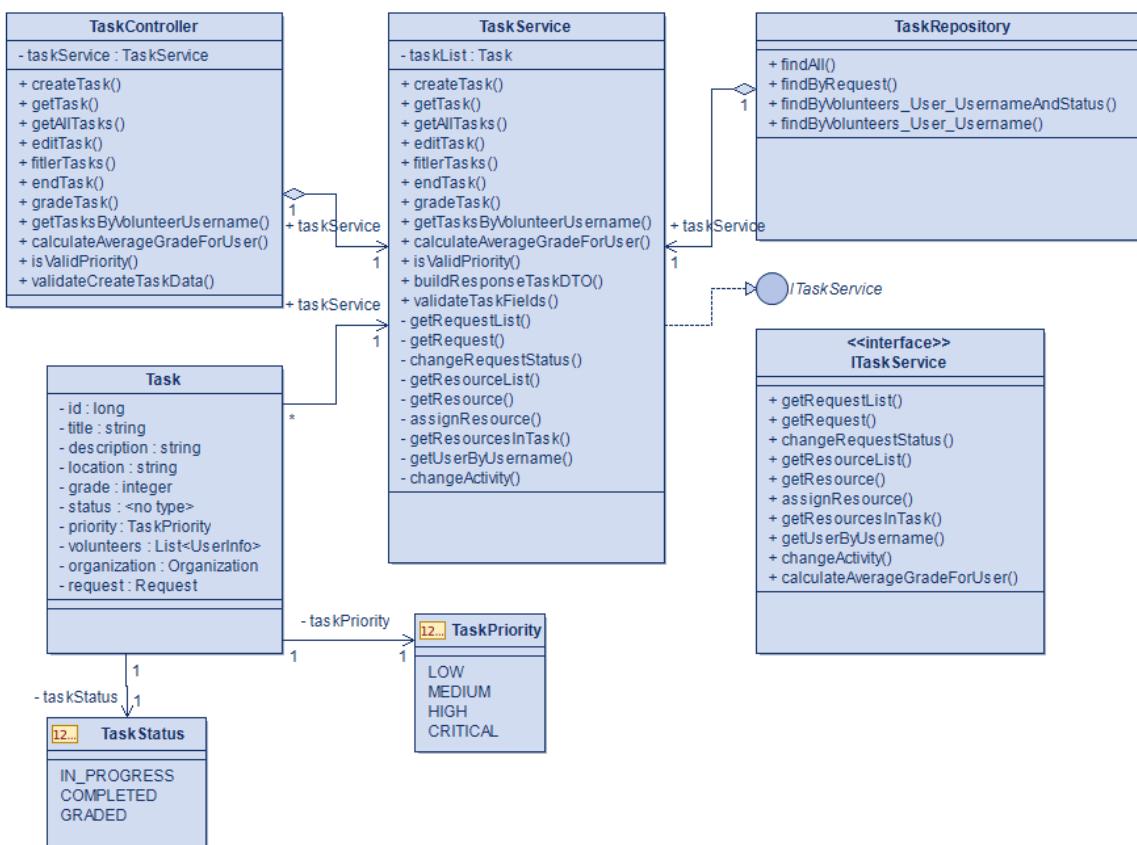


Diagram klas przedstawia statyczną strukturę modułu Task w systemie koordynacji pomocy humanitarnej. Centralnym elementem diagramu jest klasa TaskService, która zawiera większość logiki biznesowej modułu. Klasa ta implementuje interfejs ITaskService, umożliwiający kontakt z innymi modułami. Obiekty typu Task służą do przechowywania szczegółowych informacji o zadaniach. Klasa TaskService korzysta z TaskRepository do interakcji z warstwą bazy danych, zapewniając trwałość danych dotyczących zadań. Użytkownik systemu wchodzi w interakcję z modułem Task za pośrednictwem kontrolera TaskController, który stanowi warstwę komunikacji między interfejsem użytkownika a logiką biznesową. TaskController przekazuje żądania do TaskService, zarządzając przepływem danych w systemie.

Część I: Dokumentacja techniczna

2.10.4.3 Diagramy interakcji i scenariusze

Nazwa	Edycja zadań
Twórcy	Kacper Stasiak, Kacper Witek - Projektanci
Poziom ważności	Średni
Typ przypadku użycia	Ogólny, istotny
Aktorzy	Organizacja
Krótki opis	Edycja zadania
Warunki wstępne	Musi istnieć zadanie
Warunki końcowe	Zadanie uległo zmianie
Główny przepływ zdarzeń	Organizacja próbuje edytować zadanie System wywołuje funkcję editTask dla określonego zadania System weryfikuje uprawnienia sesji użytkownika System zmienia szczegóły zadania System aktualizuje repozytorium
Alternatywny przepływ zdarzeń	2a. dane zadanie nie istnieje 3a. sesja użytkownika nie ma wystarczających uprawnień
Specjalne wymagania	Wysoka niezawodność systemu Krótki czas odpowiedzi

Część I: Dokumentacja techniczna

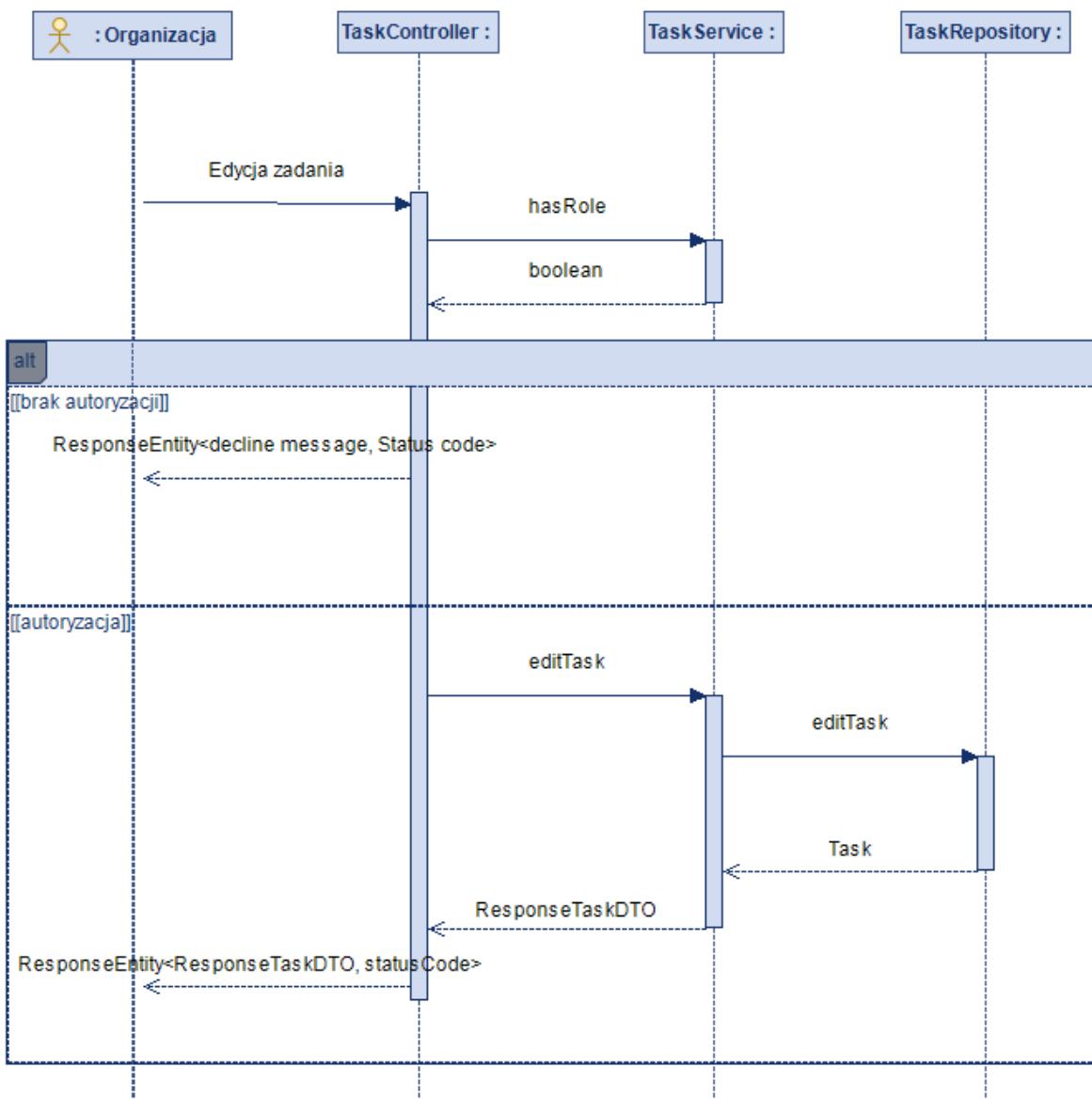


Diagram sekwencji przedstawiający przepływ komunikatów między aktorem "Organizacja", a klasami TaskController, TaskService i TaskRepository podczas realizacji procesu edycji zadania.

Część I: Dokumentacja techniczna

Nazwa	Tworzenie zadania
Twórcy	Kacper Stasiak, Kacper Witek - Projektanci
Poziom ważności	Wysoki
Typ przypadku użycia	Ogólny, istotny
Aktorzy	Organizacja
Krótki opis	Stworzenie zadania
Warunki wstępne	Musi istnieć zgłoszona potrzeba pomocy
Warunki końcowe	System dodaje zadanie do repozytorium zadań
Główny przepływ zdarzeń	<p>System weryfikuje uprawnienia sesji użytkownika</p> <p>Organizacja wybiera prośbę</p> <p>Organizacja wybiera zasoby</p> <p>Organizacja przydziela zadanie wolontariuszom</p> <p>System tworzy zadanie</p> <p>Zadanie dodawane jest do repozytorium</p>
Alternatywny przepływ zdarzeń	<p>1a. sesja użytkownika nie ma wystarczających uprawnień</p> <p>2a. dana prośba nie istnieje</p>
Specjalne wymagania	<p>Wysoka niezawodność systemu</p> <p>Krótki czas odpowiedzi</p>

SYSTEM KOORDYNACJI POMOCY HUMANITARNEJ

Część I: Dokumentacja techniczna

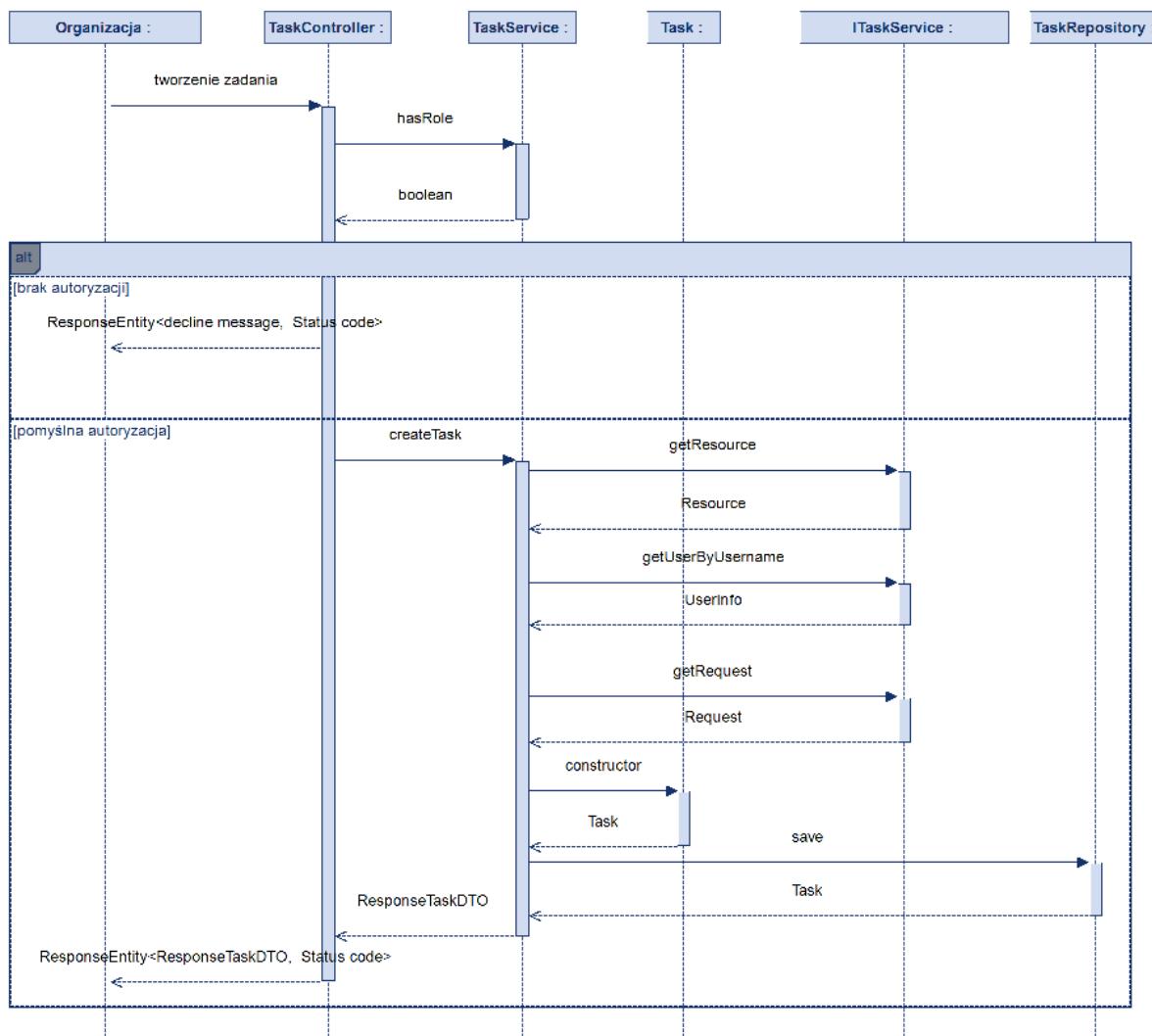


Diagram sekwencji przedstawiający przepływ komunikatów między aktorem "Organizacja", a klasami TaskController, TaskService, TaskRepository, Task i ITaskService podczas realizacji procesu tworzenia zadania.

Część I: Dokumentacja techniczna

Nazwa	Ocenianie zadania
Twórcy	Kacper Stasiak, Kacper Witek - Projektanci
Poziom ważności	Niski
Typ przypadku użycia	Ogólny
Aktorzy	Organizacja
Krótki opis	Przyznanie oceny zadaniu
Warunki wstępne	Musi istnieć zadanie, które jest skończone i nieocenione
Warunki końcowe	Ocena zadania uległa zmianie
Główny przepływ zdarzeń	<p>Organizacja pomocowa dokonuje próby oceny zadania</p> <p>System weryfikuje uprawnienia sesji użytkownika</p> <p>System weryfikuje istnienie zadania</p> <p>System weryfikuje status zadania</p> <p>Organizacja wpisuje ocenę</p> <p>System zapisuje zmiany w zadaniu</p> <p>System aktualizuje repozytorium</p>
Alternatywny przepływ zdarzeń	<p>2a. Sesja użytkownika nie ma wystarczających uprawnień</p> <p>3a. Zadanie nie istnieje</p> <p>4a. Zadanie nie jest skończone</p> <p>4b. Zadanie jest ocenione</p>
Specjalne wymagania	<p>Kontrola dostępu w zależności od roli</p> <p>Wysoka niezawodność systemu</p> <p>Krótki czas odpowiedzi</p>

Część I: Dokumentacja techniczna

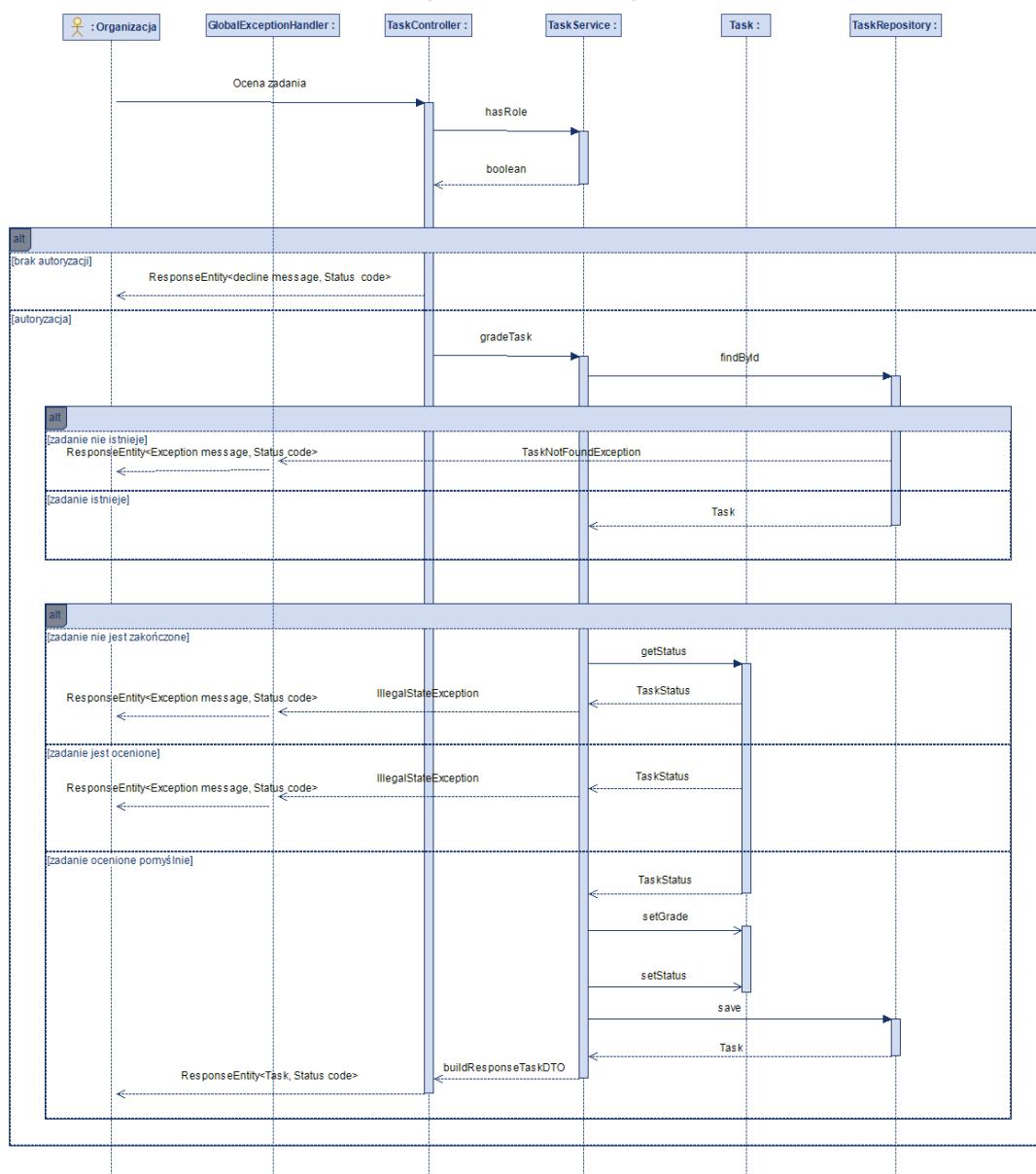


Diagram sekwencji przedstawiający przepływ komunikatów między aktorem "Organizacją", a klasami TaskController, TaskService, TaskRepository, Task i GlobalExceptionHandler podczas realizacji procesu oceny zadania.

2.10.4.4 Diagramy czynności

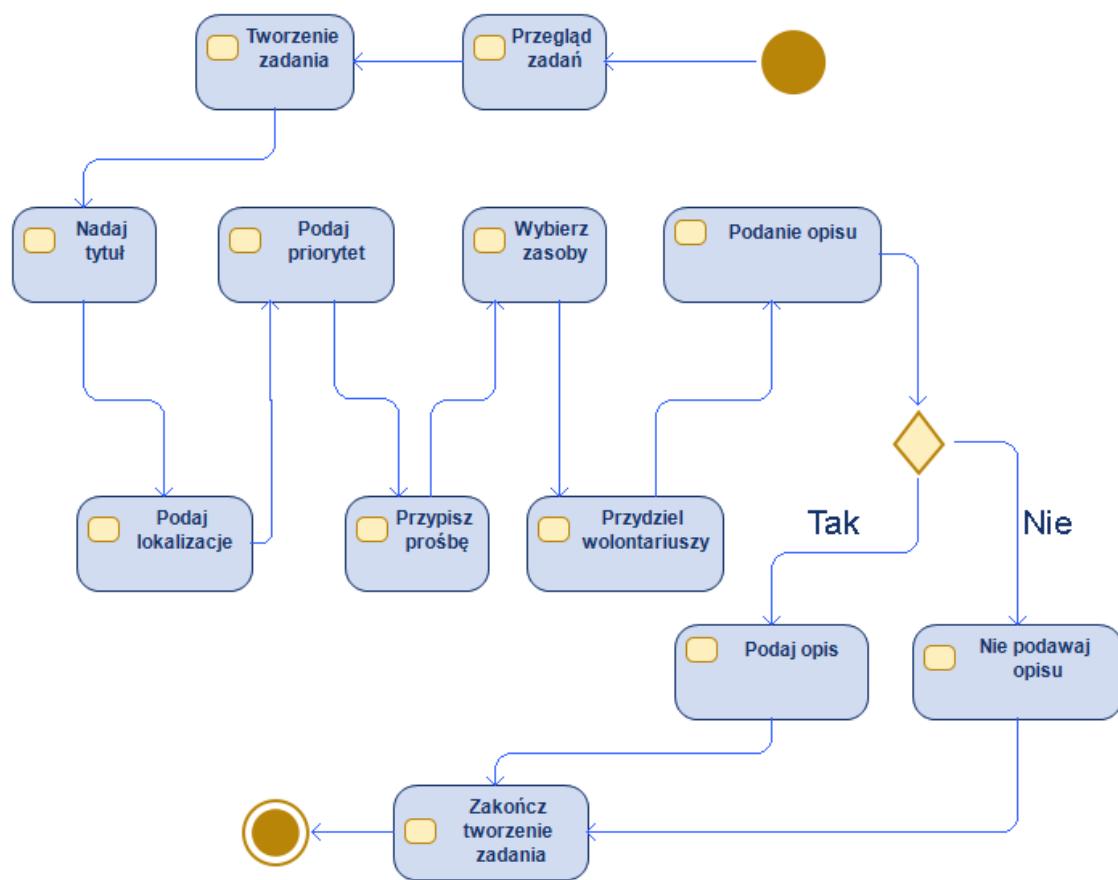


Diagram czynności przedstawia proces tworzenia nowego zadania w module Task. Obrazuje kolejność kroków i decyzji, które podejmuje użytkownik.

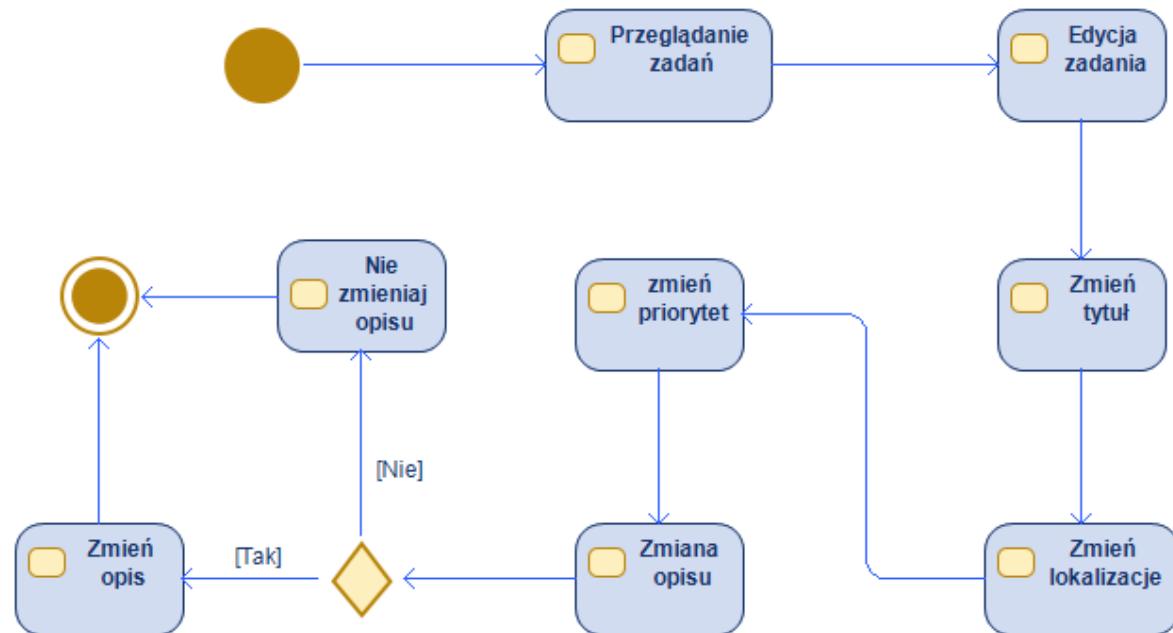


Diagram czynności przedstawia proces edycji zadania w module Task. Obrazuje kolejność kroków i decyzji, które podejmuje użytkownik.

2.10.4.5 Diagram maszyny stanowej



Powyższy diagram przedstawia maszynę stanową opisującą przepływ pracy związany z zarządzaniem zadaniami w systemie. Ilustruje stany, w jakich może znajdować się aplikacja w trakcie obsługi zadań, oraz przejścia między tymi stanami.

Część I: Dokumentacja techniczna

2.10.4.6 Diagram komponentów

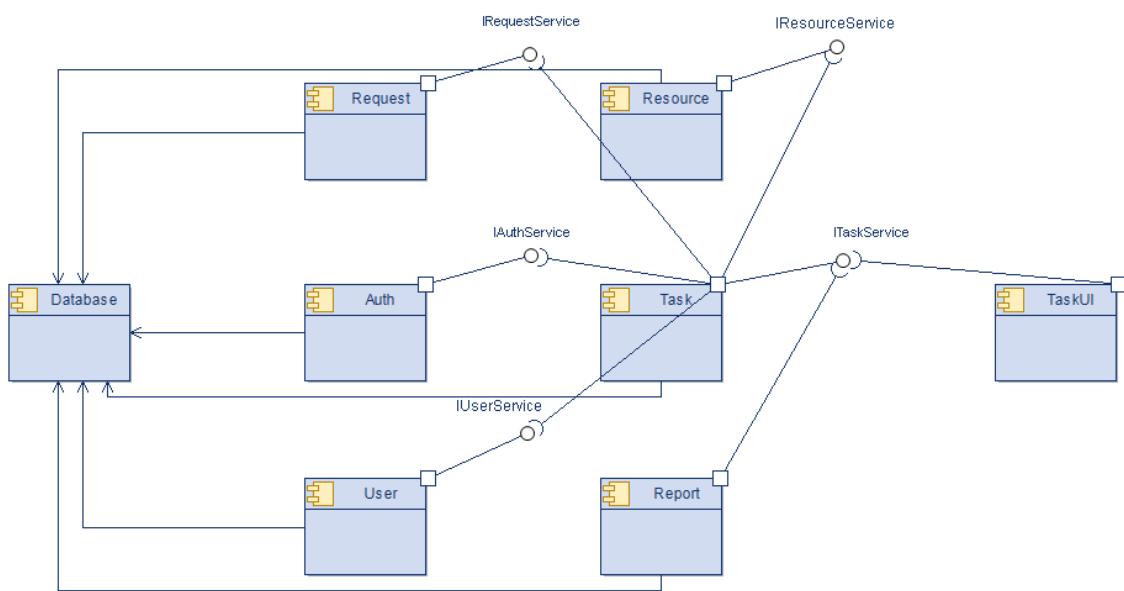


Diagram przedstawia architekturę komponentów systemu, obrazując ich rolę oraz powiązania w ramach systemu koordynacji pomocy humanitarnej. Komponent Task łączy się z interfejsami usługowymi Request, Resource, AuthService, UserService oraz TaskService przekazując im oraz czerpiąc od nich metody niezbędne do kooperacji całego systemu. Komponent Report łączy się z Task poprzez zaimplementowany interfejs ITaskservice. Komponenty połączone są również z bazą danych, dzięki czemu mogą zapisywać i odczytywać dane, utrwalając je.

2.10.4.7 Diagram pakietów

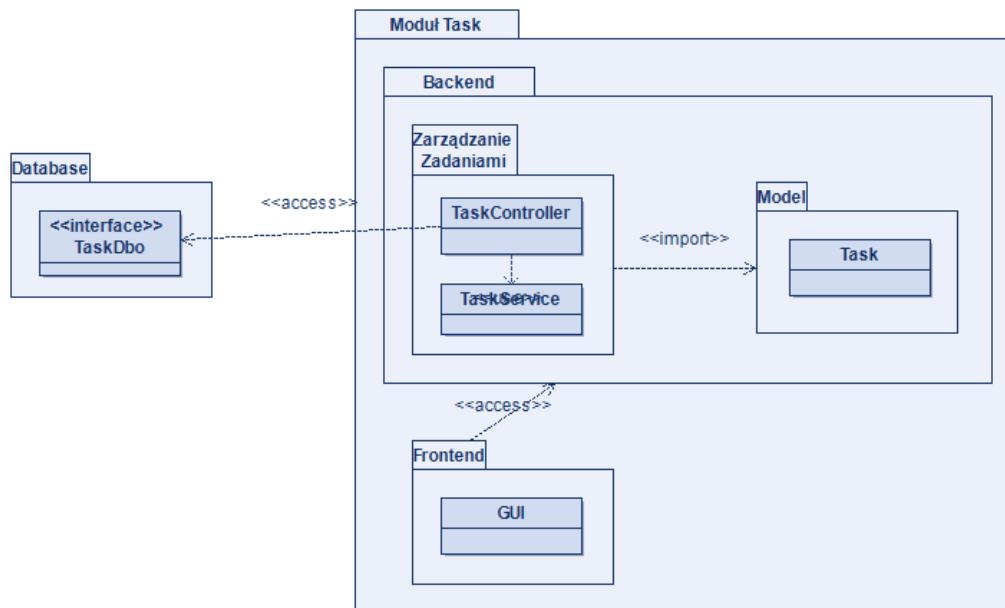


Diagram pakietów przedstawia strukturę modułu Task, rozdzielając go na warstwy funkcjonalne: Frontend, Backend oraz Model. Uwzględnia także komunikację z bazą danych.

Część I: Dokumentacja techniczna

2.10.4.8 Diagram przeglądu interakcji

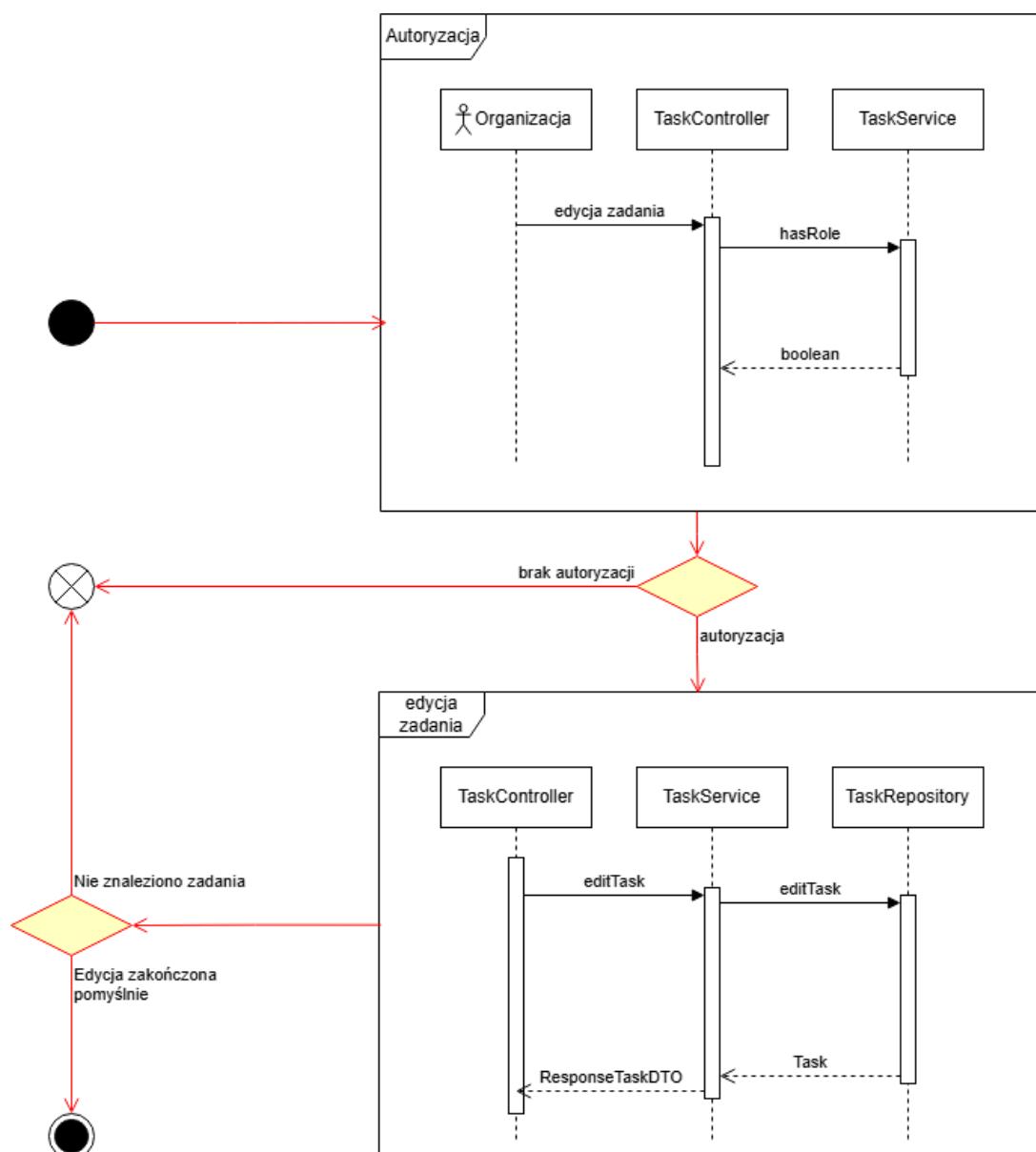


Diagram przeglądu interakcji przedstawia proces edycji zadania w module **Task**. Ilustruje wymianę komunikatów między aktorem (Organizacja), kontrolerem TaskController, TaskService oraz repozytorium TaskRepository, uwzględniając warunek autoryzacji i ewentualne błędy.

Część I: Dokumentacja techniczna

2.10.4.9 Diagram Strukturalny

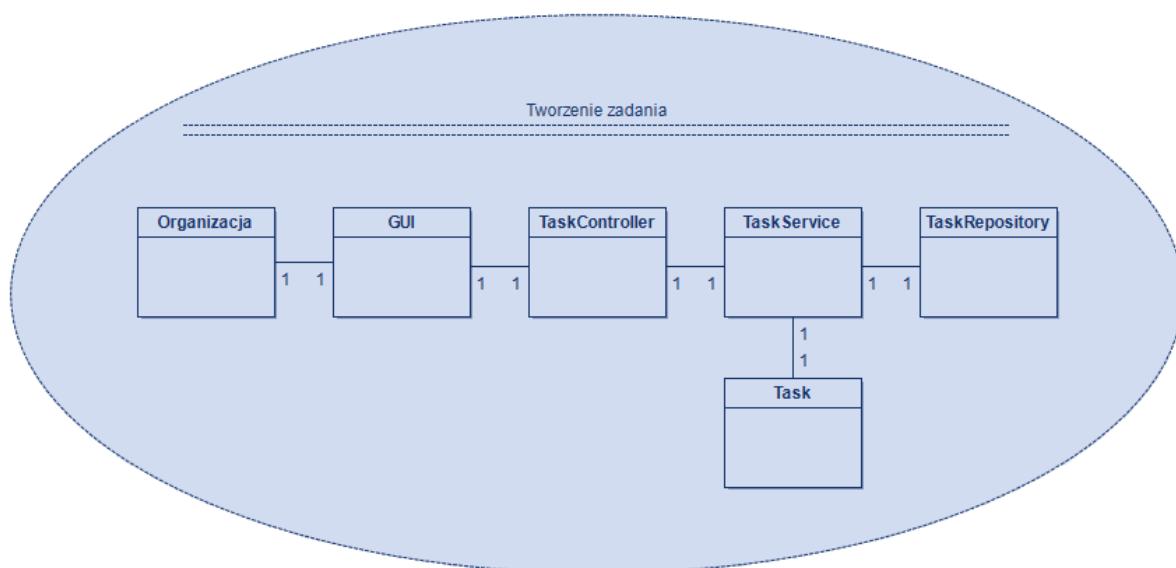


Diagram przedstawia strukturę komponentów zaangażowanych w proces tworzenia zadania oraz relacje między nimi. Organizacja inicjuje proces, GUI umożliwia użytkownikowi wprowadzanie danych, a TaskController koordynuje przepływ informacji między GUI a TaskService, który obsługuje logikę biznesową. Dane zadania trafiają do TaskRepository, odpowiedzialnego za zapis w bazie danych, a model Task przechowuje szczegóły zadania.

2.10.4.10 Diagram harmonogramowania

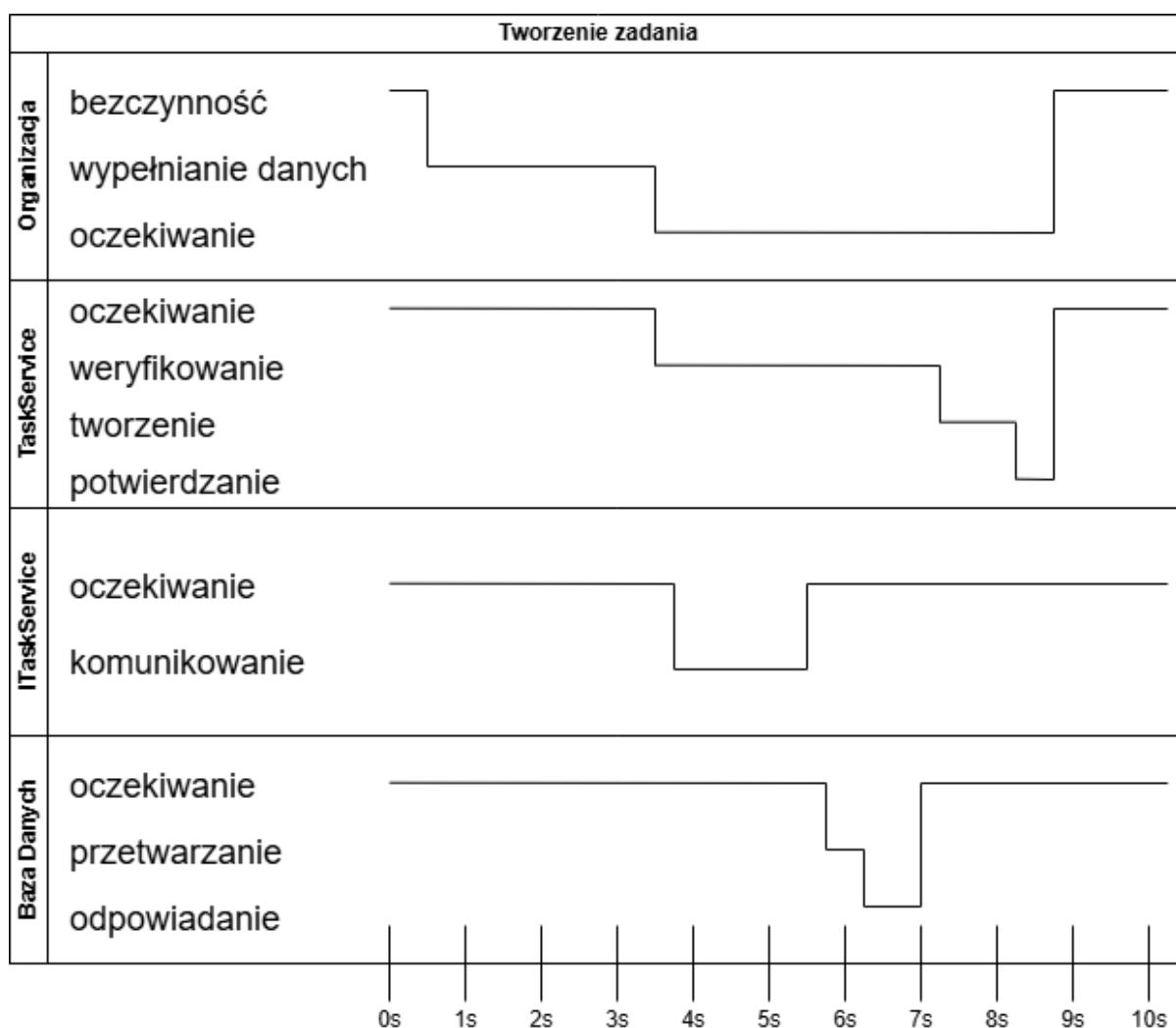


Diagram przedstawia proces tworzenia zadania w systemie, ukazując przepływ informacji między organizacją i komponentami TaskService, interfejsem ITaskService i Bazą Danych. Proces rozpoczyna się od wprowadzenia danych przez organizację, po czym TaskService obsługuje logikę biznesową i komunikuje się z innymi warstwami za pomocą ITaskService. Na końcu dane zadania są zapisywane w Bazie Danych. przepływ informacji między różnymi komponentami systemu. Proces został przedstawiony w układzie pionowym, gdzie każda warstwa reprezentuje kolejny etap realizacji funkcji.

2.11 Moduł User

2.11.1 Cel

Celem modułu jest zarządzanie informacjami o użytkownikach systemu, takimi jak poszkodowani, organizacje pomocowe, wolontariusze, darczyńcy i przedstawiciele władz.

2.11.2 Założenia

Umożliwienie osobom zarejestrowanym w systemie wypełnienie informacji o profilu.

Udostępnianie informacji o użytkownikach zarejestrowanych w systemie innym modułom.

Umożliwienie osobom zarejestrowanym w systemie dokonania edycji profilu.

Umożliwienie organizacjom pomocowym wyświetlenia listy wolontariuszy.

Umożliwienie wolontariuszom i przedstawicielom władz wyświetlenia listy organizacji pomocowych.

Umożliwienie wolontariuszom aplikowania do organizacji pomocowej.

2.11.3 Wymagania

1. Funkcjonalne:

- Umożliwienie użytkownikom edycji profilu.
 - Każdy użytkownik ma możliwość edycji swoich danych osobowych, podanych podczas wypełniania profilu, w zależności od tego jaka rola została wybrana.
 - Edytować można tylko swój profil.
- Zarządzanie informacjami o użytkownikach zarejestrowanych w systemie.
 - Informacje o użytkownikach są pobierane z bazy danych.
- Lista użytkowników zarejestrowanych.
 - Wolontariusz może zobaczyć listę dostępnych organizacji pomocowych.
 - Organizacja pomocowa może zobaczyć listę wolontariuszy.
 - Przedstawiciel władz może zobaczyć listę dostępnych organizacji pomocowych.
- Wypełnienie informacji o profilu użytkownika.
 - Formularz do wypełnienia danych osobowych dostępny w profilu użytkownika (wolontariusz, poszkodowany, darczyńca, przedstawiciel władz).
 - Darczyńca, wolontariusz, przedstawiciel władzy i poszkodowany podaje imię, nazwisko oraz numer PESEL.
 - Formularz do wypełnienia informacji o organizacji pomocowej.
 - Organizacja pomocowa podaje nazwę organizacji.
- Aplikowanie wolontariuszy do organizacji pomocowych.
 - Wolontariusz może aplikować do wybranej organizacji pomocowej.
 - Wolontariusz może usunąć swoją aplikację do organizacji pomocowej.
- Zarządzanie wolontariuszami przez organizacje pomocowe.
 - Zaakceptowanie aplikacji wolontariusza.
 - Odrzucenie aplikacji wolontariusza.
 - Usunięcie wolontariusza z organizacji pomocowej.

Część I: Dokumentacja techniczna

- Ocena profilu wolontariusza
 - Wyświetlenie średniej ocen z zrealizowanych zadań.
- Zarządzanie interfejsem
 - Wyświetlenie funkcjonalności dostępnych dla danej roli.

2. Niefunkcjonalne:

- Wydajność
 - Reakcja na akcje użytkownika w czasie nie dłuższym niż 2 sekundy.
- Użyteczność
 - Intuicyjny i przyjazny interfejs użytkownika
- Niezawodność
 - Dostępność przez 99% czasu
- Kompatybilność
 - Wsparcie dla najnowszych wersji przeglądarek internetowych oraz dostęp na urządzeniach mobilnych (przez przeglądarkę).

2.11.4 Diagramy

2.11.4.1 Diagram przypadków użycia dla wolontariusza

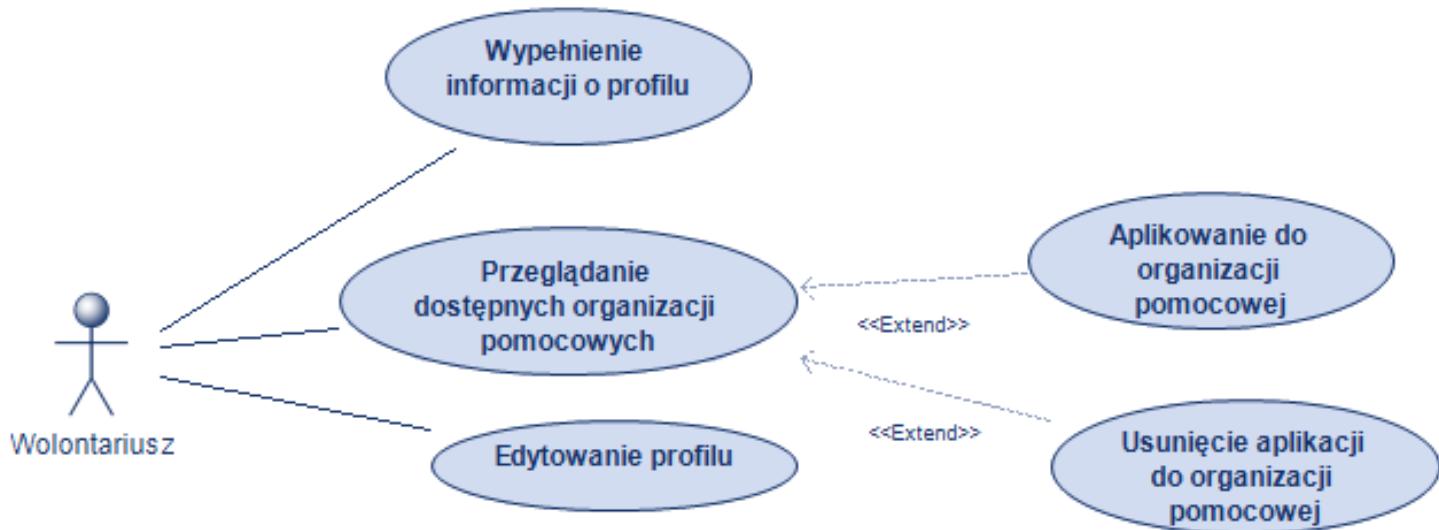


Diagram przypadków użycia przedstawia funkcjonalności systemu dostępne dla wolontariusza. Wolontariusz po zalogowaniu może kolejno:

Wypełnić informacje o profilu

Przeglądać dostępne organizacje pomocowe – ten przypadek użycia jest rozszerzony o aplikowanie do organizacji pomocowych i usuwanie aplikacji do organizacji pomocowych

Edytowanie profilu

Część I: Dokumentacja techniczna

2.11.4.2 Diagram przypadków użycia dla organizacji pomocowej

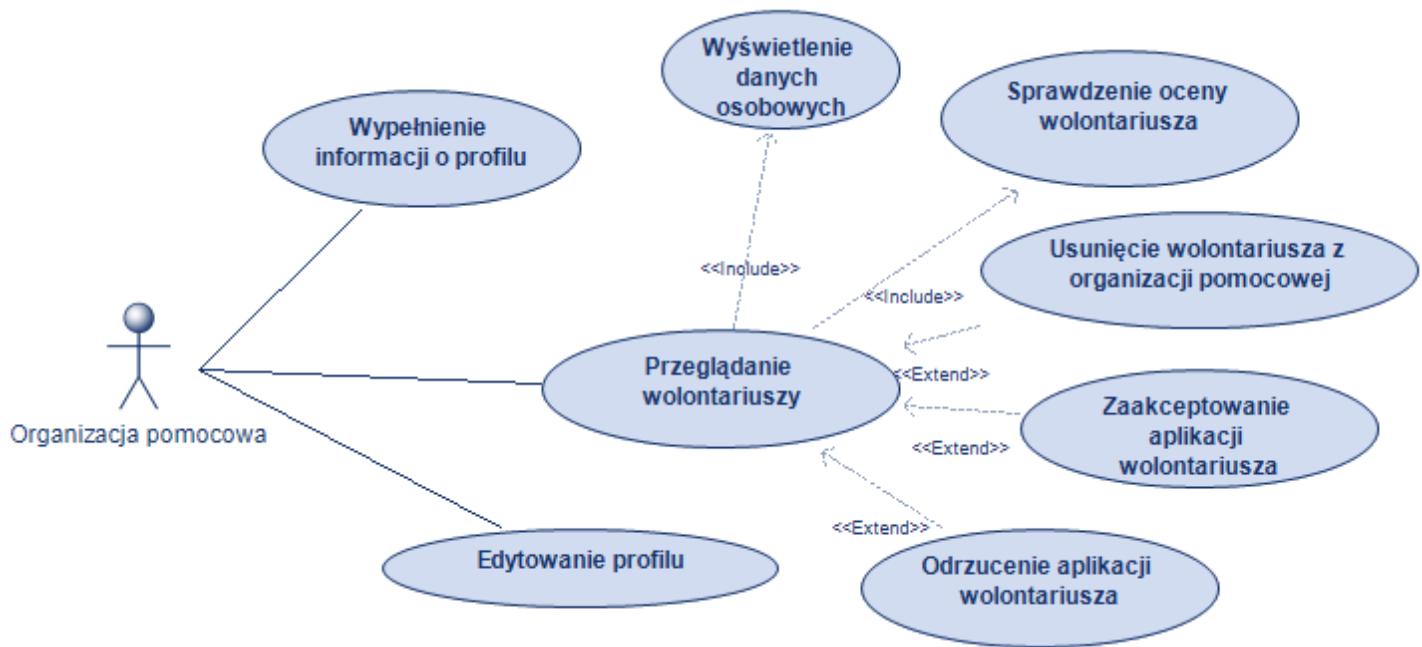


Diagram przypadków użycia przedstawia funkcjonalności systemu dostępne dla organizacji pomocowej. Organizacja pomocowa po zalogowaniu może kolejno:

Wypełnić informacje o profilu

Przeglądać wolontariuszy – ten przypadek użycia jest rozszerzony o zaakceptowanie lub odrzucenie aplikacji wolontariusza oraz o usunięcie wolontariusza z organizacji pomocowej.
W tym przypadku użycia zawarte jest wyświetlenie danych osobowych wolontariuszy oraz sprawdzenie oceny wolontariusza.

Edytowanie profilu

Część I: Dokumentacja techniczna

2.11.4.3 Diagram przypadków użycia dla przedstawiciela władz

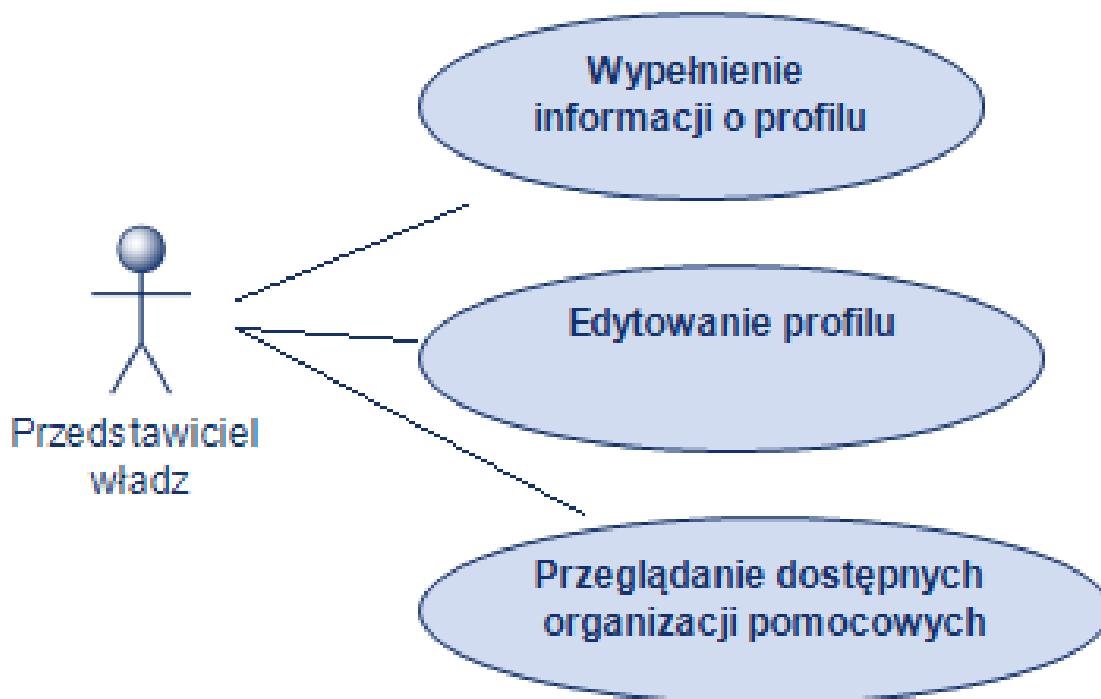


Diagram przypadków użycia przedstawia funkcjonalności systemu dostępne dla przedstawiciela władz. Przedstawiciel władz po zalogowaniu może kolejno:

Wypełnić informacje o profilu

Przeglądać dostępne organizacje pomocowe

Edytowanie profilu

2.11.4.4 Diagram przypadków użycia dla darczyńcy oraz dla poszkodowanego

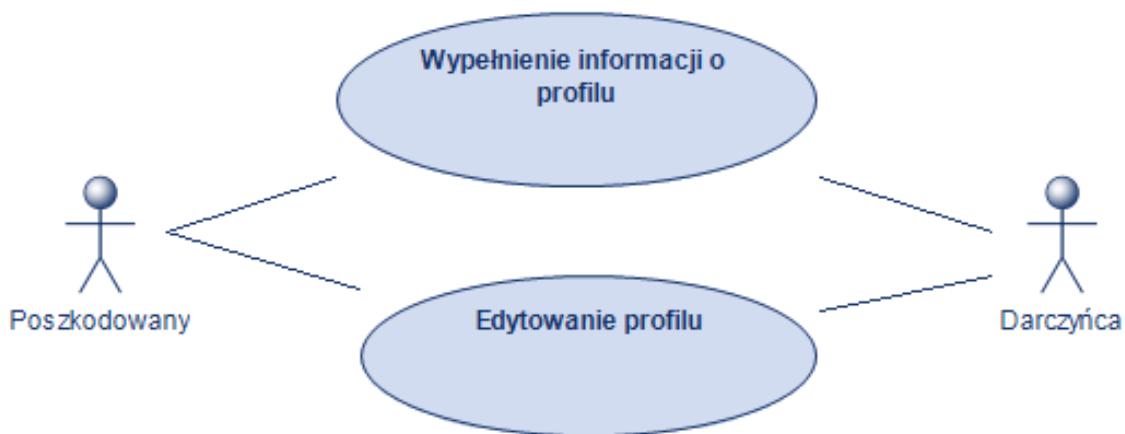


Diagram przypadków użycia przedstawia funkcjonalności systemu dostępne dla poszkodowanego oraz darczyńcy. Poszkodowany oraz darczyńca po zalogowaniu może kolejno:

Część I: Dokumentacja techniczna

Wypełnić informacje o profilu

Edytowanie profilu

2.11.4.5 Diagram klas

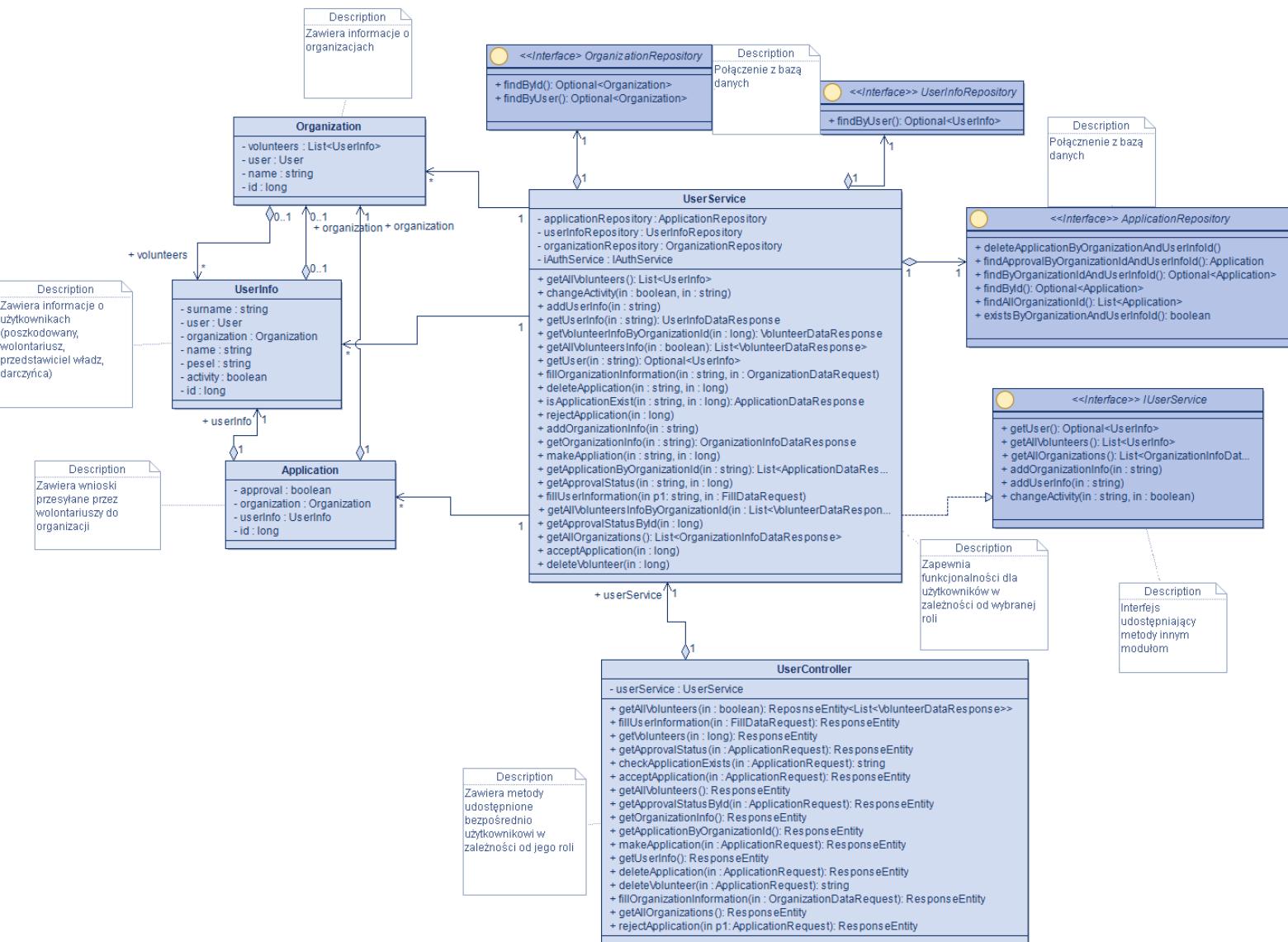


Diagram klas przedstawia strukturę modułu User w postaci klas, ich atrybutów oraz relacji. Głównym elementem modułu jest klasa **UserService**, która implementuje metody udostępniane innym modułom za pomocą interfejsu **IUserService**, a także metody niezbędne do zarządzania danymi personalnymi użytkowników zarejestrowanych w systemie. Ważnym elementem jest powyższego diagramu klas jest **UserController**, który udostępnia endpointy dla interfejsu graficznego. Z kolei klasy **Organization**, **UserInfo**, **Application** są odzwierciedleniem tabel zawartych w bazie danych. **OrganizationRepository**, **UserInfoRepository**, **ApplicationRepository** udostępniają metody, które umożliwiają połączenie z bazą danych.

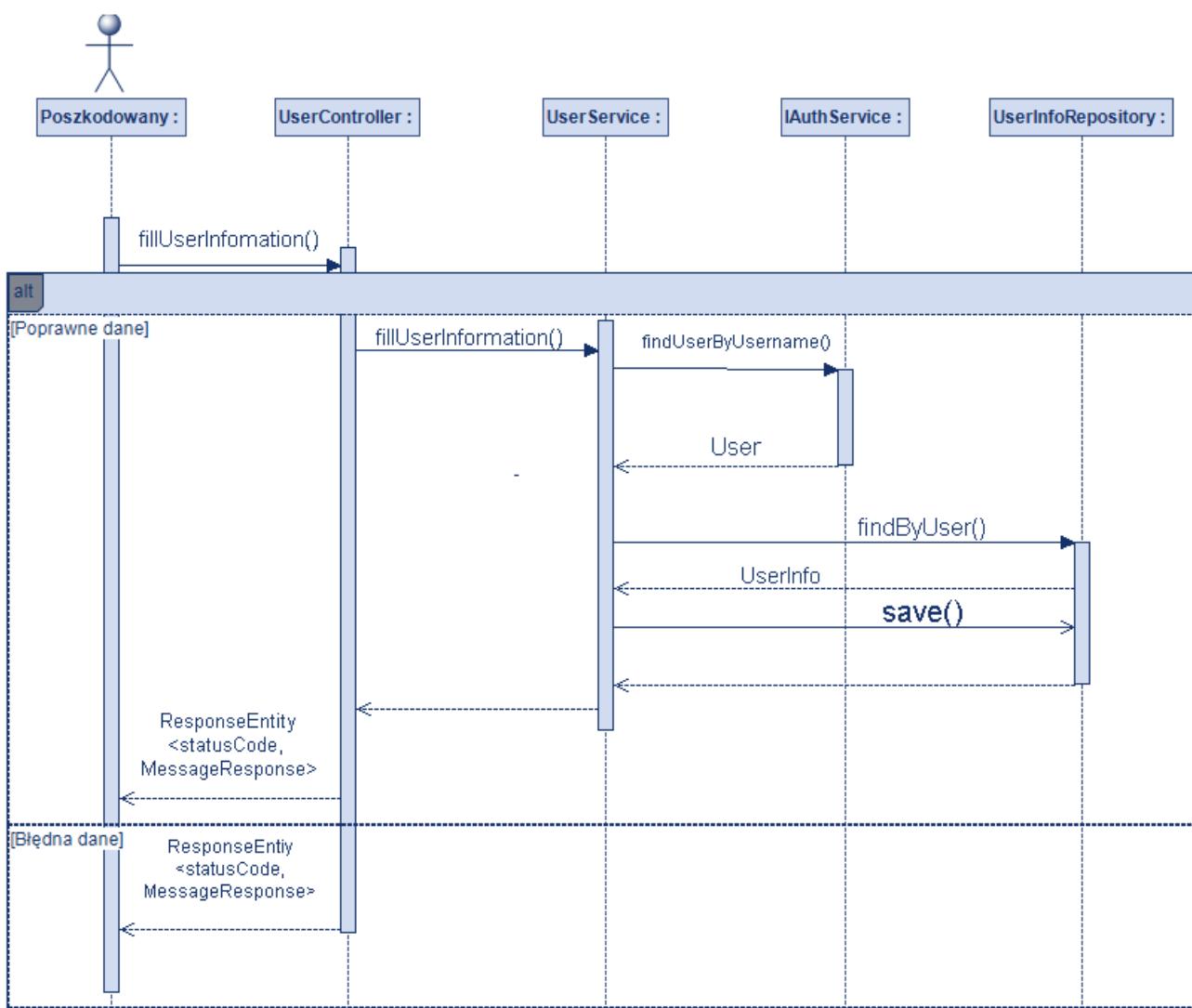
Część I: Dokumentacja techniczna

2.11.4.6 Scenariusz – wypełnienie danych o profilu

Nazwa	Wypełnienie danych o profilu
Poziom ważności	Wysoki
Typ przypadków użycia	Niezbędny
Aktorzy	Poszkodowany, Wolontariusz, Darczyńca, Przedstawiciel władz
Krótki opis	Użytkownik przy pomocy formularza dodaje dane osobowe do profilu
Warunki wstępne	Konto musi mieć przypisaną jedną z ról – ROLE_VOLUNTEER, ROLE_VICTIM, ROLE_DONOR, ROLE_AUTHORITY
Warunki końcowe	Profil użytkownika jest uzupełniany nowymi danymi, które są zapisywane w bazie danych
Główny przepływ zdarzeń	<ol style="list-style-type: none"> 1. Poszkodowany podaje informacje o swoim imieniu, nazwisku oraz numerze PESEL. 2. Informacje zostają wysłane do UserService. 3. System pobiera dane konta na postawie username. 4. System tworzy i dodaje dane personalne do użytkownika. 5. Nowy użytkownik jest dodawany do bazy danych. 6. System wyświetla komunikat potwierdzający wykonanie zadania.
Alternatywne przepływy zdarzeń	<ol style="list-style-type: none"> 1a. Poszkodowany podał błędne dane 6a. System wyświetla komunikat o błędzie wykonywanej operacji
Specjalne wymagania	Połączenie z bazą danych

Część I: Dokumentacja techniczna

2.11.4.7 Diagram sekwencji – wypełnienie danych o profilu



Część I: Dokumentacja techniczna

Diagram przedstawia zarys komunikacji w trakcie realizacji przypadku użycia „Wypełnij informacje o profilu”. Użytkownik, który posiada role wolontariusz, darczyńca, poszkodowany lub przedstawiciel władzy zgłasza żądanie wypełnienia danymi personalnymi. Wywoływana jest metoda `fillUserInformation` z `UserController`, następnie wywoływana jest ta metoda o tej samej nazwie z `UserService`. W metodzie `fillUserInformation` z `UserService` znajdowany jest obiekt `User` użytkownika, który zgłosił żądanie za pomocą metody `findUserByUsername` z interfejsu `IAuthService`. Następnie znajdowany jest powiązany z tym obiektem obiekt `UserInfo` za pomocą metody `findByUser` z `UserInfoRepository`. Kolejnym etapem jest zmiana danych i zapisanie zmian za pomocą `save` z `UserInfoRepository`. Następnym etapem jest zwroćenie komunikatu i wyświetlenie go na ekranie. W przypadku powodzenia zwracany jest komunikat o poprawnym wypełnieniu danych. W przeciwnym wypadku zwracany jest komunikat o błędzie.

2.11.4.8 Scenariusz – aplikowanie do organizacji pomocowej

Nazwa	Aplikowanie do organizacji pomocowej
Poziom ważności	Średni
Typ przypadków użycia	Istotny
Aktorzy	Wolontariusz
Krótki opis	Wolontariusz aplikuje do wybranej z listy organizacji pomocowej
Warunki wstępne	Konto musi mieć przypisaną rolę <code>ROLE_VOLUNTEER</code>
Warunki końcowe	Wysłanie aplikacji do organizacji pomocowej
Główny przepływ zdarzeń	<ol style="list-style-type: none"> Wolontariusz wybiera organizację pomocową do której chce aplikować. Wolontariusz potwierdza wysłanie aplikacji. Informacje zostają wysłane do organizacji. System wyświetla komunikat potwierdzający wykonanie operacji.
Alternatywne przepływy zdarzeń	<ol style="list-style-type: none"> Błąd przesyłania informacji do organizacji. System wyświetla komunikat o błędzie wykonywanej operacji.
Specjalne wymagania	Połączenie z bazą danych

Część I: Dokumentacja techniczna

2.11.4.9 Diagram sekwencji – aplikowanie do organizacji pomocowej

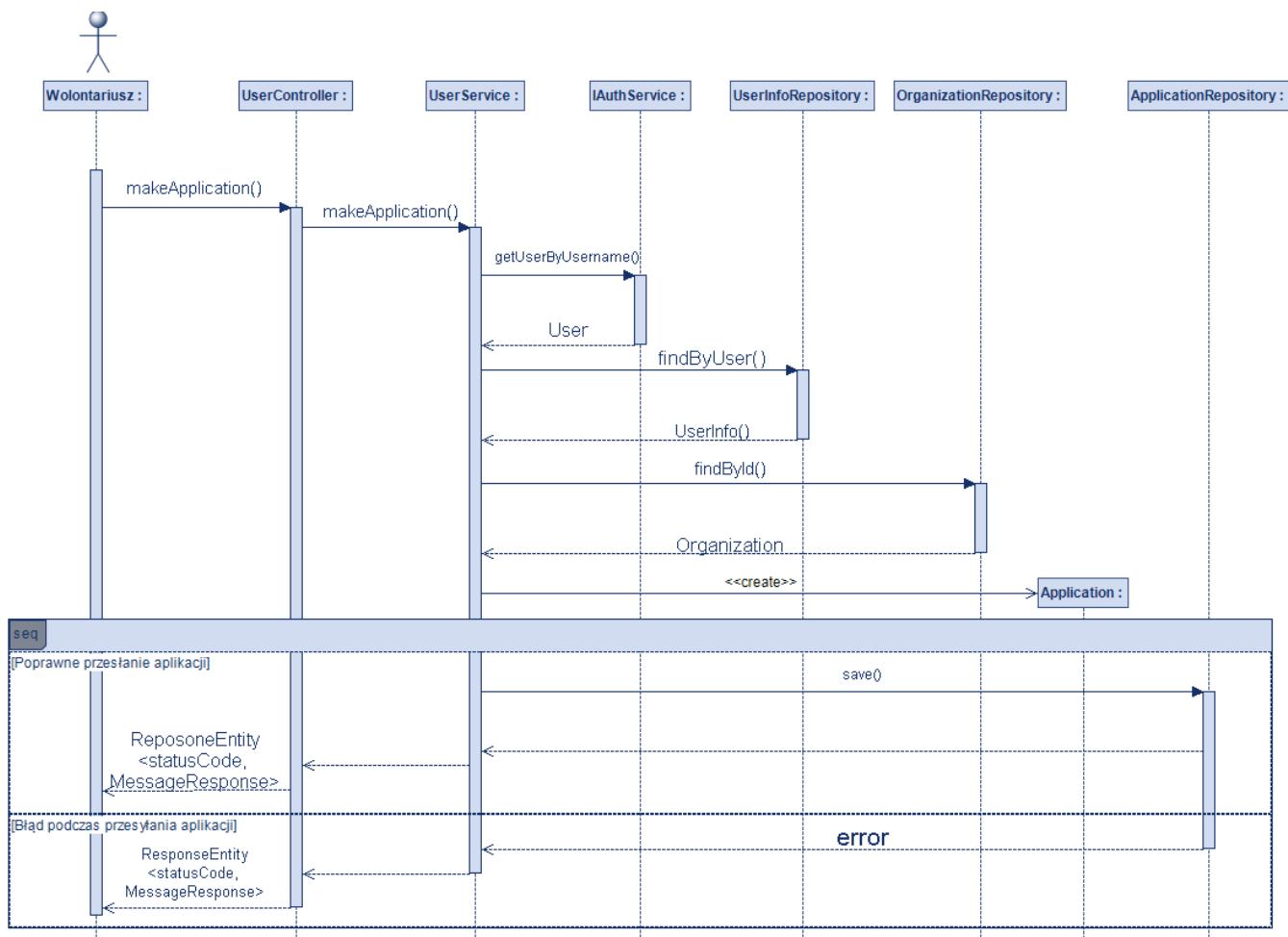


Diagram przedstawia zarys komunikacji w trakcie realizacji przypadku użycia „Aplikowanie do organizacji pomocowej”. Użytkownik, który posiada role wolontariusz zgłasza żądanie stworzenia aplikacji do organizacji pomocowej. Na początku wywoływana jest metoda makeApplication z UserController, następnie wywoływana jest ta metoda o tej samej nazwie z UserService, w której znajdowany jest obiekt User użytkownika, który zgłosił żądanie za pomocą metody findUserByUsername z interfejsu IAuthService. Następnie znajdowany jest powiązany z tym obiektem obiekt UserInfo za pomocą metody findByUser z UserInfoRepository. Następnie znaleziony jest obiekt Organization za pomocą findByUser z OrganizationRepository. Kolejnym etapem jest stworzenie obiektu Application, który posiada obiekt UserInfo oraz Organization i zapisanie go za pomocą save z ApplicationRepository. Następnym etapem jest zwrócenie komunikatu i wyświetlenie go na ekranie. W przypadku powodzenia zwracany jest komunikat o poprawnym wypełnieniu danych. W przeciwnym wypadku zwracany jest komunikat o błędzie.

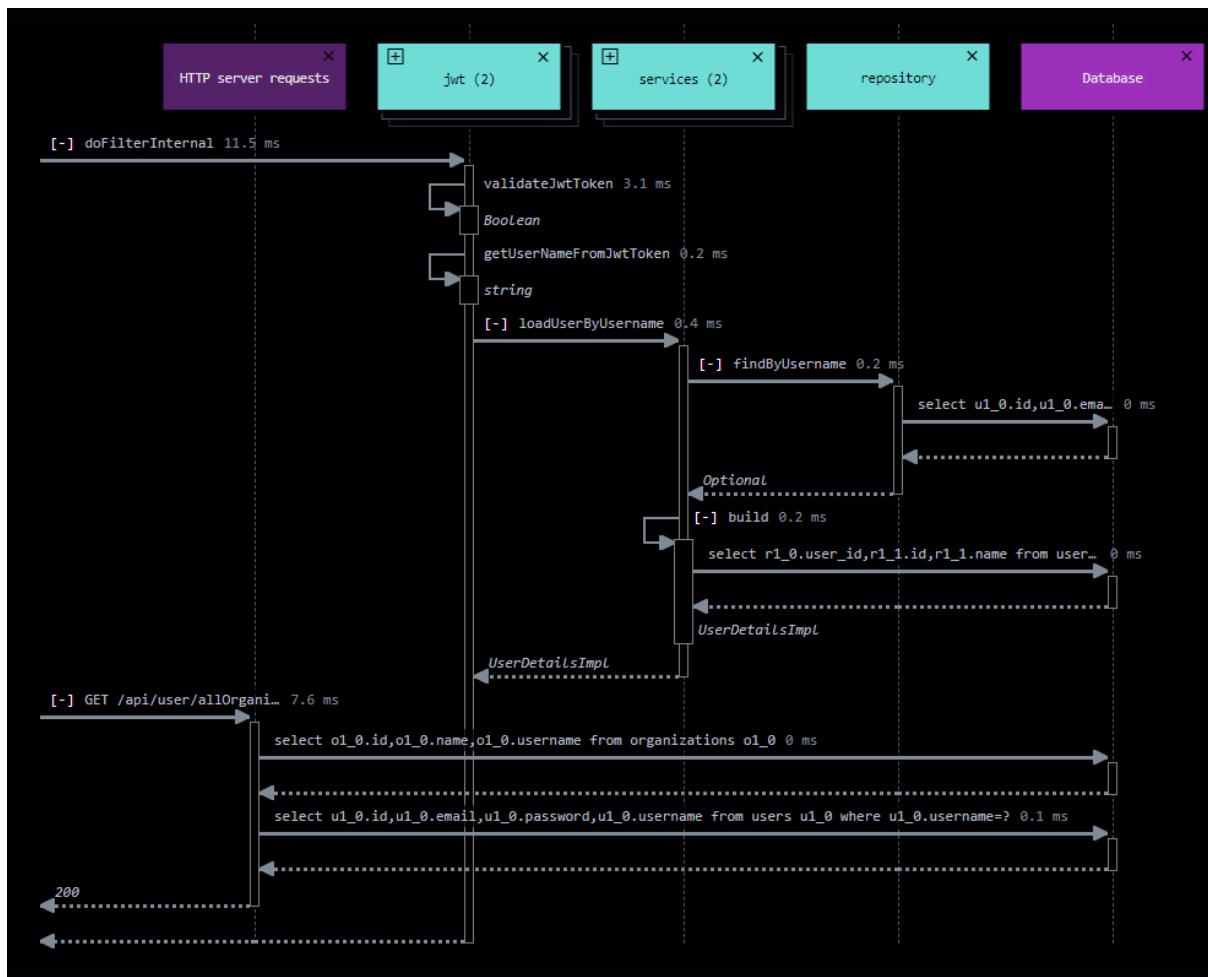
2.11.4.10 Scenariusz – Przeglądanie dostępnych organizacji pomocowych

Nazwa	Przeglądanie dostępnych organizacji pomocowych
Twórcy	Łukasz Łyczko, Wiktor Glaner
Poziom ważności	Wysoki
Typ przypadku użycia	Ogólny, istotny

Część I: Dokumentacja techniczna

Aktorzy	Wolontariusz
Krótki opis	Wyświetlanie listy organizacji
Warunki wstępne	Musi istnieć organizacja
Warunki końcowe	brak
Główny przepływ zdarzeń	<p>Wolontariusz próbuje wyświetlić listę organizacji</p> <p>System wykonuje bezpośrednie zapytanie do bazy danych do konkretnej tabeli (encja organizations)</p> <p>System zwraca odpowiedź wraz z widokiem</p>
Alternatywny przepływ zdarzeń	<p>2a. brak organizacji w systemie</p> <p>3a. sesja użytkownika nie ma wystarczających uprawnień</p>
Specjalne wymagania	<p>Wysoka niezawodność systemu</p> <p>Krótki czas odpowiedzi</p>

2.11.4.11 Diagram sekwencji – Przeglądanie dostępnych organizacji pomocowych



Część I: Dokumentacja techniczna

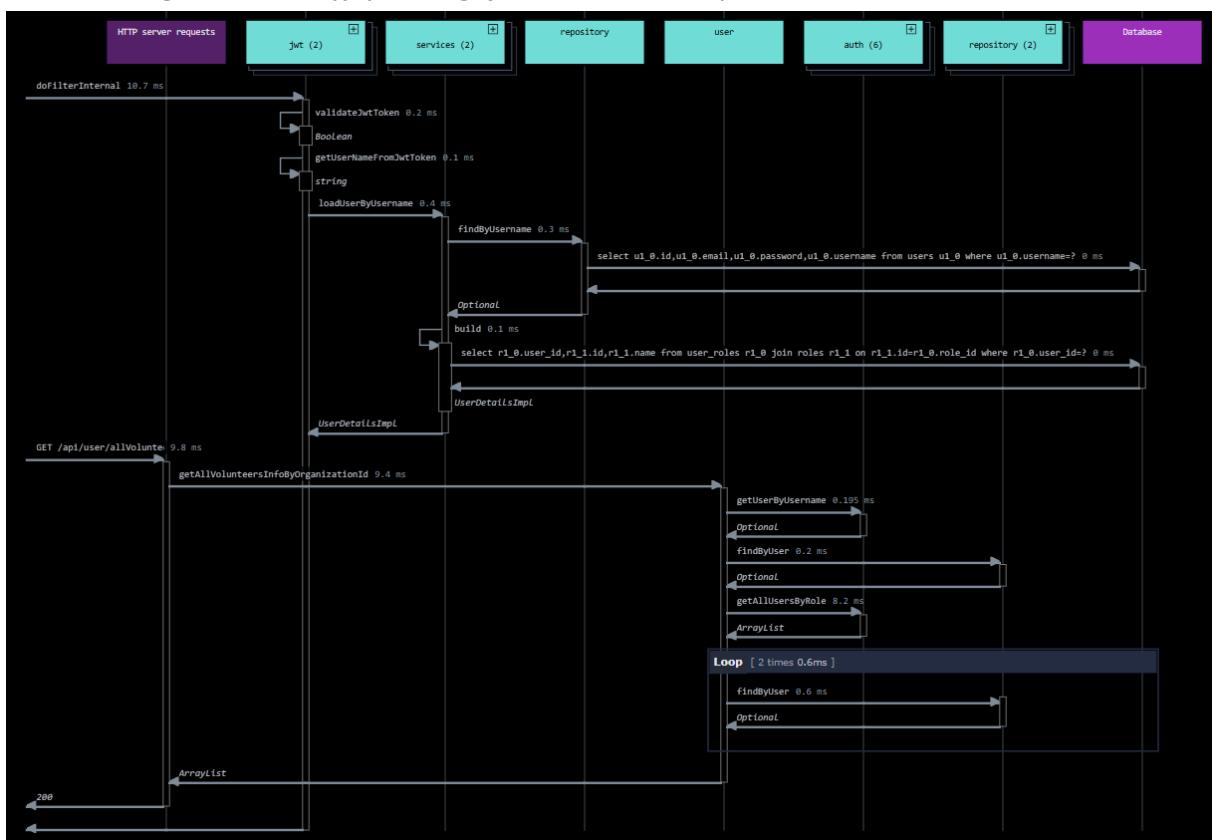
2.11.4.12 Scenariusz – Przeglądanie wolontariuszy

Nazwa	Przeglądanie wolontariuszy
Twórcy	Łukasz Łyczko, Wiktor Glaner
Poziom ważności	Wysoki
Typ przypadku użycia	Ogólny, istotny
Aktorzy	Organizacja
Krótki opis	Wyświetlanie listy wolontariuszy danej organizacji
Warunki wstępne	Musi istnieć wolontariuszy w organizacji
Warunki końcowe	brak
Główny przepływ zdarzeń	<p>Organizacja próbuje wyświetlić listę wolontariuszy</p> <p>System wywołuje metodę getAllVolunteers z zapytaniem GET</p> <p>System weryfikuje uprawnienia sesji użytkownika</p> <p>System zwraca odpowiedź wraz z widokiem</p>
Alternatywny przepływ zdarzeń	<p>2a. brak wolontariuszy w systemie</p> <p>3a. sesja użytkownika nie ma wystarczających uprawnień</p>
Specjalne wymagania	<p>Wysoka niezawodność systemu</p> <p>Krótki czas odpowiedzi</p>

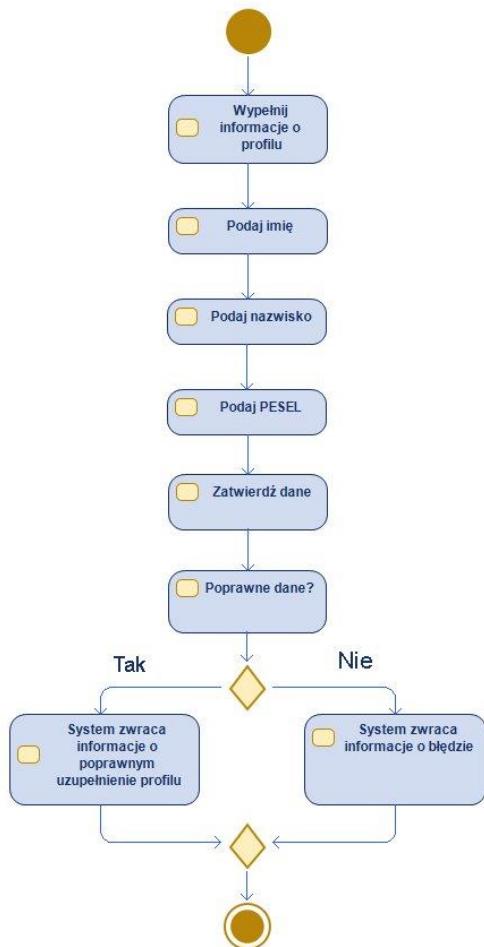
SYSTEM KOORDYNACJI POMOCY HUMANITARNEJ

Część I: Dokumentacja techniczna

2.11.4.13 Diagram sekwencyjny – Przeglądanie wolontariuszy

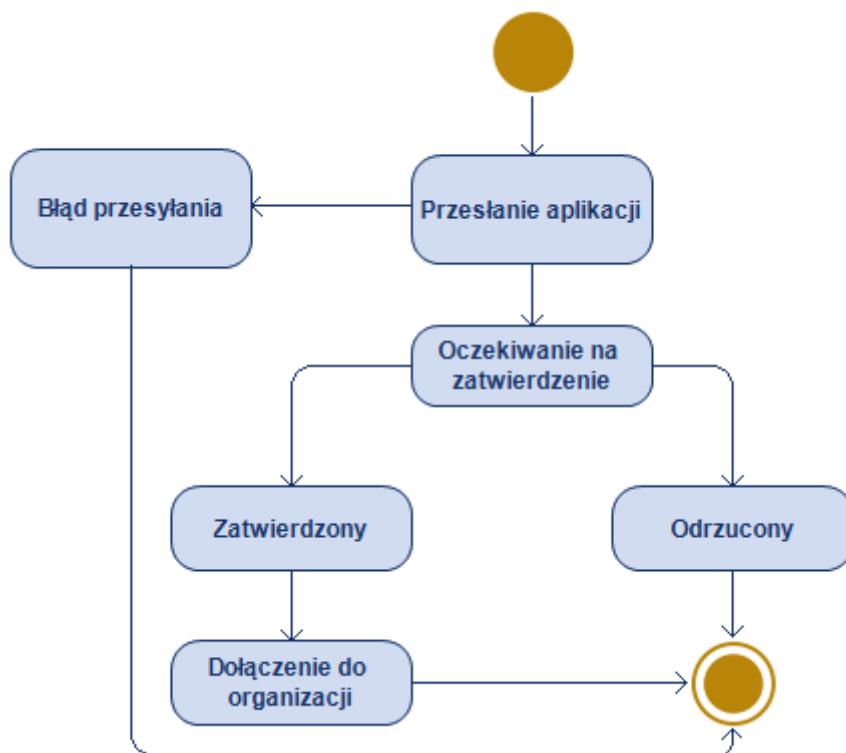


2.11.4.14 Diagram czynności



Powyższy diagram przedstawia czynności wykonywane podczas realizacji przypadku użycia „Wypełnij informacje o profilu”. Użytkownik wypełnia kolejno pola na imię, nazwisko, pesel, a następnie zatwierdza dane. Przebieg przewiduje dwa przypadki błędu – gdy pesel nie składa się z jedenastu cyfr oraz gdy któryś z pól zostało pozostawione puste. W takim wypadku, system wyświetli odpowiedni komunikat błędu i nie dopuści do wypełnienia danych, a sterowanie zostanie zwrócone do użytkownika. W przeciwnym przypadku system zwróci informacje o poprawnym uzupełnieniu profilu.

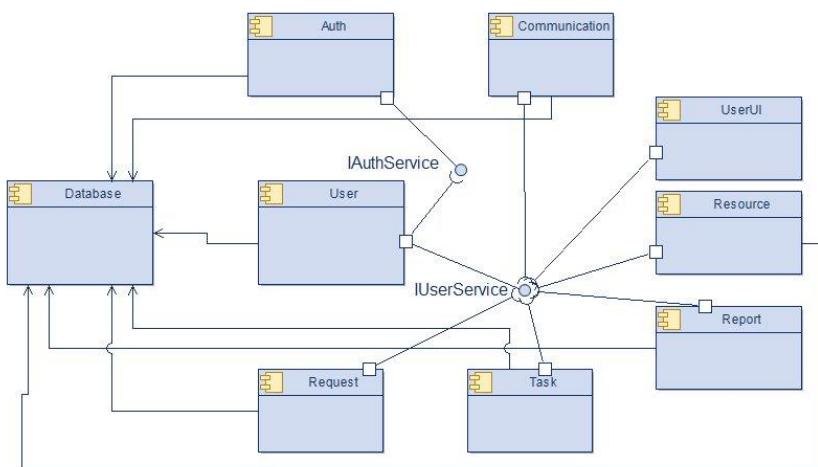
2.11.4.15 Diagram maszyny stanowej



Powyższy diagram przedstawia możliwe stany wniosku o przyjęcie do organizacji pomocy humanitarnej. Stanem początkowym jest „Przesłanie aplikacji”, oznaczające wysłanie do bazy informacji o stworzeniu nowego wniosku. Zadanie w tym stanie może przejść albo do stanu „Oczekiwanie na zatwierdzenie” w przypadku pomyślnego dodania informacji do bazy lub „Błąd przesyłania” w przypadku niepowodzenia. Jeżeli zdanu przeszło w stan „Oczekiwanie na zatwierdzenie” może zostać zatwierzone, co powoduje przejście w stan „Zatwierdzony”, a następnie przejście w stan „Dołączenie do organizacji” bądź odrzucone, co powoduje przejście w stan „Odrzucony”.

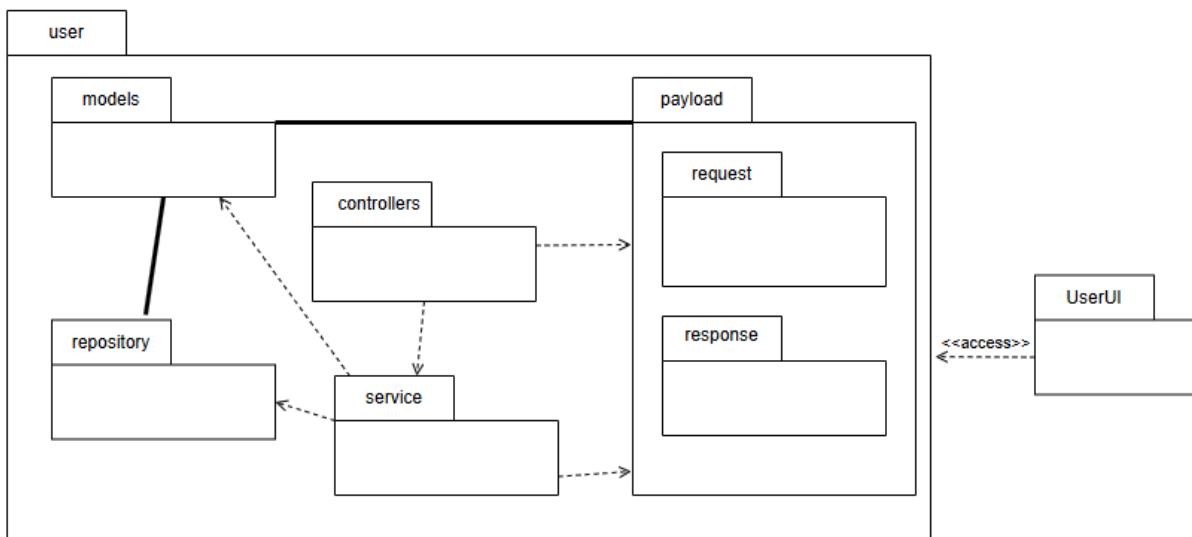
Część I: Dokumentacja techniczna

2.11.4.16 Diagram komponentów



Powyższy diagram przedstawia zależności komponentu User z innymi komponentami zawartymi w systemie. Komponent User jest odpowiedzialny za zarządzanie użytkownikami oraz obsługę aplikacji wolontariuszy do organizacji pomocowych. W tym module realizowany jest także komponent o nazwie UserUI, który odpowiada za warstwę prezentacji i komunikacji z użytkownikiem systemu.

2.11.4.17 Diagram pakietów

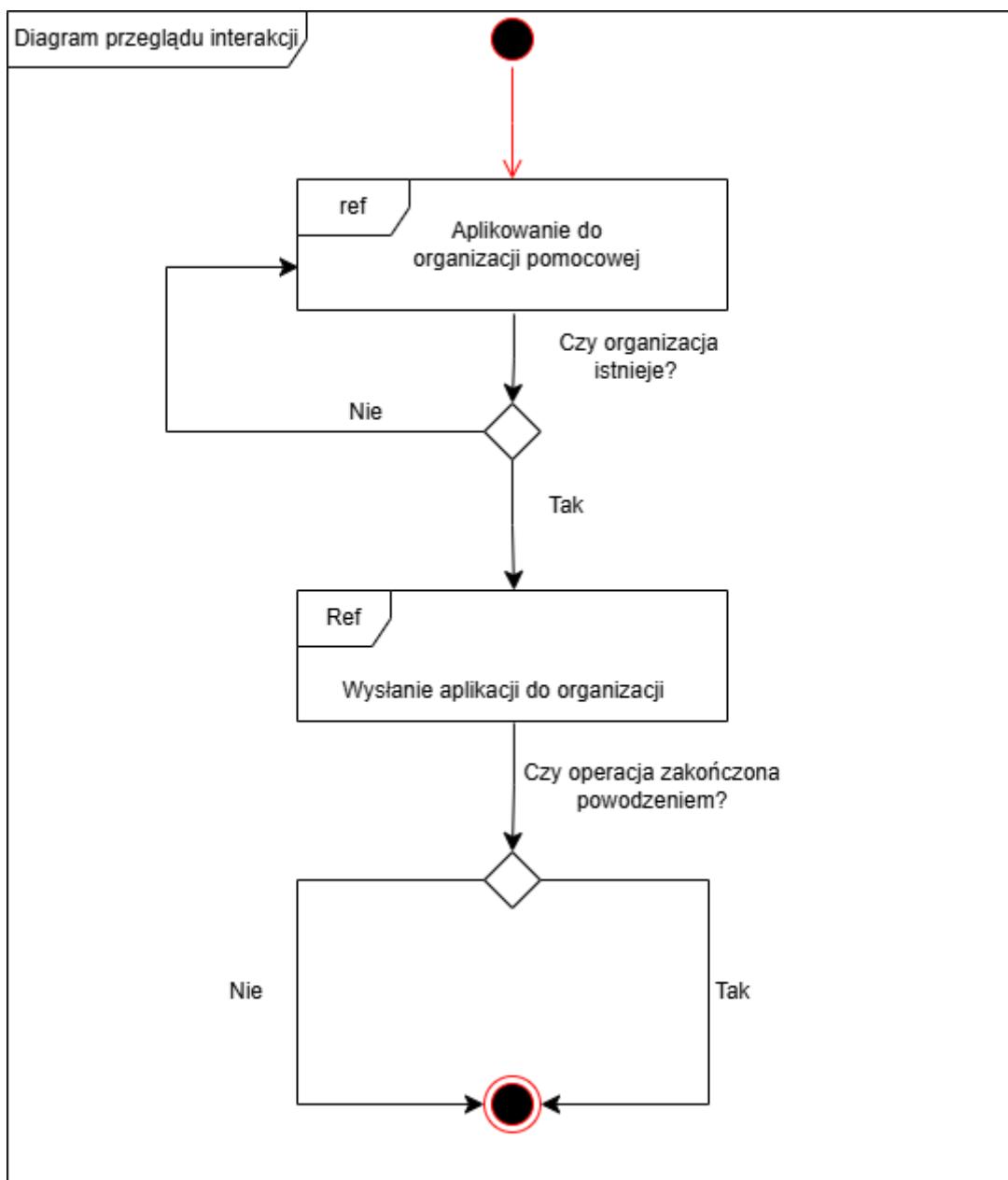


Powyższy diagram przedstawia strukturę pakietów wewnątrz modułu User oraz zależności między nimi. Każdy pakiet grupuje klasy odpowiedzialne za określoną funkcjonalność systemu.

Część I: Dokumentacja techniczna

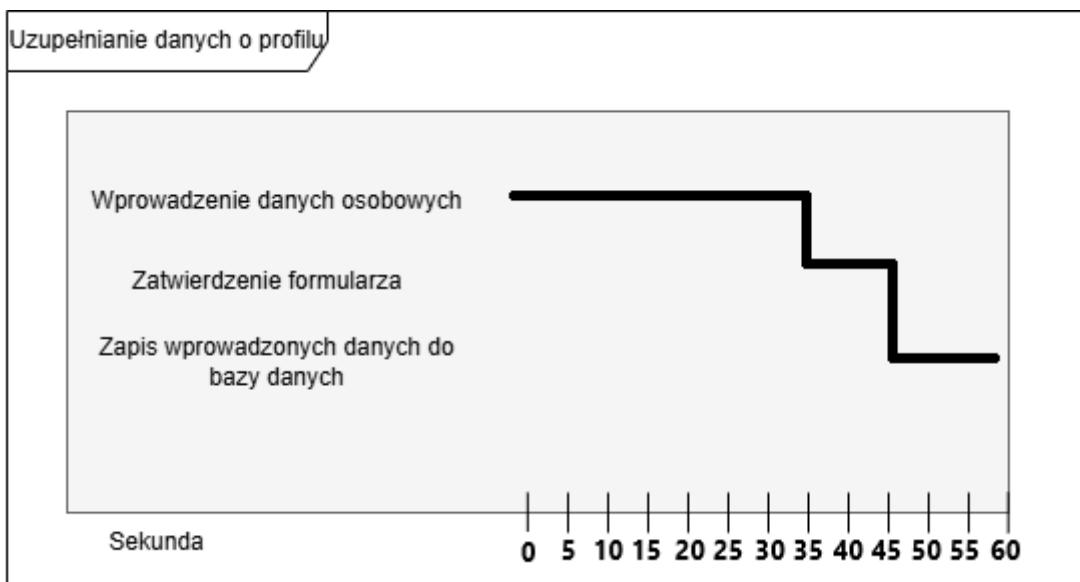
Pakiet **models** zawiera klasy reprezentujące obiekty używane w aplikacji, które są odzwierciedleniem tabel zawartych w bazie danych. Pakiet **repository** odpowiedzialny jest za połączenie z bazą danych. Pakiet **controllers** – zawiera klasę typu Manager oraz za pomocą endpointów udostępnia metody dla **UserUI**, który jest reprezentacją graficzną dla użytkownika. **Service** – zawiera interfejs służący do udostępniania metod dla innych modułów oraz **UserService**, który zawiera metody odpowiedzialne za główną logikę modułu oraz ma bezpośredni dostęp do pakietu **models**. Pakiet **payload** – zawiera w sobie pakiety response oraz request, które zawierają obiekty klas służących jako Data Transfer Object. Response zawiera obiekty DTO z odpowiedziami natomiast request z zapytaniami przesyłanymi z komponentu **UserUI**.

2.11.4.18 Diagram przeglądu interakcji – aplikowanie do organizacji

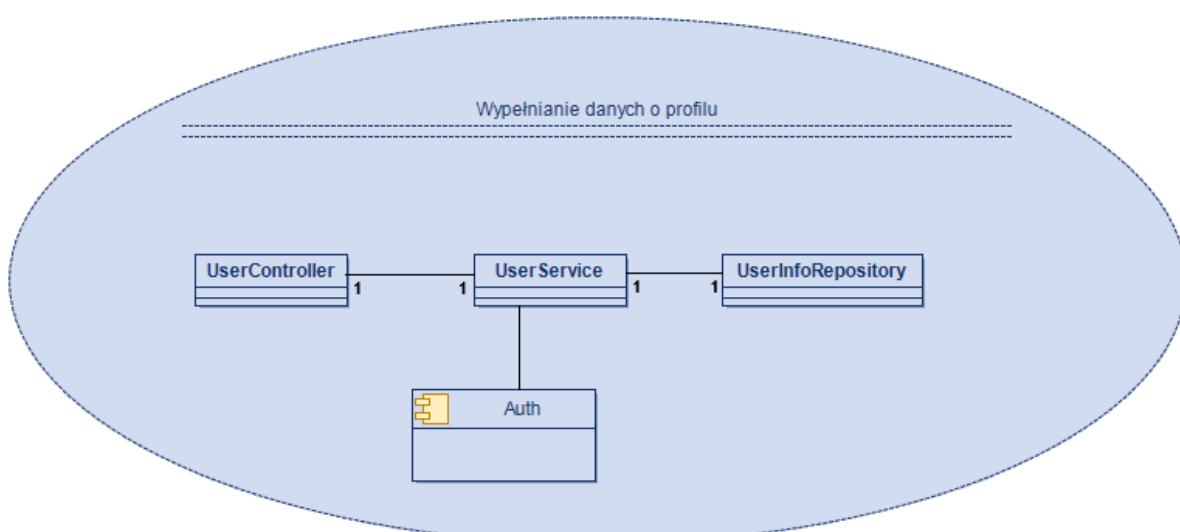


Część I: Dokumentacja techniczna

Diagram przeglądu interakcji przedstawia współpracę diagramów interakcji. W momencie, kiedy użytkownik wysyła aplikacje do organizacji następuje wymiana komunikatów między obiektami. Jeżeli istnieje dana organizacja następuje wymiana komunikatów między obiektem aplikowania do organizacji pomocowej, a wysłaniem aplikacji do organizacji. Natomiast jeżeli dana organizacja nie istnieje to komunikat o błędzie wraca do GUI. Jeżeli pierwszy warunek został spełniony obiekt wysyłania wymienia komunikat z bazą danych. Jeżeli operacja nie przebiegła poprawnie do bazy danych przesyłany jest komunikat o błędzie. W przeciwnym przypadku dane zostają zapisane w bazie.

2.11.4.19 Diagram harmonogramowania

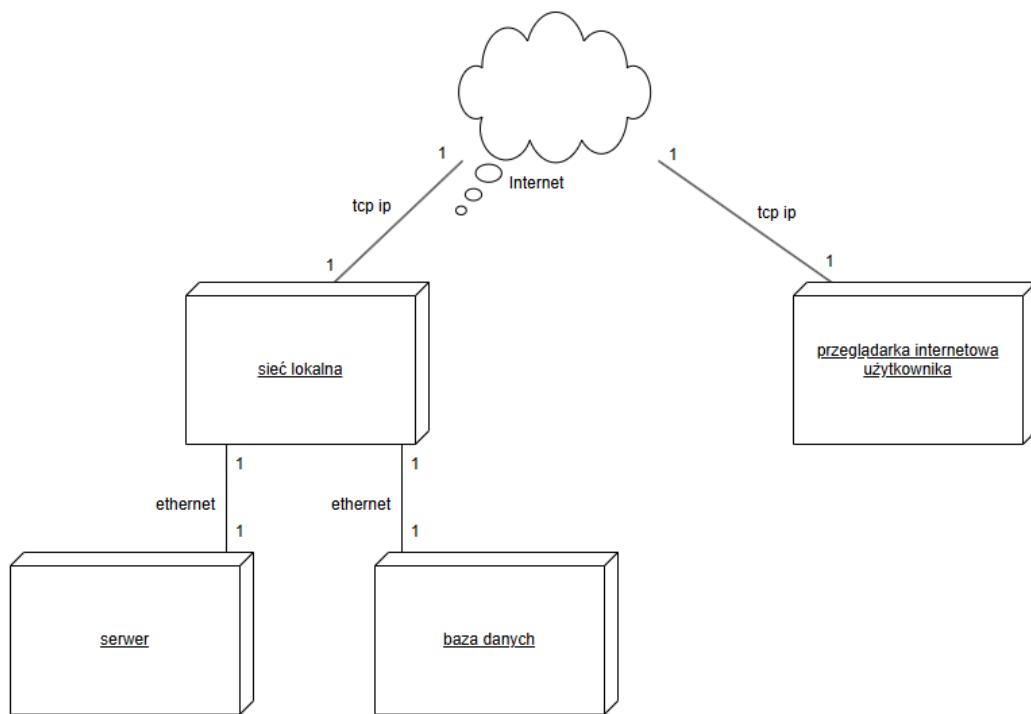
Powyższy diagram reprezentuje na osi czasu zmiany dopuszczalnych stanów jakie może przyjmować obiekt dla przypadku użycia „Uzupełnianie danych o profilu”. Podane zdarzenia to Wprowadzenie danych osobowych, Zatwierdzenie formularza, Zapis wprowadzonych danych do bazy danych. Stan zatwierdzania formularza oraz zapis wprowadzonych danych do bazy zajmuje mniej czasu niż stan wprowadzania danych osobowych.

2.11.4.20 Diagram strukturalny

Część I: Dokumentacja techniczna

Diagram przedstawia kooperację modułu **User** podczas wypełniania danych o profilu użytkownika. W tym procesie wykorzystywany jest moduł **Auth**. Po przejściu na endpoint odpowiedzialny za przyjęcie danych dalsze wykonywanie operacji jest przekazywane do **UserService**, jeśli cała operacja zakończy się powodzeniem to nowe dane o użytkowniku zostaną zapisane w bazie danych.

2.12 Diagram wdrożeń systemu



3 Część II: Dokumentacja użytkownika

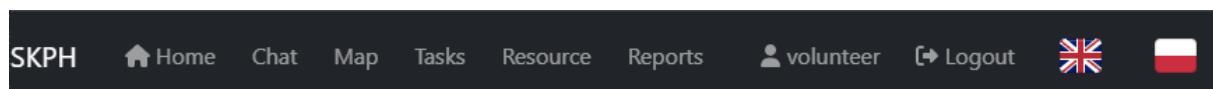
Część II: Dokumentacja użytkownika

3.1 Budowa strony i język aplikacji

Strona składa się z dwóch kluczowych elementów.

3.1.1 Panel nawigacyjny

1. Panel nawigacyjny widoczny na samej górze strony internetowej. Panel służy do nawigowania po zasobach dostępnych dla użytkowników oraz korzystania z usług takich jak zmiana języka czy wylogowywanie/logowanie/rejestracja.



3.1.2 Zawartość zasobu

2. Zawartość zasobu. To zawartość strony pod panelem nawigacyjnym. Przedstawia ona zawartość zasobów danej ścieżki, z której użytkownik obecnie korzysta. Poniżej przykład zawartości zasobu home będącego stroną główną aplikacji.

The screenshot shows a grid of eight cards representing different application features:

- Reports**: Check the reports
- Profile**: Edit your profile
- Resource**: Add resources
- Map**: Check or add reports
- Chat**: Send a message to someone
- Volunteers**: Check volunteers in organization
- Application**: Check application to organization
- Tasks**: Check tasks

3.1.3 Zmiana języka aplikacji

Zmiana języka jest dostępna poprzez kliknięcie odpowiadającej językowi flagi w panelu nawigacyjnym.



W momencie naciśnięcia jednej z flag język systemu zostanie zmieniony.

The screenshot shows the same grid of cards, but the text is now in Polish:

- Reports**: Sprawdź raporty
- Profile**: Edytuj profil
- Resource**: Dodaj zasoby
- Map**: Sprawdź lub dodaj raporty
- Chat**: Wyślij wiadomość kogoś
- Volunteers**: Sprawdź wolontariuszy w organizacji
- Application**: Sprawdź aplikację do organizacji
- Tasks**: Sprawdź zadania

Część II: Dokumentacja użytkownika

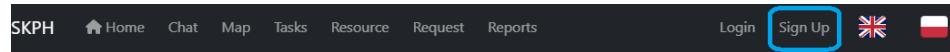
SKPH  Strona główna Czat Mapa Zadania Zadób Request Raporty  Wyloguj się  

Profil Edytuj swój profil	Zasób Dodaj zasoby	Mapa Sprawdź kde dodać raporty
Czat Wyślij wiadomość do kogoś	Wolontariusze Sprawdź wolontariuszy w organizacji	Wnioski Sprawdź otrzymane wnioski

3.2 Rejestracja i logowanie

3.2.1 Rejestracja

Aby korzystać z zasobów aplikacji użytkownik musi być zalogowany. Jeśli ten warunek nie jest spełniony użytkownik musi zarejestrować się klikając na przycisk „Sign Up” w panelu nawigacyjnym.

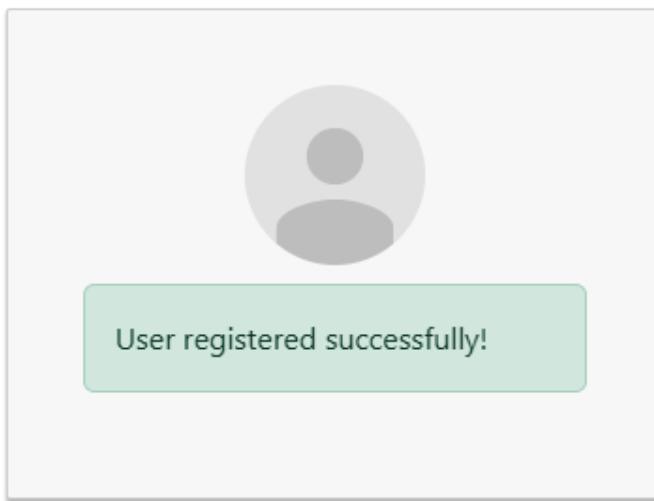
A registration form with the following fields:

- Username: user
- Email: example@email.com
- Password: masked text
- Role: Organization

The form has a large placeholder icon of a person at the top. At the bottom is a blue "Sign up" button.

Następnie należy wypełnić pola tekstowe i kliknąć niebieski przycisk „Sign up” na dole formularza.

Po wypełnieniu kwestionariusza i naciśnięciu przycisku rejestracji wyświetli się komunikat informujący o przebiegu rejestracji.



W przypadku podania niepoprawnych danych, komunikaty wyświetlane są na bieżąco w formularzu



Username

Username is required!

Email

Email is required!

Password

Password is required!

Role
 Victim

Sign up

Error: Username is already taken!

3.2.2 Logowanie

Po poprawnej rejestracji użytkownik powinien przejść do zakładki logowania i wypełnić formularz.



Username
 user

Password

Login

Po naciśnięciu przycisku logowania użytkownik zostanie przeniesiony na stronę główną, skąd ma dostęp do innych funkcjonalności systemu, zgodnych z wybraną przez niego rolą.

Część II: Dokumentacja użytkownika

SKPH

[Home](#)[Chat](#)[Map](#)[Tasks](#)[Resource](#)[Request](#)[Reports](#) [user](#) [Logout](#)**Profile**

Edit your profile

Resource

Add resources

Map

Check or add reports

Chat

Send a message to someone

Volunteers

Check volunteers in organization

Application

Check application to organization

W przypadku podania niepoprawnych danych, komunikaty wyświetlane są na bieżąco w formularzu.



Username

Username is required

Password

Password is required

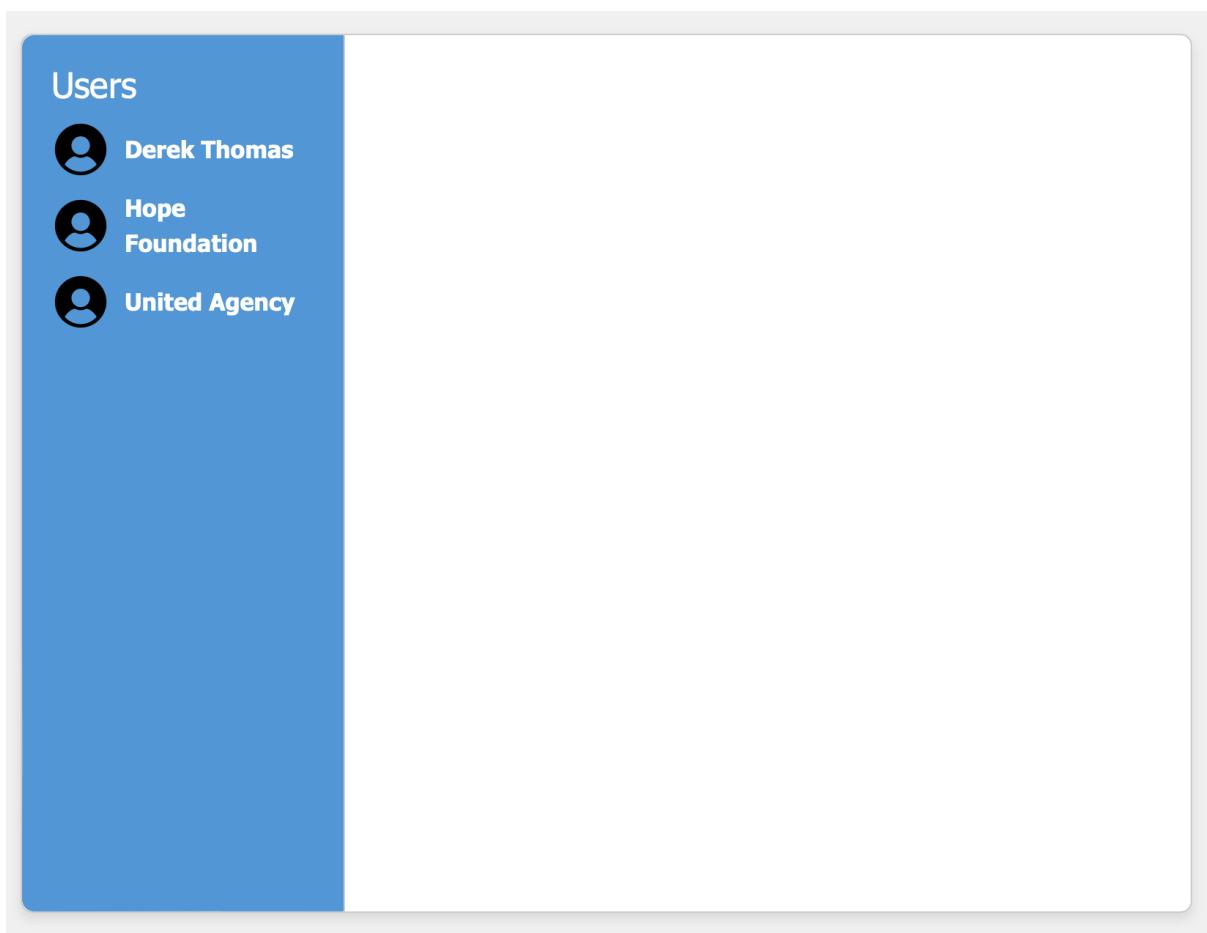
Login

Bad credentials

3.3 Komunikator- przesyłanie i odbiór wiadomości tekstowych

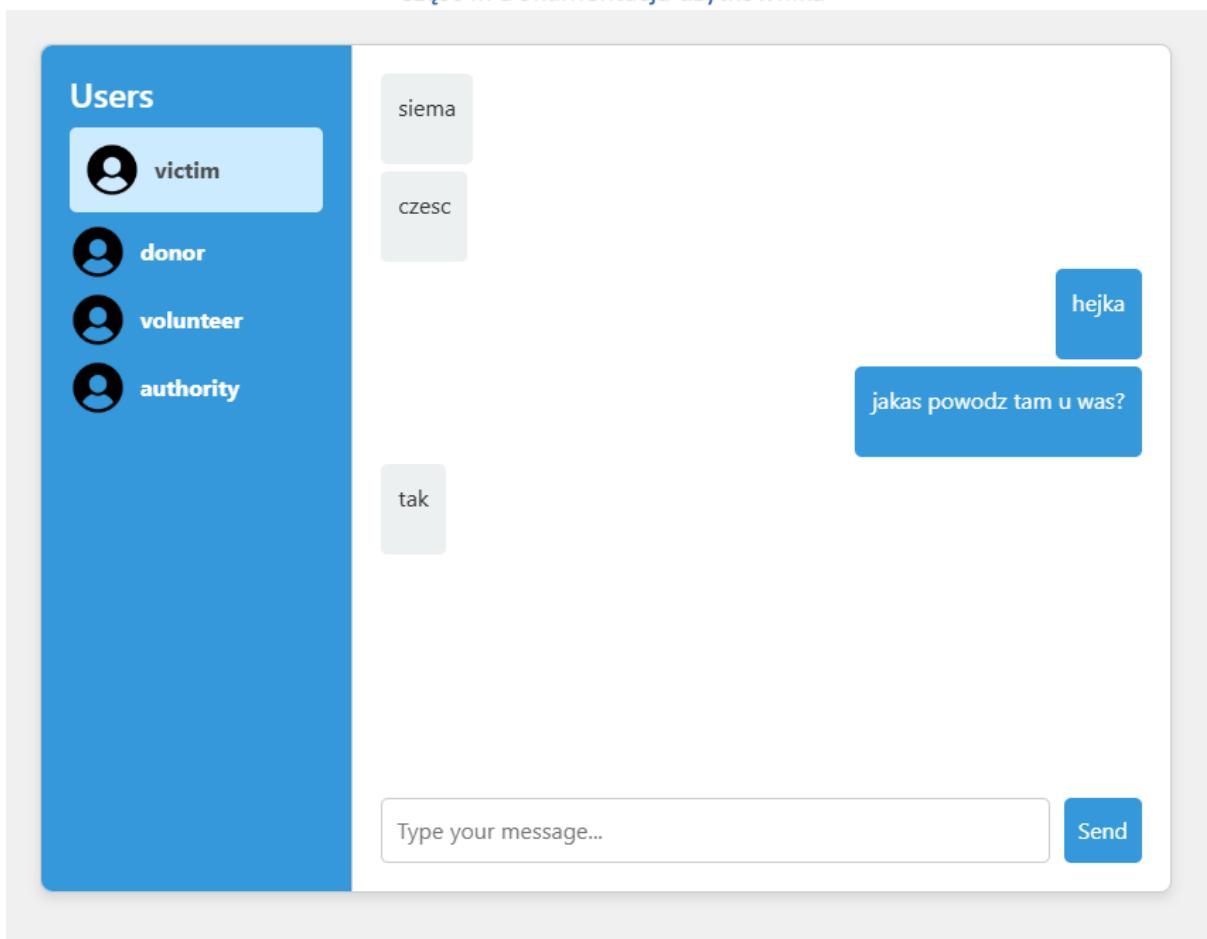
3.3.1 Wybór użytkownika do komunikacji

Komunikator umożliwia wysyłanie i odbiór wiadomości od różnych użytkowników. Niebieski panel po lewej stronie umożliwia wybór użytkownika, którego wiadomości chcemy przeglądać lub wysłać.



3.3.2 Wysyłanie/odbiór wiadomości

Wysyłanie wiadomości i ich przeglądanie jest możliwe dzięki panelowi na prawo od niebieskiej listy użytkowników. Aby otworzyć konwersację należy kliknąć na wybranego użytkownika w niebieskim panelu w efekcie czego wyświetlona zostanie zawartość konwersacji z danym użytkownikiem.



Pole tekstowe na dole umożliwia wprowadzenie wiadomości. Przycisk „Send” umożliwia nadanie wiadomości.

3.4 Mapa

3.4.1 Role użytkowników i ich uprawnienia

Moduł map obsługuje trzy role użytkowników:

→ Poszkodowani:

- Widzą wyłącznie swoje własne zgłoszenia w sekcji "REQUESTS".
- Widzą wszystkie dostępne zasoby w sekcji "RESOURCES" oraz na mapie.

→ Przedstawiciele władz:

- Widzą wszystkie zgłoszenia w sekcji "REQUESTS".
- Widzą wszystkie dostępne zasoby w sekcji "RESOURCES" oraz na mapie.

→ Organizacje pomocowe:

- Widzą wszystkie swoje zgłoszenia w sekcji "REQUESTS".
- Widzą wyłącznie zasoby przypisane do ich organizacji w sekcji "RESOURCES", bez względu na ich dostępność.

3.4.2 Ekran główny

Po wejściu do modułu użytkownik zobaczy:

→ Sekcję "RESOURCES" (Zasoby) po lewej stronie:

- Lista dostępnych zasobów z kolumnami:
 - Name – nazwa zasobu (np. "Water").
 - Quantity – ilość zasobu (np. 100).
 - Location – przycisk "Show on map" umożliwiający wyświetlenie lokalizacji zasobu na mapie.

→ Sekcję "REQUESTS" (Zgłoszenia) poniżej:

- Lista zgłoszeń od użytkowników (w tym przykładzie jest pusta):
 - User – nazwa użytkownika (np. "Michelle Henderson").
 - Name – nazwa zasobu (np. "Water").
 - Location – przycisk "Show on map" umożliwiający wyświetlenie

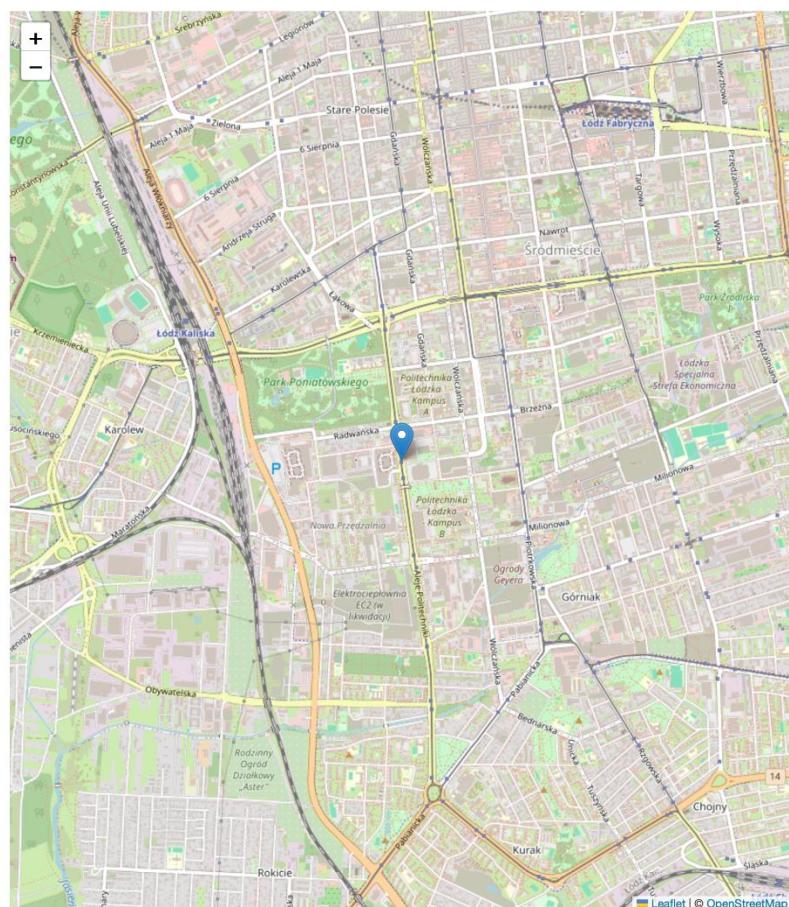
→ Mapę po prawej stronie:

- Interaktywna mapa z możliwością powiększania, przesuwania oraz wybierania lokalizacji.
- Po wybraniu lokalizacji na mapie pojawia się okno dialogowe do dodawania nowych zasobów lub zgłoszeń.

Część II: Dokumentacja użytkownika

RESOURCES

Name	Quantity	Location
Water	100	Show on map
Medical Help	1	Show on map



REQUESTS

User	Name	Location

3.4.3 Wyświetlanie lokalizacji zasobu

- W sekcji "RESOURCES" kliknij przycisk "Show on map" obok zasobu.
- Mapa automatycznie przesunie się do lokalizacji danego zasobu i zaznaczy go pinezką.
- Pinezka pozwala na łatwą identyfikację położenia zasobu na mapie.

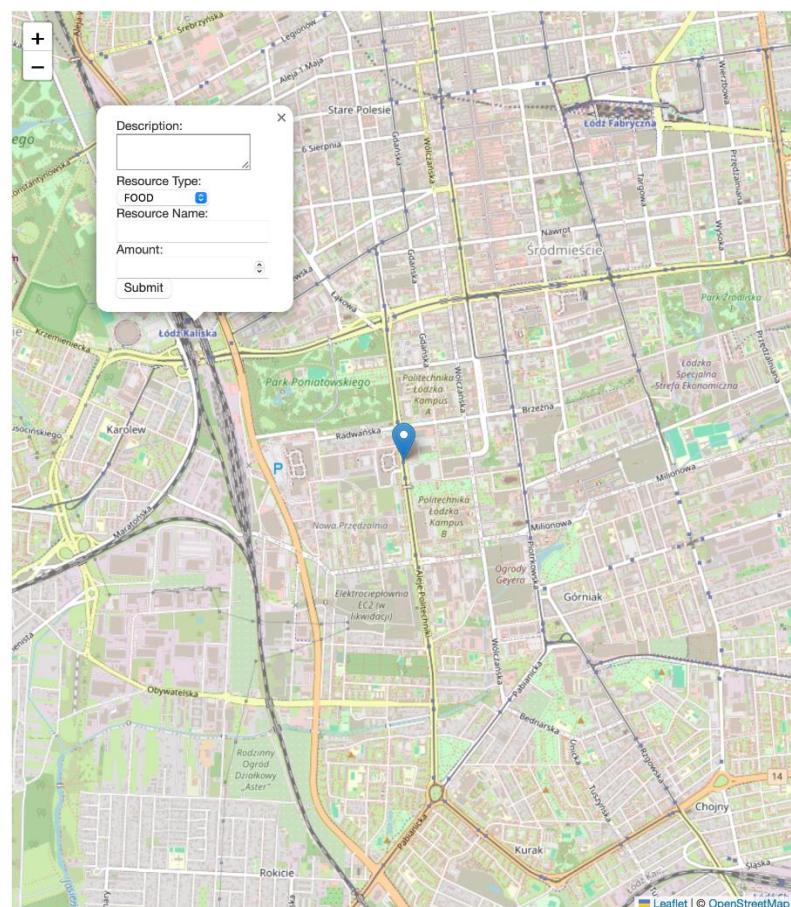
3.4.4 Dodawanie nowych zgłoszeń

- Kliknij na mapie w dowolnym miejscu, gdzie chcesz dodać zasób.

Część II: Dokumentacja użytkownika

RESOURCES

Name	Quantity	Location
Water	100	Show on map
Medical Help	1	Show on map

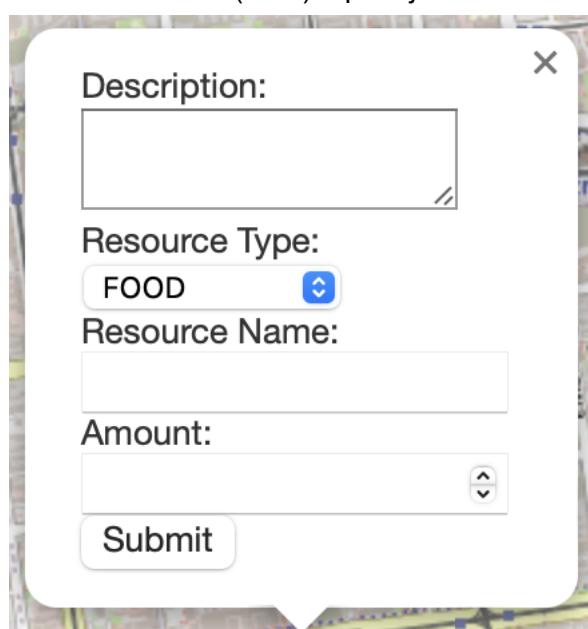


REQUESTS

User	Name	Location

→ Pojawi się okno dialogowe z polami do wypełnienia:

- Resource Type (typ zasobu) – wybierz typ zasobu z listy rozwijanej (np. "FOOD").
- Resource Name (nazwa zasobu) – wpisz nazwę zasobu (np. "Water").
- Amount (ilość) – podaj ilość zasobu.



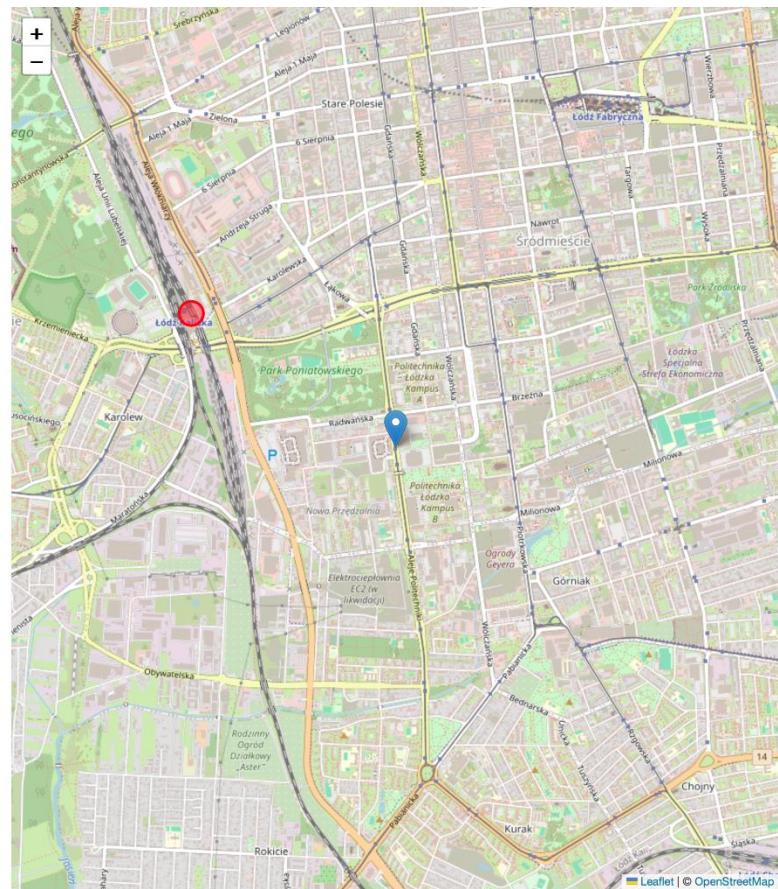
→ Kliknij przycisk "Submit", aby dodać zasób.

Część II: Dokumentacja użytkownika

- Zasób pojawi się na mapie w wybranej lokalizacji oraz w sekcji "REQUESTS".

RESOURCES

Name	Quantity	Location
Water	100	Show on map
Medical Help	1	Show on map



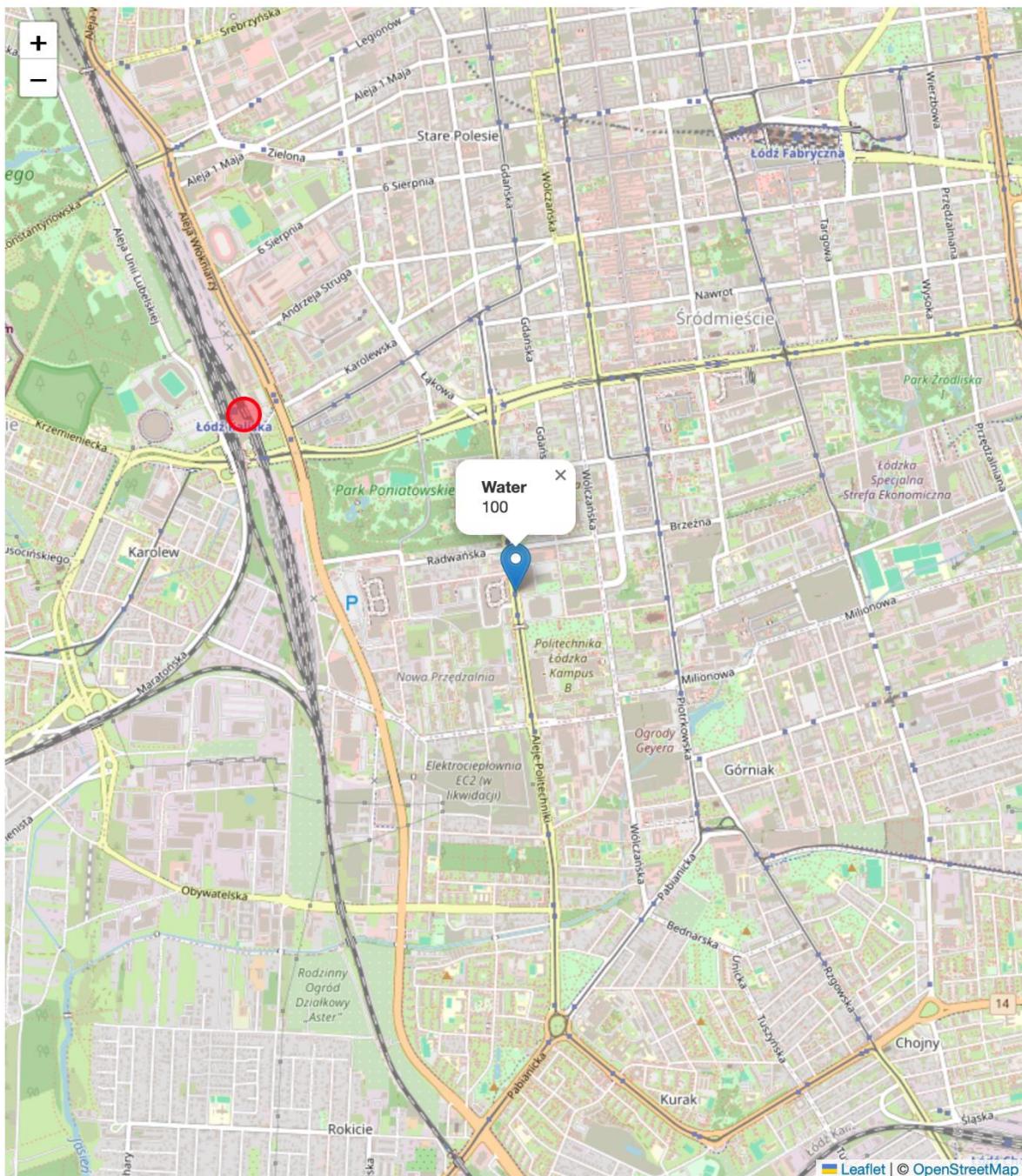
REQUESTS

User	Name	Location
Derek Thomas	Food	Show on map

3.4.5 Nawigacja i interakcja z mapą

- Możesz przesuwać mapę poprzez kliknięcie i przeciąganie myszką.
- Powiększenie i pomniejszenie mapy jest możliwe za pomocą przycisków "+" i "-" w lewym górnym rogu mapy.
- Kliknięcie na istniejącą pinezkę wyświetla szczegóły przypisanego zasobu lub zgłoszenia.

Część II: Dokumentacja użytkownika



3.4.6 Wyznaczanie trasy z zasobu do zgłoszenia

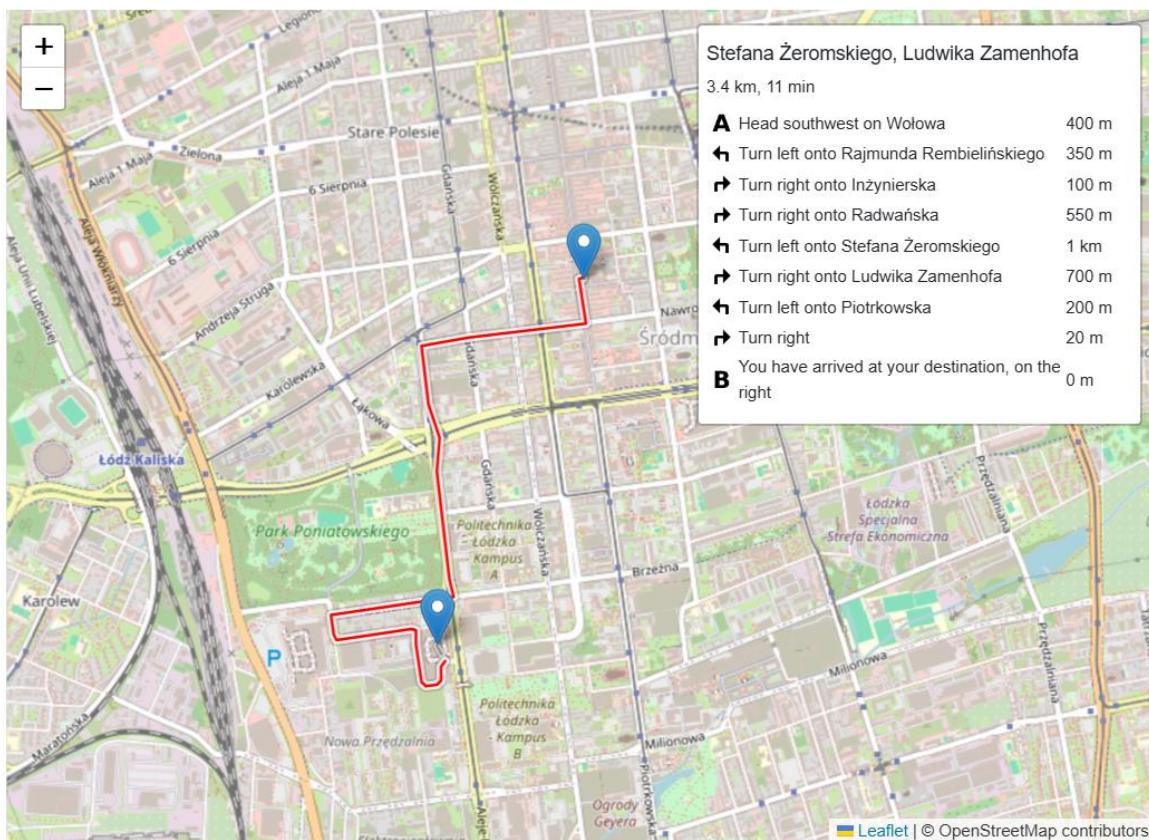
Wolontariusz w zasobie Zadania ma przycisk „route” umożliwiający mu przeniesienie do mapy z wyznaczoną trasą jaką ma podążać od zasobu do zgłoszenia

							Info	End	Route
2	nowe zadanie	PCK	lodz srodmiescie	Medium	In progress	Not graded	Info	End	Route

[Previous](#) [Next](#)

W wyniku kliknięcia przycisku (o ile zadanie nie zostało zakończone) ukaże się użytkownikowi taka mapa

Część II: Dokumentacja użytkownika



Legenda dostarcza informacje o przebiegu trasy od zasobu.

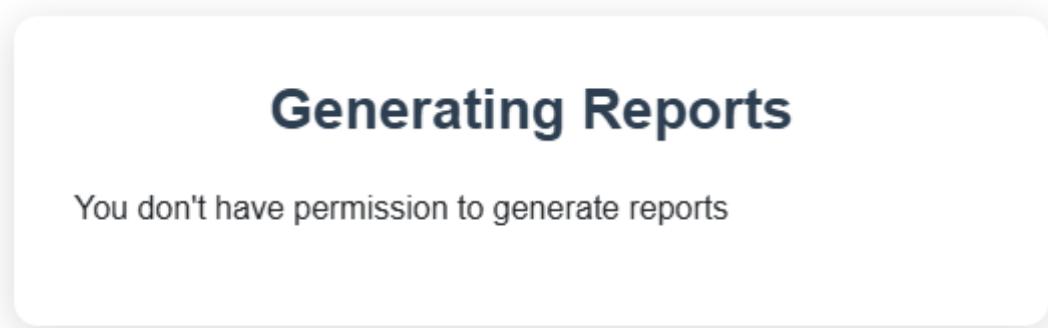
3.5 Raporty

3.5.1 Uprawnienia do generowania raportów

Moduł raportowania jest dostępny tylko dla użytkowników o rolach:

1. Organizacja pomocowa
2. Przedstawiciel władz
3. Darczyńca

Próba wejścia do zakładki „reports” przez innego użytkownika zostanie uniemożliwiona i zostanie wyświetlony odpowiedni komunikat o braku uprawnień:



3.5.2 Raporty Darczyńcy

W przypadku zalogowania się na konto Darczyńcy i przejście do zakładki „reports” zostanie wyświetcone poniższe okno z menu raportów dla tego użytkownika.

A screenshot of a web page titled "Generating Reports". It features a dropdown menu labeled "Report Type" with a list of options. Below the dropdown are two blue buttons: "Download Report" and "View report". The entire interface is contained within a light gray box with rounded corners.

Po rozwinięciu „select boxa” pojawią się następujące opcje :

Generating Reports

Report Type

- [Donation report](#)
- [Tax report](#)

W przypadku wyboru „Donation report” i wcisnienia przycisku „View report” zostanie pokazany raport potwierdzający wszystkie dotację dokonane przez zalogowanego użytkownika.

Generating Reports

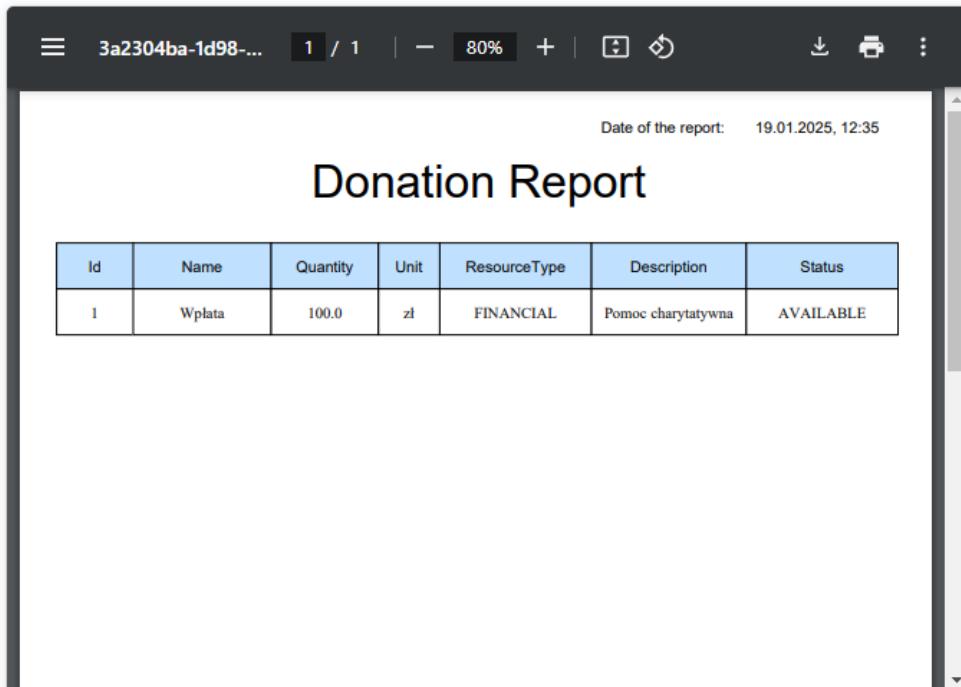
Report Type

Donation report

Download Report

View report

Report preview



Część II: Dokumentacja użytkownika

W przypadku wyboru „Tax report” i wcisnięcia przycisku „View report” zostanie pokazany raport ulgi podatkowej zawierający łączną sumę wpłat pieniężnych dokonanych przez niego oraz jego dane osobowe.

Generating Reports

Report Type

Tax report

Download Report**View report**

Report preview

The screenshot shows a report preview window with the following details:

Tax Report

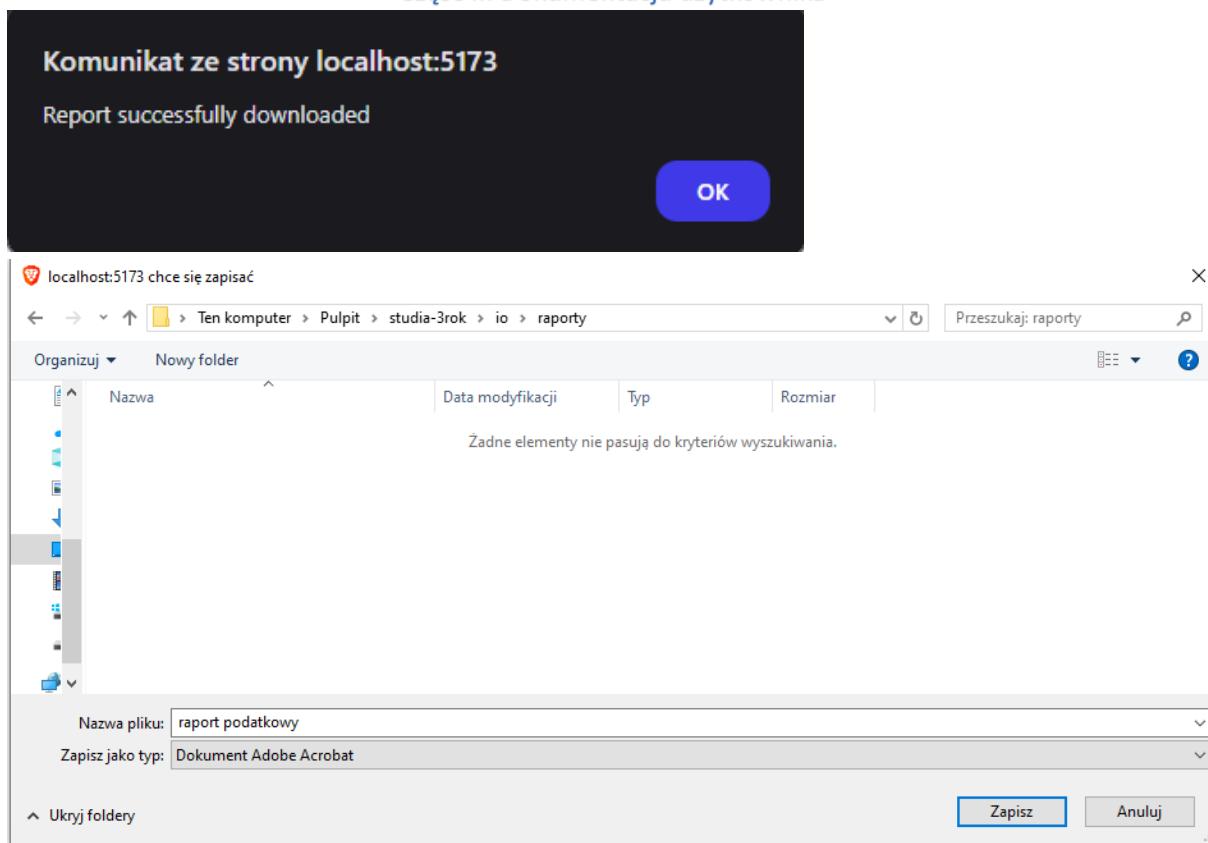
Date of the report: 19.01.2025, 12:36

Imie: Andrzej
Nazwisko: Nowak
PESEL: 01234567899

Suma ulgi podatkowej: 100.0 zł

Raporty można również pobrać za pomocą wybrania typu raportu i wcisnięcia przycisku „Download report”. Następnie zostanie wyświetlony komunikat o powodzeniu tej operacji i zostanie otworzone okno z wyborem miejsca zapisu.

Część II: Dokumentacja użytkownika



3.5.3 Raporty Organizacji pomocowej oraz Przedstawicieli władz.

Organizacja pomocowa oraz Przedstawiciel władz mają dostęp do tych samych raportów takich jak:

1. Task report
2. Request report
3. Resource report

Generating Reports

Report Type

- Task report
- Request report
- Resource report

[View report](#)

Tak jak w przypadku poprzednich raportów, Organizacja pomocowa oraz przedstawiciel władz ma możliwość ich podglądu oraz pobrania i zapisania w wybranym miejscu na urządzeniu.

Menu raportów jest obsługiwane zarówno w języku angielskim :

Generating Reports

Report Type

[Download Report](#)

[View report](#)

Oraz w języku polskim

Generowanie raportów

Typ raportu

[Pobierz raport](#)

[Podgląd raportu](#)

Szablony raportów są zawsze w języku angielskim, a dane są pobierane z baz danych i nie mamy wpływu na to w jakim języku zostaną podane.

3.6 Zgłoszenia

Dostęp do modułu zgłoszeń w menu głównym ma tylko zalogowany użytkownik, który posiada rolę poszkodowanego albo przedstawiciela władzy. W przypadku, gdy nieautoryzowany użytkownik przejdzie na stronę modułu, wyświetla się informacja o odmowie dostępu.

Access denied

Come back home

3.6.1 Funkcje dla poszkodowanego

Użytkownik, który posiada rolę poszkodowanego po wybraniu modułu zgłoszeń w menu głównym ma możliwość dodania nowego zgłoszenia oraz wyświetlenia listy wszystkich swoich zgłoszeń.

3.6.1.1 Zgłaszcenie potrzeb

Po wybraniu panelu zgłaszania nowej potrzeby użytkownikowi wyświetla się formularz, w którym musi wybrać typ potrzebnego zasobu z listy, wprowadzić jego nazwę oraz niezbędną ilość. Następnie może wprowadzić dodatkowy opis oraz koniecznie zaznaczyć lokalizację zgłoszenia na mapie. Po wypełnieniu formularza poszkodowany potwierdza dodanie nowej potrzeby poprzez wcisnięcie przycisku. W przypadku prawidłowo podanych danych wyświetli się komunikat o powodzeniu operacji, zaś w przypadku podania błędnych danych bądź niewypełnieniu wszystkich wymaganych pól – odpowiedni komunikat o błędzie.

Create a request

Resource Type:

Name:

Amount:

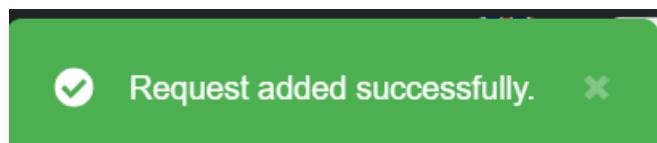
Description:

Location:

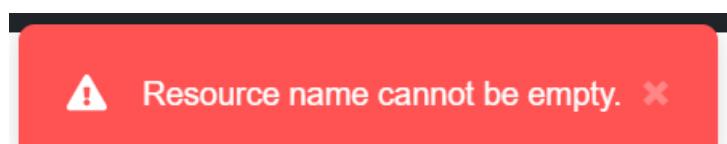
Send

Część II: Dokumentacja użytkownika

Formularz dodawania nowego zgłoszenia



Komunikat o poprawnym dodaniu zgłoszenia



Przykładowy komunikat o błędzie przy dodawaniu zgłoszenia

3.6.1.2 Wyświetlenie listy zgłoszeń

Poszkodowany ma możliwość wyświetlenia listy wszystkich swoich zgłoszeń wraz z informacjami o nich. Dodatkowo podczas wyświetlania listy może edytować zgłoszenia oraz usunąć istniejące zgłoszenia. Obie te czynności można wykonać tylko, gdy zgłoszenie ma status zarejestrowane (registered) oraz skutkuje wyświetleniem odpowiedniego komunikatu potwierdzającego operację, bądź informującego o niepowodzeniu.

ID:	11		
Description:			
Amount:	3		
Resource Type:	Food		
Name:	bread		
Status:	Registered		
Reporter:	<ul style="list-style-type: none"> ○ Name: TestName ○ Surname: TestSurname ○ Email: vict@gmail.com 		

Lista zgłoszeń poszkodowanego

ID:	12		
Description:	warm		
Amount:	4		
Resource Type:	Clothing		
Name:	socks		
Status:	Registered		
Reporter:	<ul style="list-style-type: none"> ○ Name: TestName ○ Surname: TestSurname ○ Email: vict@gmail.com 		

Część II: Dokumentacja użytkownika



ID: 11

Description:

Amount: 3

Type: Food

Name: bread

Status: Registered

Reporter:

- Name: TestName
- Surname: TestSurname
- Email: vict@gmial.com



Edycja zgłoszenia

Request updated successfully. 

Komunikat o poprawnej aktualizacji zgłoszenia

3.6.2 Funkcje dla przedstawiciela władzy

Użytkownik, który posiada rolę przedstawiciel władzy po wybraniu modułu zgłoszeń w menu głównym ma możliwość wyświetlenia listy wszystkich istniejących zgłoszeń.

3.6.2.1 Wyświetlenie listy zgłoszeń

Przedstawiciel władzy ma możliwość wyświetlenia listy wszystkich istniejących zgłoszeń wraz z informacjami o nich oraz podstawowymi informacjami o twórcy tego zgłoszenia. Podczas wyświetlania listy istnieje możliwość usunięcia dowolnego zgłoszenia, które ma status zarejestrowane, co skutkuje wyświetleniem komunikatu o poprawnym usunięciu zgłoszenia.

SYSTEM KOORDYNACJI POMOCY HUMANITARNEJ

Część II: Dokumentacja użytkownika

<input type="button" value="X"/>
ID: 11
Description:
Amount: 3
Resource Type: Food
Name: bread
Status: Registered
Reporter:
◦ Name: TestName
◦ Surname: TestSurname
◦ Email: vici@gmail.com



A map of a residential area in Zarzew, Poland. A red circle marks the location of the resource. The map includes street names like "Młyńska", "Kamienna", and "Zarzew". A legend at the bottom right indicates "Leaflet | © OpenStreetMap".

<input type="button" value="X"/>
ID: 13
Description:
Amount: 7
Resource Type: Food
Name: water
Status: Registered
Reporter:
◦ Name: Name
◦ Surname: Surname
◦ Email: testowy@gmail.com



A map of the Łódź area, specifically around the "Politechnika Łódzka - Kampus B" and "Nowa Przedmieścia" districts. A red circle marks the location of the resource. The map includes street names like "Radwańska", "Mazurska", and "Karolew". A legend at the bottom right indicates "Leaflet | © OpenStreetMap".

3.7 Zasoby

Po zalogowaniu oraz po przejściu do modułu Resource w zależności od roli możemy wykonywać różne funkcje. Do modułu Resource mają dostęp role: Organizacja, Darczyńca oraz Władze. Każda z tych ról posiada możliwość wykonywania następujących funkcji:

3.7.1 Organizacja

wyświetla zasoby swojej organizacji z możliwością filtrowania i sortowania

dodawanie zasobu do swojej puli zasobów

modyfikacja swoich zasobów

3.7.2 Władze

możliwość wyświetlania wszystkich zasobów z informacją, do której organizacji należą, z możliwością filtrowania i sortowania

3.7.3 Darczyńca

historia swoich darowizn z możliwością filtrowania i sortowania

przekazanie daru do konkretnej organizacji

wyświetlenie wykorzystania daru

3.7.4 Rodzaje zasobów

Finanse

Zasoby finansowe, takie jak gotówka, darowizny pieniężne czy inne środki pieniężne.

Sprzęt

Narzędzia, urządzenia i inne zasoby techniczne (np. komputery, maszyny, narzędzia ręczne).

Mieszkanie

Zasoby związane z zakwaterowaniem i wyposażeniem domowym, takie jak domy, mieszkania, meble, sprzęt AGD, czy elementy wyposażenia kuchni.

Transport

Środki transportu, takie jak samochody, rowery, pojazdy użytkowe.

Jedzenie

Produkty spożywcze oraz napoje przeznaczone do spożycia.

Odzież

Ubrania, obuwie i akcesoria.

Substancje medyczne

Leki, materiały medyczne (np. bandaże, strzykawki) oraz inne środki wykorzystywane w opiece zdrowotnej.

Inne

Zasoby, które nie mieścią się w powyższych kategoriach, takie jak gry, książki, zabawki czy artykuły.

3.7.5 Statusy zasobów

Dostępny

Część II: Dokumentacja użytkownika

Zasób jest dostępny do użytku i gotowy do przypisania lub wykorzystania.

Przypisany

Zasób został w całości przypisany do konkretnego celu, osoby lub projektu i nie jest już dostępny do dalszej dyspozycji.

Uszkodzony

Zasób jest uszkodzony i wymaga naprawy lub wymiany. Może być częściowo lub całkowicie niefunkcjonalny.

Przeterminowany

Zasób przekroczył swoją datę ważności lub nie jest już użyteczny (np. przeterminowane jedzenie, nieaktualne bilety, środki medyczne po dacie ważności).

3.7.6 Widok darczyńcy

id ↑	Nazwa ↓	Opis	Ilość ↑	Status	Data dodania ↑	Termin ważności ↑
172	\$ Pieniądze	88/100 zł	DOSTEPNY	2025-01-19		Użycie

Po przejściu do modułu Resource darczyńca widzi swoją historię dodanych zasobów, może ją filtrować po statusie lub typie oraz sortować po nazwie ilości lub dacie dodania/przeterminowania po kliknięciu w strzałkę obok nazwy kolumny lub zobaczyć użycie daru. Dodatkowo śledzenie ile zostało daru można sprawdzić w kolumnie ilość, jako (aktualna ilość / łączna ilość).

Część II: Dokumentacja użytkownika

The screenshot shows the 'Przekaż darowiznę' (Send donation) dialog box. It includes fields for 'Nazwa' (Name), 'Opis' (Description), 'Ilość' (Quantity) with unit 'szt' (unit), 'Lokalizacja' (Location with a map), 'Typ' (Type: INNE), 'Wybierz organizację' (Select organization), and 'Termin ważności' (Valid until: dd.mm.rrrr). There are also 'Uzycie' (Use) and 'Uzycie' (Use) buttons.

Po kliknięciu przycisku "Przekaż darowiznę" darczyńca musi wpisać nazwę, opis (opcjonalnie), ilość, miarę, typ zasobu, wybrać organizację, w imieniu której przekazuje dar oraz lokalizację na mapie. Ilość musi być liczba większa od 0. Dodatkowo przy wyborze typu zasobu JEDZENIE i MEDYCZNE użytkownik musi wybrać termin ważności.

The screenshot shows the 'Użycie darowizny' (Use donation) dialog box. It displays a list of donations used:

- ID Zgłoszenia: 4 Ilość: 3 zł
- ID Zgłoszenia: 5 Ilość: 3 zł
- ID Zgłoszenia: 8 Ilość: 3 zł
- ID Zgłoszenia: 12 Ilość: 3 zł

Below the list are checkboxes for categories: FINANSE, SPRZĘT, ZAKWATEROWANIE, and INNE. There are 'Cancel' and 'OK' buttons.

Po kliknięciu przycisku "Użycie" wyświetla się lista zgłoszeń do których została użyta przekazana darowizna.

Część II: Dokumentacja użytkownika

3.7.7 Widok władz

The screenshot shows the 'Resource' module interface. At the top, there are two filter sections: 'Filter by status' (with options EXPIRED, AVAILABLE, ASSIGNED, DAMAGED) and 'Filter by type' (with options FOOD, TRANSPORT, CLOTHING, MEDICAL, FINANCIAL, EQUIPMENT, HOUSING, OTHER). Below these is a search bar labeled 'Search by name' with the placeholder 'Enter name to search...'. The main area displays a table of resources:

Name ↓	Description	Quantity ↑	Status	Added Date ↑	Expiration Date ↑	Organization
Woda	5 L	DAMAGED	2025-01-19	2026-02-14	testnazwa	
Suzarka	1 szt	AVAILABLE	2025-01-19		testnazwa	
Strzykawki	50/100 szt	AVAILABLE	2025-01-19	2025-05-31	testnazwa	
Spodnie	0/3 szt	ASSIGNED	2025-01-19		testnazwa	
Samochód	1 szt	AVAILABLE	2025-01-19		testnazwa	
Pluszaki	5 szt	AVAILABLE	2025-01-19		testnazwa	
Pieniądze	100 zł	AVAILABLE	2025-01-19		testnazwa	
Pieniądze	100 zł	AVAILABLE	2025-01-19		testnazwa	

Po przejściu do modułu Resource władze widzą dodane zasoby, mogą je filtrować po statusie lub typie oraz sortować po nazwie ilości lub dacie dodania/przeterminowania. W kolumnie Organization wyświetla się nazwa organizacji do której należy zasób lub ID tej organizacji, jeżeli nie posiada ona nazwy. Tak samo jak przy darowiznie, ilość zasobu jest wyrażana jako (aktualna ilość / łączna ilość).

3.7.8 Widok organizacji

Nazwa	Opis	Ilość	Status	Data dodania	Termin ważności
Cukier		2 kg	DOSTĘPNY	2025-01-19	2027-11-20
Buty		2 szt	DOSTĘPNY	2025-01-19	

Po przejściu do modułu organizacja widzi swoje zasoby, może je filtrować po statusie lub typie oraz sortować po nazwie, ilości lub dacie dodania/przeterminowania. Może również edytować oraz usuwać zasoby, które nie są już przydzielone.

Po kliknięciu przycisku "Dodaj zasób" Organizacja musi wpisać nazwę, opis (opcjonalnie), ilość, miarę, typ zasobu oraz wybrać lokalizację na mapie. Ilość musi być większa od 0 oraz przy wyborze typu zasobu JEDZENIE oraz MEDYCZNE użytkownik musi wybrać termin ważności zasobu.

Część II: Dokumentacja użytkownika

Zasoby

Dodaj Zasób

Filtruj po statusie

- PRZETERNINOWANY
- DOSTĘPNY
- PRZYDZIELONY
- USZKODZONY

Wyszukaj po nazwie

Wpisz nazwę żeby wyszukać...

Nazwa ↓	Opis	Ilość ↑
Woda	5 L	
Suszarka	1 szt	
Strzykawki	50/100 szt	
Spodnie	0/3 szt	
Samochód	1 szt	

Lokalizacja

Opis

Wpisz opis...

Status

USZKODZONY

Ilość

5

Cancel OK

Edytuj Usuń

Edytuj Usuń

Edytuj Usuń

Edytuj Usuń

Edytuj Usuń

Po kliknięciu przycisku “Edytuj” przy wybranym przez organizację zasobie ma ona możliwość zmiany lokalizacji, opisu, ilości lub oznaczenia zasobu jako uszkodzony (tylko zasoby dostępne można tak oznaczać).

Przypisywanie zasobu do potrzeby odbywa się w module Task w trakcie tworzenia zadania.

3.8 Zadania

Zasób zadań w systemie koordynacji pomocy humanitarnej umożliwia efektywne zarządzanie zadaniami zarówno organizacjom, jak i wolontariuszom. Zaprojektowany z myślą o prostocie i funkcjonalności, moduł wspiera kluczowe procesy w ramach działań pomocowych, ułatwiając organizację i realizację zadań.

3.8.1 Funkcjonalności według roli użytkowników

3.8.1.1 Dla organizacji:

Przegląd wszystkich zadań oraz możliwość filtrowania ich według różnych kryteriów

Tworzenie nowych zadań z określeniem ich szczegółów, takich jak lokalizacja, priorytet oraz wymagane zasoby.

Przydzielanie zadań do odpowiednich wolontariuszy na podstawie ich dostępności.

Ocena realizacji zadań po ich zakończeniu, co umożliwia bieżące monitorowanie jakości pracy.

3.8.1.2 Dla wolontariuszy:

Przegląd przypisanych zadań oraz możliwość filtrowania ich według różnych kryteriów

Aktualizacja statusu zadania, co pozwala na bieżące raportowanie postępów.

Dzięki zróżnicowanym rolom i dopasowanym funkcjom moduł Task wspiera sprawne działanie organizacji pomocowych oraz ułatwia wolontariuszom efektywne wypełnianie swoich obowiązków.

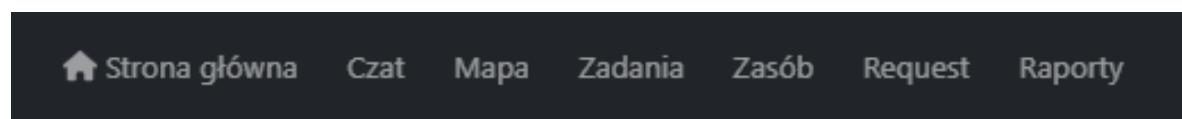
3.8.2 Opis roli i uprawnień użytkownika

Organizacja to podmiot, który uczestniczy w działaniach humanitarnych poprzez planowanie, koordynację i realizację zadań. Może to być fundacja, stowarzyszenie, organizacja non-profit lub inna instytucja zajmująca się pomocą.

Wolontariusz to jednostka wspierająca działania organizacji w ramach swoich kompetencji i dostępnego czasu. Wolontariusze angażują się w realizację konkretnych zadań przypisanych przez organizacje.

3.8.3 Kompletny opis funkcjonalności i elementów

Użytkownik zalogowany jako organizacja lub wolontariusz ma dostęp do modułu Task poprzez panel nawigacyjny.



Po kliknięciu na zasób, wyświetla się lista wszystkich dostępnych zadań przypisanych do danej roli.

Organizacje mają dostęp do wszystkich zadań, zarówno tych przypisanych do ich organizacji, jak i do zadań należących do innych organizacji.

Zadania

+ Utwórz zadanie

Wyszukaj po tytule		Status		Filtruj po		Wartość filtra				
Wpisz słowo kluczowe		Wszystkie		Brak						
#	Tytuł	Organizacja	Lokalizacja	Priorytet	Status	Ocena	Informacje	Edytuj	Zakończ	Trasa
1	Wazne zadanie	Organizacja1	Krakow	Wysoki	Ocenione	4	Informacje	Ocena	Zakończ	Trasa
2	Inne ważne zadanie	Organizacja1	Lodz	Niski	W trakcie	Nieocenione	Informacje	Edytuj	Zakończ	Trasa
3	Tytuł zadania	Organizacja1	Warsow	Krytyczny	Zakończone	Nieocenione	Informacje	Ocena	Zakończ	Trasa
5	Tytul	Organizacja2	Krakow	Średni	W trakcie	Nieocenione	Informacje	Edytuj	Zakończ	Trasa
6	Tytul2	Organizacja2	Lodz	Wysoki	Zakończone	Nieocenione	Informacje	Ocena	Zakończ	Trasa
4	Zadania organizacji2	Organizacja2	Gdansk	Wysoki	Ocenione	5	Informacje	Ocena	Zakończ	Trasa
7	Zadanie do zrobienia	Organizacja1	Wroclaw	Krytyczny	Zakończone	Nieocenione	Informacje	Ocena	Zakończ	Trasa

Poprzedni Następny

Lista zadań dla organizacji

Wolontariusze mają dostęp jedynie do zadań przypisanych bezpośrednio do nich.

Zadania

Wyszukaj po tytule Status Filtruj po Wartość filtra

#	Tytuł	Organizacja	Lokalizacja	Priorytet	Status	Ocena	Informacje	Zakończ	Trasa
1	Wazne zadanie	Organizacja1	Krakow	Wysoki	Ocenione	4	Informacje	Zakończ	Trasa
2	Inne ważne zadanie	Organizacja1	Lodz	Niski	W trakcie	Nieocenione	Informacje	Zakończ	Trasa

Poprzedni Następny

Lista zadań wolontariusza

Listę zadań można filtrować na podstawie słów kluczowych zawartych w tytule zadania lub przez wybór jednego z dostępnych kryteriów filtrowania.

Search by title

Enter keyword

Pole filtrowania po tytule

Status

Wszystkie

- Wszystkie**
- W trakcie
- Zakończone
- Ocenione

Filtruj po

Brak

- Brak**
- Priorytet
- Organizacja
- Lokalizacja

Rozwinięte pola filtrowania po statusie oraz innych cechach zadania

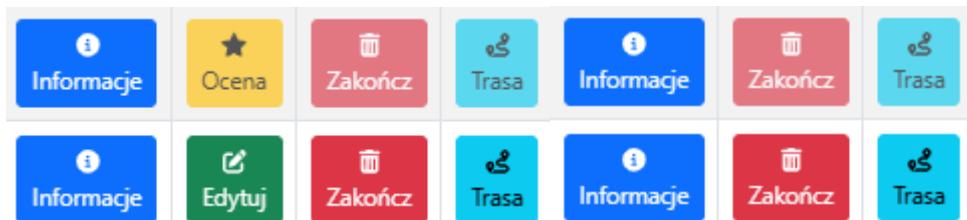
Lista zadań obsługuje również stronicowanie, co ogranicza maksymalną liczbę wyświetlanych zadań na jednej stronie.

Poprzedni Następny

Przyciski nawigacyjne stronicowania

Część II: Dokumentacja użytkownika

Każde zadanie posiada przyciski umożliwiające zarządzanie nim, które znajdują się po prawej stronie listy.



Przyciski do zarządzania zadaniami dla organizacji oraz wolontariusza

1. Informacje - przejście do strony zadania, gdzie wyświetlane są szczegółowe informacje o zadaniu.
2. Zakończ - przycisk zmieniający status zadania na ZAKOŃZONE.
3. Edytuj – przejście do strony edycji zadania, gdzie można zmodyfikować szczegóły.
4. Trasa – przycisk przekierowuje użytkownika do komponentu mapy, wyznaczając trasę pomiędzy lokalizacjami zasobu (resource) i wniosku (request).
5. Ocena – przejście do strony oceny zadania, umożliwiającej ocenienie wykonania zadania.

Organizacje mają również dostęp do przycisku „Utwórz zadanie” na stronie listy zadań, który przenosi je na stronę do tworzenia nowych zadań.

+ Utwórz zadanie

Przycisk „Utwórz zadanie”

SYSTEM KOORDYNACJI POMOCY HUMANITARNEJ

Część II: Dokumentacja użytkownika

[Wróć](#)

Tytuł	<input type="text" value="Tytuł"/>
Opis	<input type="text" value="Opis"/>
Lokalizacja	<input type="text" value="Lokalizacja"/>
Priorytet	<input type="text" value="Priorytet"/>
Status	<input type="text" value="W trakcie"/>

Wniosek

<input type="text" value="Wyszukaj po nazwie wniosku"/>	<input type="text" value="Filtruj po typie"/>						
ID	Nazwa	Opis	Typ	Ilość	Lokalizacja	Zgłaszący	Akcja
12	Thermal Jackets	Winter clothing for refugees in Warsaw	Oдежд	200	52.2298, 21.0118	Nie podano	Wybierz
18	Winter Sleeping Bags	Sleeping bags for survivors in Katowice	Oдежд	120	50.2649, 19.0238	Nie podano	Wybierz

[Poprzedni](#) [Następny](#)

Wolontariusze

Imię i nazwisko	Nazwa użytkownika	E-mail	Akcja
Brak danych	Zosia	zosia123@wp.pl	Wybierz

[Poprzedni](#) [Następny](#)

Zasoby

<input type="text" value="Wyszukaj po nazwie wniosku"/>	<input type="text" value="Filtruj po typie"/>					
Nazwa zasobu	Opis	Typ	Lokalizacja	Dostępna ilość	Ilość do przydzielenia	Akcja
Narzędzia	Dlugi tekst do testowania.	Sprzęt	30.2297 21.0122	10 pcs	<input type="text" value="Wpisz ilość"/>	Wybierz
Leki	Dlugi tekst do testowania.	Medykamenty	30.2297 21.0122	11 pcs	<input type="text" value="Wpisz ilość"/>	Wybierz
Woda	Dlugi tekst do testowania.	Jedzenie	30.2297 21.0122	11 pcs	<input type="text" value="Wpisz ilość"/>	Wybierz

[Poprzedni](#) [Następny](#)

Zatwierdź

Formularz tworzenia zadania, który umożliwia wprowadzenie tytułu, opisu, lokalizacji i priorytetu, a także wybór wniosku, wolontariuszy i zasobów.

Część II: Dokumentacja użytkownika

Strona edycji zadania pozwala jedynie na edytowanie tytułu, opisu, lokalizacji oraz priorytetu.

Inne ważne zadanie

Tytuł
Inne ważne zadanie

Opis
Nic ciekawego w opisie

Ocena
0

Lokalizacja
Łódz

Priorytet
Niski

Status
W trakcie

Wróć

Strona edycji zadania

Strona oceny zadania umożliwia wyłącznie zmianę oceny przyznanej zadaniu.

Tytuł zadania

Tytuł
Tytuł zadania

Opis
Opis opis opis

Ocena
0

Lokalizacja
Warsow

Priorytet
Krytyczny

Status
Zakończony

Wróć

Strona oceny zadania

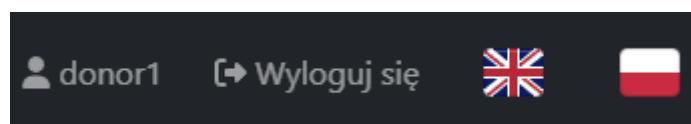
3.9 Użytkownik

3.9.1 Wypełnienie oraz edycja danych o profilu

W celu wypełnienia lub edycji danych o profilu należy nacisnąć przycisk „**Profil**” znajdujący się na stronie głównej.



Fragment widoku przycisków strony głównej dla zalogowanego użytkownika lub przycisk z nazwą użytkownika, który znajduje się na pasku nawigacyjnym.



Fragment widoku paska nawigacyjnego dla zalogowanego użytkownika
Po naciśnięciu jednego z tych przycisków nastąpi przeniesienie do panelu użytkownika, który zawiera dane na temat konta:

- Nazwa użytkownika,
- Email,
- Rola

oraz dane personalne:

→ Użytkownik

- Imię
- Nazwisko
- PESEL

→ Organizacja

- Nazwa

Dane konta

Nazwa użytkownika	Email	Rola
donor1	donor1@op.pl	Darczyńca

Dane personalne

Imię	Nazwisko	PESEL	Edytuj dane
Nie podano	Nie podano	Nie podano	

Część II: Dokumentacja użytkownika

Fragment widoku panelu użytkownika dla użytkownika z niewypełnionymi danymi.

Dane konta

Nazwa użytkownika	Email	Rola
organization2	organization2@op.pl	Organizacja

Dane organizacji

Nazwa organizacji	Edytuj dane
Nie podano	

Fragment widoku panelu użytkownika dla organizacji z niewypełnionymi danymi.

3.9.2 Wypełnienie danych użytkownika

W celu wypełnienia danych o użytkowniku należy nacisnąć przycisk „**edytuj dane**”. Po jego naciśnięciu pojawi się formularz do wypełnienia danych. Do którego należy wpisać dane personalne.

Dane personalne

Imię	Nazwisko	PESEL	Przejdź do edytowania
Nie podano	Nie podano	Nie podano	

Imię
Dominik

Nazwisko
Kowalski

PESEL:
111111111111

[Zapisz](#)

Fragment widoku formularza do wypełniania danych użytkownika.

Wszystkie pola muszą zostać wypełnione, a numer PESEL musi składać się dokładnie z jedenastu cyfr. W przeciwnym wypadku zostanie wyświetlony odpowiedni komunikat. Następnie konieczne jest zatwierdzenie danych poprzez naciśnięcie przycisku „**zapisz**”. Po poprawnym zapisaniu danych wyświetli się komunikat o powodzeniu przeprowadzenia operacji. W przypadku niepowodzenia zapisu danych zostanie wyświetlona informacja o błędzie. Po poprawnym zapisaniu, dane będą widoczne w panelu użytkownika i będzie możliwa ich edycja.

Dane konta

Nazwa użytkownika	Email	Rola
donor1	donor1@op.pl	Darczyńca

Dane personalne

Imię	Nazwisko	PESEL	Edytuj dane
Marek	Kowalski	11111111111	

Fragment widoku panelu użytkownik dla użytkownika z wypełnionymi danymi

3.9.3 Wypełnienie danych organizacji

W celu wypełnienia danych o organizacji należy nacisnąć przycisk „**edytuj dane**”. Po jego naciśnięciu pojawi się formularz do wypełnienia danych. Do którego należy wpisać dane organizacji.

Dane organizacji

Nazwa organizacji	<input type="text" value="Nie podano"/>	<button>Przestań edytować</button>
Nazwa organizacji	<input type="text" value="Organizacja2"/>	
Zapisz	<button>Zapisz</button>	

Fragment widoku formularza danych dla organizacji.

Wszystkie pola muszą zostać wypełnione. W przeciwnym wypadku zostanie wyświetlony odpowiedni komunikat. Następnie trzeba zatwierdzić dane naciskając przycisk „**zapisz**”. Po poprawnym zapisaniu danych powinien wyświetlić się komunikat o powodzeniu przeprowadzenia operacji. W przypadku niepowodzenia zapisu danych zostanie wyświetlona informacja o błędzie. Po poprawnym zapisaniu, dane będą widoczne w panelu użytkownika i będzie możliwa ich edycja.

Dane konta

Nazwa użytkownika

organization2

Email

organization2@op.pl

Rola

Organizacja

Dane organizacji

Nazwa organizacji

Organizacja2

Edytuj dane

Fragment widoku panelu użytkownik dla organizacji z wypełnionymi danymi

3.9.4 Edycja danych użytkownika

W celu edycji danych o użytkowniku należy nacisnąć przycisk „**edytuj dane**”. Po jego naciśnięciu pojawi się formularz do edycji danych. W którym należy zmienić dane.

Dane personalne

Imię

Dominik

Nazwisko

Kowalski

PESEL

111111111111

Przezana edytować

Imię

Marek

Nazwisko

Nowak

PESEL:

111111111111

Zapisz

Fragment widoku formularza danych dla użytkownika.

Nie można zmienić danych na puste oraz na numer PESEL, który nie zawiera dokładnie jedenastu cyfr. Jeżeli wystąpi taka sytuacja to zostanie wyświetlony odpowiedni komunikat. Następnie trzeba zatwierdzić dane naciskając przycisk „**zapisz**”. Po poprawnym zapisaniu danych powinien wyświetlić się komunikat. W przypadku niepowodzenia zapisu danych zostanie wyświetlona informacja o błędzie.

3.9.5 Edycja danych organizacji

W celu edycji danych o organizacji należy nacisnąć przycisk „**edytuj dane**”. Po jego naciśnięciu pojawi się formularz do edycji danych. W którym należy zmienić dane.

Dane organizacji

Nazwa organizacji

Organizacja2

Przestań edytować

Nazwa organizacji

Organizacja3

Zapisz

Fragment widoku formularza danych dla organizacji.

Nie można zmienić danych na puste. Jeżeli wystąpi taka sytuacja to zostanie wyświetlony odpowiedni komunikat. Następnie trzeba zatwierdzić dane naciskając przycisk „**zapisz**”. Po poprawnym zapisaniu danych powinien wyświetlić się komunikat. W przypadku niepowodzenia zapisu danych zostanie wyświetlona informacja o błędzie.

3.9.6 Przeglądanie organizacji

W celu przejścia do panelu dostępnych organizacji należy nacisnąć przycisk „**Organizacje**” znajdujący się na stronie głównej. Możliwość przeglądania organizacji jest dostępna tylko dla użytkowników z rolą „**Przedstawiciel władz**”.

Czat

Wyślij wiadomość do kogoś

Organizacje

Sprawdź dostępne organizacje

Fragment widoku przycisków strony głównej dla użytkownika z rola „**Przedstawiciel władz**”.

Po naciśnięciu przycisku nastąpi przeniesienie do panelu dostępnych organizacji, w którym wyświetlana jest lista wszystkich dostępnych organizacji zawierająca **ID organizacji** oraz **Nazwę organizacji**.

List organizacji

ID	Nazwa organizacji
1	Organizacja1
2	Organizacja2

Fragment widoku panelu dostępnych organizacji dla użytkownika z rola „**Przedstawiciel władz**”.

3.9.7 Aplikowanie do organizacji

W celu przejścia do panelu wniosków należy nacisnąć przycisk „**Wnioski**” znajdujący się na stronie głównej. Możliwość składania wniosków jest dostępna tylko dla użytkowników z rolą „**Wolontariusz**”.



Fragment widoku przycisków strony głównej dla użytkownika z rolą „**Wolontariusz**”.

Po naciśnięciu przycisku nastąpi przeniesienie do panelu wniosków, w którym wyświetlana jest lista wszystkich dostępnych organizacji zawierająca **ID organizacji**, **Nazwę organizacji**, **Status** oraz **Przycisk do wysłania aplikacji lub jej usunięcia**.

List organizacji

ID	Nazwa organizacji	Status	Akcja
1	Organizacja1		Wyślij wniosek
2	Organizacja2		Wyślij wniosek

Fragment widoku panelu wniosków dla użytkownika z rolą „**Wolontariusz**”.

W celu złożenia wniosku do wybranej organizacji należy nacisnąć przycisk „**Wyślij wniosek**” obok nazwy organizacji, do której wniosek ma być złożony. Po poprawnym złożeniu wniosku zostanie wyświetlony komunikat. W przypadku niepowodzenia zostanie wyświetlony komunikat o błędzie.

Po poprawnym wysłaniu wniosku przycisk zmieni się na przycisk pozwalający usunąć wniosek.

List organizacji

ID	Nazwa organizacji	Status	Akcja
1	Organizacja1	Usuń wniosek	
2	Organizacja2		Wyślij wniosek

Fragment widoku panelu wniosków z wysłanym wnioskiem

Jeżeli zmieni się status wniosku zostanie on wyświetlony. Możliwe są dwa stany wniosku **Zatwierdzony** lub **Odrzucony**.

List organizacji

ID	Nazwa organizacji	Status	Akcja
1	Organizacja1	Zatwierdzona	Usuń wniosek
2	Organizacja2		Wyślij wniosek

Fragment widoku panelu wniosków z wysłanym wnioskiem, który ma status **Zatwierdzony**

Część II: Dokumentacja użytkownika

List organizacji

ID	Nazwa organizacji	Status	Akcja
1	Organizacja1	Odrzucona	<input checked="" type="radio"/> Usuń wniosek
2	Organizacja2		<input type="button" value="Wyślij wniosek"/>

Fragment widoku panelu wniosków z wysłanym wnioskiem, który ma status **Odrzucony**

W celu usunięcia wniosku należy nacisnąć przycisk „**Usuń wniosek**” obok nazwy organizacji, do której wniosek został złożony. Po poprawnym złożeniu wniosku zostanie wyświetlony komunikat. W przypadku niepowodzenia zostanie wyświetlony komunikat o błędzie. Po poprawnym wysłaniu wniosku przycisk zmieni się na przycisk pozwalający wysłać wniosek.

3.9.8 Zarządzanie wnioskami

W celu przejścia do panelu wniosków należy nacisnąć przycisk „**Wnioski**” znajdujący się na stronie głównej. Możliwość zarządzania wnioskami jest dostępna tylko dla użytkowników z rolą „**Organizacja**”.



Fragment widoku przycisków strony głównej dla użytkownika z rolą „**Organizacja**”.

Po naciśnięciu przycisku nastąpi przeniesienie do panelu wniosków, w którym wyświetlana jest lista wszystkich złożonych wniosków do organizacji zawierająca **ID wolontariusza, Imię, Nazwisko, Status** oraz **Przycisk do wyświetlania dodatkowych informacji o wolontariuszu, zaakceptowania wniosku oraz odrzucenia**.

Lista aplikacji do organizacji

ID	Imię	Nazwisko	Status	Akcja
3	Jan	Kowalski	<input checked="" type="radio"/> Informacja <input checked="" type="checkbox"/> Zatwierdź <input checked="" type="radio"/> Odrzuć	
5	Mateusz	Nowak	<input checked="" type="radio"/> Informacja <input checked="" type="checkbox"/> Zatwierdź <input checked="" type="radio"/> Odrzuć	

Fragment widoku panelu wniosków dla użytkownika z rolą „**Organizacja**”.

3.9.9 Dodatkowe informacje o wolontariuszu

W celu wyświetlenia dodatkowych informacji o wolontariuszu należy nacisnąć przycisk „**Informacja**” obok wolontariusza, którego dodatkowe dane mają zostać wyświetlane. Po naciśnięciu przycisku wyświetli się panel dodatkowych informacji.

Dodatkowe informacje zawierają:

- Imię
- Nazwisko
- Pesel
- Email
- Nazwę użytkownika

Część II: Dokumentacja użytkownika

Informacja o wolontariuszu

Powrót

Imię	Jan
Nazwisko	Kowalski
Pesel	1111111111
Email	volunteer1@op.pl
Nazwa użytkownika	volunteer1

Fragment widoku panelu dodatkowych informacji o wolontariuszu

Aby powrócić do poprzedniego panelu należy nacisnąć przycisk „**Powrót**”

3.9.10 Akceptacja wniosku

W celu zaakceptowania wniosku należy nacisnąć przycisk „**Zatwierdź**” obok wolontariusza, którego wniosek ma być zaakceptowany. Po naciśnięciu przycisku wyświetli się komunikat o poprawnym zatwierdzeniu wniosku lub komunikat o błędzie. Po poprawnym zaakceptowaniu wniosku przyciski „**Zatwierdź**” oraz „**Odrzuć**” przestaną być dostępne oraz status zmieni się na **Zatwierdzony**.

Lista aplikacji do organizacji

ID	Imię	Nazwisko	Status	Akcja
3	Jan	Kowalski	Zatwierdzona	<input type="button" value="Informacja"/>
5	Mateusz	Nowak		<input type="button" value="Informacja"/> <input checked="" type="button" value="Zatwierdź"/> <input type="button" value="Odrzuć"/>

Fragment widoku panelu wniosków z zaakceptowanym wnioskiem

3.9.11 Odrzucenie wniosku

W celu odrzucenia wniosku należy nacisnąć przycisk „**Odrzuć**” obok wolontariusza, którego wniosek ma być odrzucony. Po naciśnięciu przycisku wyświetli się komunikat o poprawnym odrzuceniu wniosku lub komunikat o błędzie. Po poprawnym odrzuceniu wniosku przyciski **Zatwierdź** oraz **Odrzuć** przestaną być dostępne oraz status zmieni się na **Odrzucony**.

Lista aplikacji do organizacji

ID	Imię	Nazwisko	Status	Akcja
5	Mateusz	Nowak		<input type="button" value="Informacja"/> <input checked="" type="button" value="Zatwierdź"/> <input type="button" value="Odrzuć"/>
3	Jan	Kowalski	Odrzucona	<input type="button" value="Informacja"/>

Fragment widoku panelu wniosków z odrzuconym wnioskiem

3.9.12 Zarządzanie wolontariuszami

W celu przejścia do panelu wolontariuszy należy nacisnąć przycisk „**Wolontariusze**” znajdujący się na stronie głównej. Możliwość zarządzania wolontariuszami jest dostępna tylko dla użytkowników z rolą „**Organizacja**”.



Fragment widoku przycisków strony głównej dla użytkownika z rola „Organizacja”.

Po naciśnięciu przycisku nastąpi przeniesienie do panelu wolontariuszy, w którym wyświetlana jest lista wszystkich wolontariuszy w organizacji zawierająca **ID wolontariusza, Imię, Nazwisko, Status** oraz **Przycisk do wyświetlania dodatkowych informacji o wolontariuszu oraz usunięcia wolontariusza**.

3.9.13 Dodatkowe informacje o wolontariuszu

W celu wyświetlenia dodatkowych informacji o wolontariuszu należy nacisnąć przycisk „**Informacja**” obok wolontariusza, którego dodatkowe dane mają zostać wyświetcone. Po naciśnięciu przycisku wyświetli się panel dodatkowych informacji.

Dodatkowe informacje zawierają:

- Imię
- Nazwisko
- Pesel
- Email
- Nazwę użytkownika
- Ocena

The image shows a screenshot of a modal window titled "Informacja o wolontariuszu". In the top right corner is a "Powrót" (Return) button. The window contains five input fields: "Imię" (Name) with "Jan" entered; "Nazwisko" (Surname) with "Kowalski" entered; "Pesel" with "1111111111" entered; "Email" with "volunteer1@op.pl" entered; and "Nazwa użytkownika" (Username) with "volunteer1" entered. Each field has a small placeholder label above it.

Fragment widoku panelu dodatkowych informacji o wolontariuszu

Aby powrócić do poprzedniego panelu należy nacisnąć przycisk „**Powrót**”

3.9.14 Usunięcie wolontariusza

W celu usunięcia wolontariusza należy nacisnąć przycisk „**Usuń**” obok wolontariusza, który ma być usunięty. Po naciśnięciu przycisku wyświetli się komunikat o poprawnym usunięciu wolontariusza lub komunikat o błędzie. Po poprawnym usunięciu wolontariusza zniknie on z listy wolontariuszy będących w organizacji.

Lista wolontariuszy

ID	Imię	Nazwisko	Status	Akcja
1	Jan	Kowalski	Dostępny	Informacja - Usuń
2	Mateusz	Nowak	Dostępny	Informacja - Usuń

Fragment widoku panelu wolontariuszy przed usunięciem wolontariusza

Lista wolontariuszy

ID	Imię	Nazwisko	Status	Akcja
2	Mateusz	Nowak	Dostępny	Informacja - Usuń

Usunięto wolontariusza o imieniu i nazwisku: Jan Kowalski

Fragment widoku panelu wolontariuszy po usunięciu