

# **Dokumentacja projektu zaliczeniowego**

Przedmiot: Inżynieria oprogramowania

Temat: MediApp - System zarządzania dla przychodni lekarskiej  
Autorzy: **Maciej Dominiak, Nataniel Antosik**  
Grupa: I1-213B  
Kierunek: informatyka  
Rok akademicki: 2  
Poziom i semestr: I/4  
Tryb studiów: stacjonarne

# **1 Spis treści**

2 Odnośniki do innych źródeł	4
3 Słownik pojęć	5
4 Wprowadzenie	6
4.1 Cel dokumentacji	6
4.2 Przeznaczenie dokumentacji	6
4.3 Opis organizacji lub analiza rynku	6
4.4 Analiza SWOT organizacji	6
5 Specyfikacja wymagań	7
5.1 Charakterystyka ogólna	7
5.2 Wymagania funkcjonalne	7
5.3 Wymagania niefunkcjonalne	8
6 Zarządzanie projektem	9
6.1 Zasoby ludzkie	9
6.2 Harmonogram prac	9
6.3 Etapy/kamienie milowe projektu	9
7 Zarządzanie ryzykiem	10
7.1 Lista czynników ryzyka	10
7.2 Ocena ryzyka	10
7.3 Plan reakcji na ryzyko	10
8 Zarządzanie jakością	11
8.1 Scenariusze i przypadki testowe	11
9 Projekt techniczny	12
9.1 Opis architektury systemu	12
9.2 Technologie implementacji systemu	12
9.3 Diagramy UML	12
9.4 Charakterystyka zastosowanych wzorców projektowych	12
9.5 Projekt bazy danych	12
9.6 Projekt interfejsu użytkownika	12
9.7 Procedura wdrożenia	13
10 Dokumentacja dla użytkownika	14
11 Podsumowanie	15

11.1	Szczegółowe nakłady projektowe członków zespołu	15
12	Inne informacje	16

## **2 Odnośniki do innych źródeł**

- Wersjonowanie kodu - w tym celu została wykorzystana platforma Github  
[https://github.com/Seclare/IO\\_2020\\_44308\\_44260](https://github.com/Seclare/IO_2020_44308_44260)

### **3 Słownik pojęć**

Recepcjonista - osoba rejestrująca oraz obsługująca pacjentów.

Lekarz - osoba posiadająca wiedzę i uprawnienia do leczenia ludzi.

Pacjent - osoba korzystająca z usług placówki medycznej.

Pielegniarka - osoba sprawująca opiekę medyczną nad pacjentem.

Dyrektor placówki - osoba zarządzająca pracownikami ośrodka zdrowia.

Melcormed - przychodnia lekarska dla której tworzone jest oprogramowanie, w dokumentacji określana również jako placówka, ośrodek zdrowia.

MediApp - nazwa projektowanego systemu zarządzania, w dokumentacji określane również jako aplikacja, oprogramowanie, system.

PACS-system informatyczny do archiwizacji i transmisji obrazów z urządzeń diagnostycznych.

RIS - Radiologiczny System Informatyczny, pozwalający na obsługę całego procesu realizacji badania od momentu planowania i rejestracji pacjenta w pracowni radiologicznej, przez proces wykonania badania, aż po stworzenie i wydanie opisu.

System P1 - Elektroniczna Platforma Gromadzenia, Analizy i Udostępniania Zasobów Cyfrowych o Zdarzeniach Medycznych, pozwalająca na wypisanie elektronicznej recepty.

## **4 Wprowadzenie**

### ***4.1 Cel dokumentacji***

Celem dokumentacji jest zapoznanie czytelnika ze specyfikacją wdrażanego systemu. Pozwoli to na łatwiejsze zrozumienie implementowanego systemu jak i sprawniejsze opracowywanie projektu, poprzez jasno wyznaczone wymagania oraz cele.

### ***4.2 Przeznaczenie dokumentacji***

Dokumentacja przeznaczona jest dla osób zaangażowanych w projektowanie wdrażanego systemu.

### ***4.3 Opis organizacji lub analiza rynku***

Organizacją docelową jest przychodnia lekarska "Melcormed", znajdująca się w Szczecinie przy ulicy Szarych Szeregów 48. Swoją działalność prowadzi od stycznia 2016 roku. "Melcormed" ma podpisany kontrakt z Narodowym Funduszem Zdrowia na leczenie w Podstawowej oraz Ambulatoryjnej Opiece Zdrowotnej. Dlatego wszelkie usługi w ramach POZ jak i AOZ dla pacjentów ubezpieczonych i zadeklarowanych w danej przychodni realizowane są bezpłatnie. Ośrodek zdrowia przyjmuje wyłącznie osoby zarejestrowane. Osoby, które nie są pacjentami placówki, mogą wypełnić deklarację, następnie przynieść ją do przychodni i zapisać się na wizytę. Zapisanie na wizytę polega na przyjściu do placówki "Melcormed" oraz ustaleniu terminu wizyty u wybranego lekarza z recepcjonistą. Po dokonaniu zapisu pacjent otrzymuje kartkę, na której możemy znaleźć informacje dotyczące daty oraz godziny umówionej wizyty jak i numer pacjenta w kolejce do danego lekarza. Ośrodek zdrowia posiada dokumentacje 1264 zarejestrowanych pacjentów. Dziennie z usług przychodni korzysta około 26 osób, zaś sama placówka otwarta jest codziennie w godzinach od 8:00 do 16:00. Organizacja zatrudnia 20 pracowników, przy czym połowa pracuje na pół etatu. W skład personelu wchodzi 4 lekarzy podstawowej opieki zdrowotnej, 4 pielęgniarki, 2 recepcjonistki, 2 laryngologów, 2 ginekologów, ortopeda, alergolog, diabetolog, kardiolog oraz 2 chirurgów. Do usług oferowanych przez daną placówkę należy: rejestracja klientów, wydawanie recept, umawianie wizyt u poszczególnych lekarzy placówki (do lekarzy specjalistów na podstawie skierowań), wypisywanie skierowań do lekarzy z placówki "Melcormed" oraz wypisywanie skierowań na zabiegi, które są dokonywane w danej placówce jak i poza nią. Badania laboratoryjne oraz diagnostyka obrazowa wykonywana jest przez zewnętrzne firmy, na podstawie skierowań, wypisanych przez lekarzy "Melcormed". Oryginał wyników badań otrzymuje pacjent, zaś kopia wyników zostaje przesłana pocztą do placówki a następnie przechowywana w dokumentacji pacjenta.

#### **4.4 Analiza SWOT organizacji**

Mocne strony	Słabe strony
-kadra składająca się z młodych osób pragnących nabyc doświadczenia -możliwość uzyskania pomocy od wykwalifikowanych osób -mała rotacja pracowników	-brak poprzedniego doświadczenia pracowników -słaba organizacja pracy skutkująca przeciążeniem pracowników
Szanse	Zagrożenia
-personel na wysokim zaawansowaniu technologicznym -zwiększenie wydajności placówki -przyspieszenie wykonywania oferowanych usług	-brak chęci użytkowników do przystosowania się do nowego trybu pracy -konieczność przeszkolenia pracowników ośrodka zdrowia

### **5 Specyfikacja wymagań**

#### **5.1 Charakterystyka ogólna**

##### **5.1.1 Definicja produktu**

“MediApp” jest systemem programistycznym, którego celem jest wspomaganie dotychczasowej działalności przychodni.

##### **5.1.2 Podstawowe założenia**

Wdrażany system “MediApp” będzie usprawniał pracę w placówce medycznej “Melcormed”, poprzez przeniesienie wszystkich danych zapisanych w formie papierowej do nowo utworzonej bazy danych. Umożliwi to łatwiejszy dostęp do historii chorób lekarzowi co przyspieszy czas ich obsługi. Będzie możliwe wypisywanie recept w formie tradycyjnej oraz dzięki naszemu systemowi, wygenerowanie unikalnego kodu na daną receptę dla klienta, dzięki temu dana osoba nie będzie musiała przychodzić do placówki. Jeżeli lekarz zaleci pacjentowi badania w szpitalu może w łatwy sposób wysłać odpowiednie informacje do innej placówki wraz z odpowiednimi danymi pacjenta.

##### **5.1.3 Cel biznesowy**

Przychodnia lekarska “Melcormed”, wdrażając opisywany system, ma na celu usprawnienie funkcjonowania placówki, poprzez wykorzystanie systemu informatycznego. Pozwoli on na automatyzację poszczególnych czynności oraz ich cyfryzację, co zwiększy wydajność placówki.

#### **5.1.4 Użytkownicy**

- Recepcjonista
- Lekarz
- Pielęgniarka
- Dyrektor placówki
- Pacjent

#### **5.1.5 Korzyści z systemu**

##### **1.Recepcjonista**

- a)Łatwiejszy i szybszy dostęp do harmonogramu pracy lekarzy
- b)Szybsza rejestracja pacjentów
- c)Elektroniczna lista zarejestrowanych pacjentów
- d)Szybsza i łatwiejsza możliwość zmiany danych pracownika lub pacjenta.

##### **2.Lekarz**

- a)Łatwiejszy i szybszy dostęp do własnego harmonogramu pracy
- b)Elektroniczny dostęp do historii chorób pacjenta

##### **3.Pielęgniarka**

- a)Łatwiejszy i szybszy dostęp do własnego harmonogramu pracy
- b)Elektroniczny dostęp do historii chorób pacjenta

##### **4.Dyrektor placówki**

- a)Łatwiejsze zarządzanie pracownikami
- b)Łatwiejszy i szybszy dostęp do harmonogramu pracy pracowników

##### **5.Pacjent**

- a)Elektroniczny dostęp do historii przepisanych leków
- b)Elektroniczny dostęp do wyników badań
- c)Elektroniczna recepta.
- d)Elektroniczne skierowanie.

#### **5.1.6 Ograniczenia projektowe i wdrożeniowe**

1.Klient wymaga, aby możliwe było obsługiwanie oprogramowania z poziomu strony internetowej, zarówno poprzez stanowisko komputerowe, jak i telefon komórkowy.

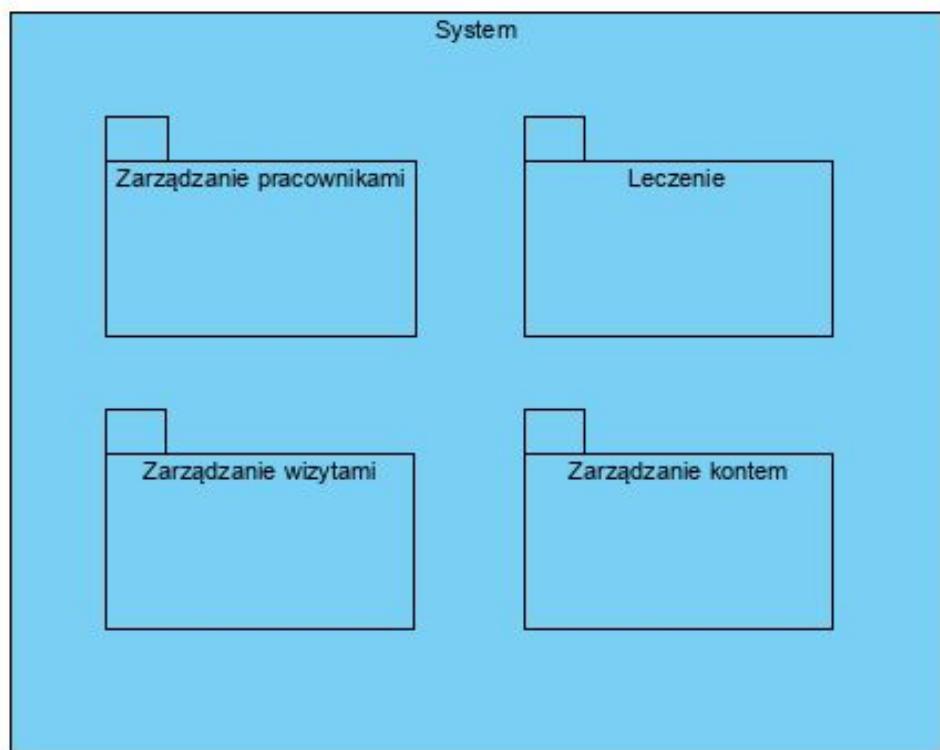
2.Klient dysponuje serwerem, na którym ma pracować oprogramowanie MediApp. Jednostka serwerowa posiada 32 GB RAM-u oraz 1TB pamięci zewnętrznej.

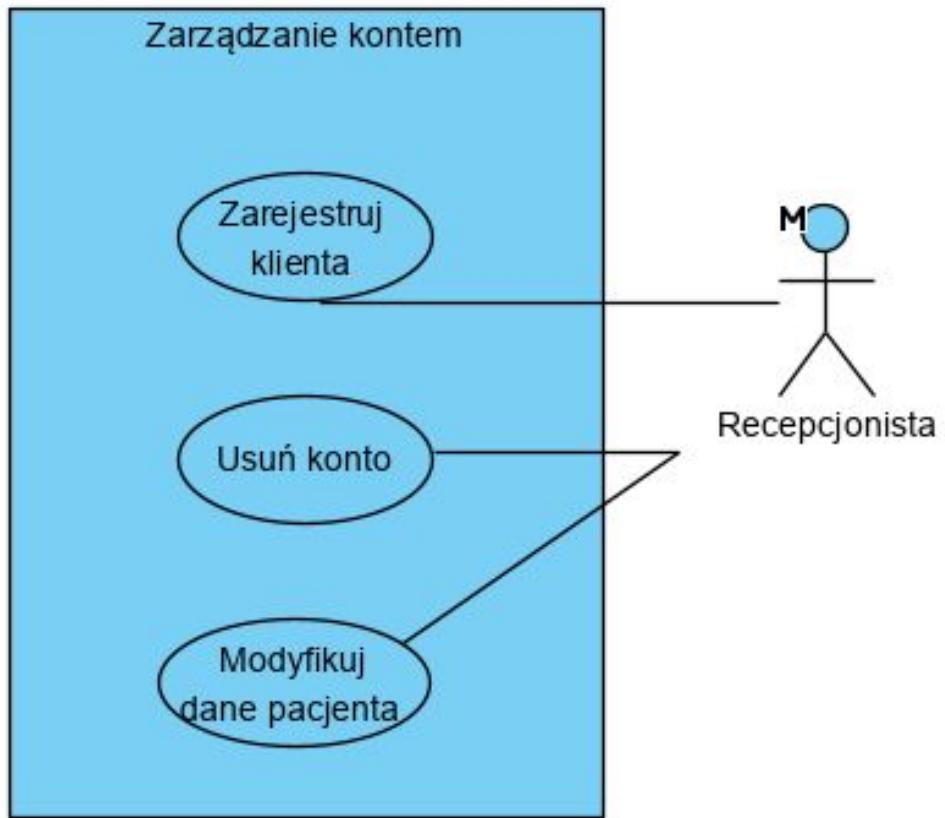
## **5.2 Wymagania funkcjonalne**

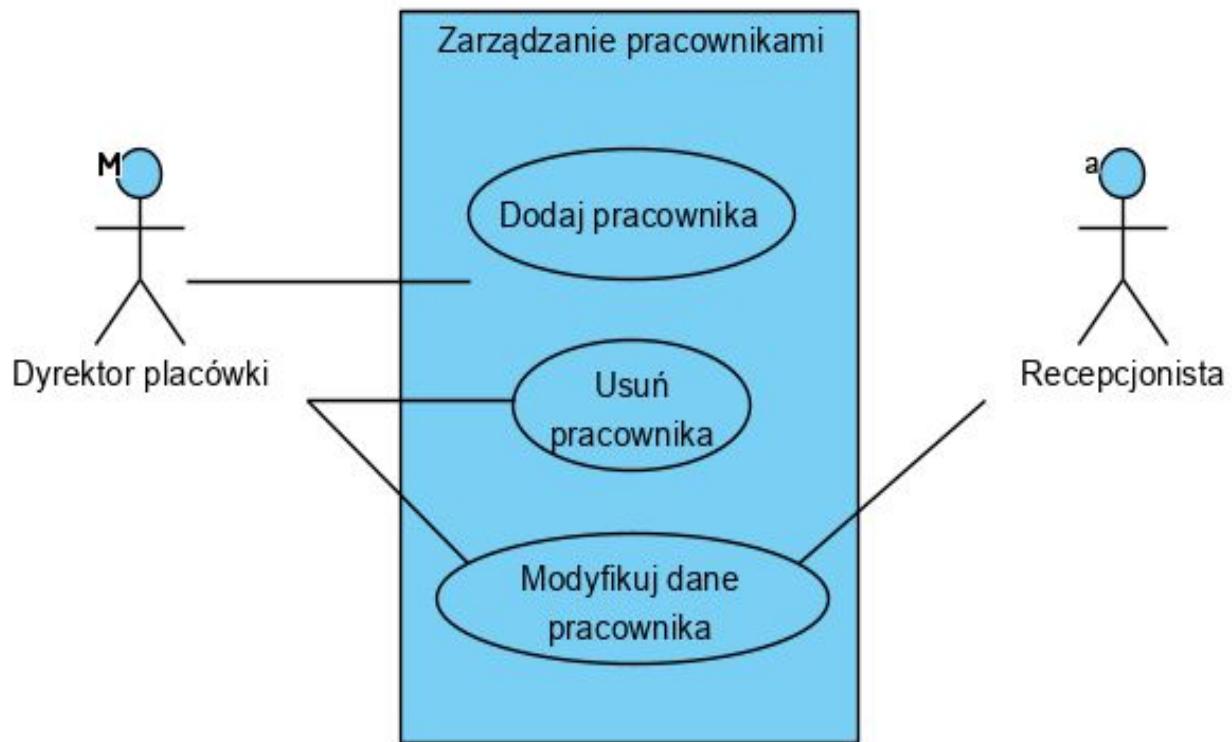
### **5.2.1 Lista wymagań**

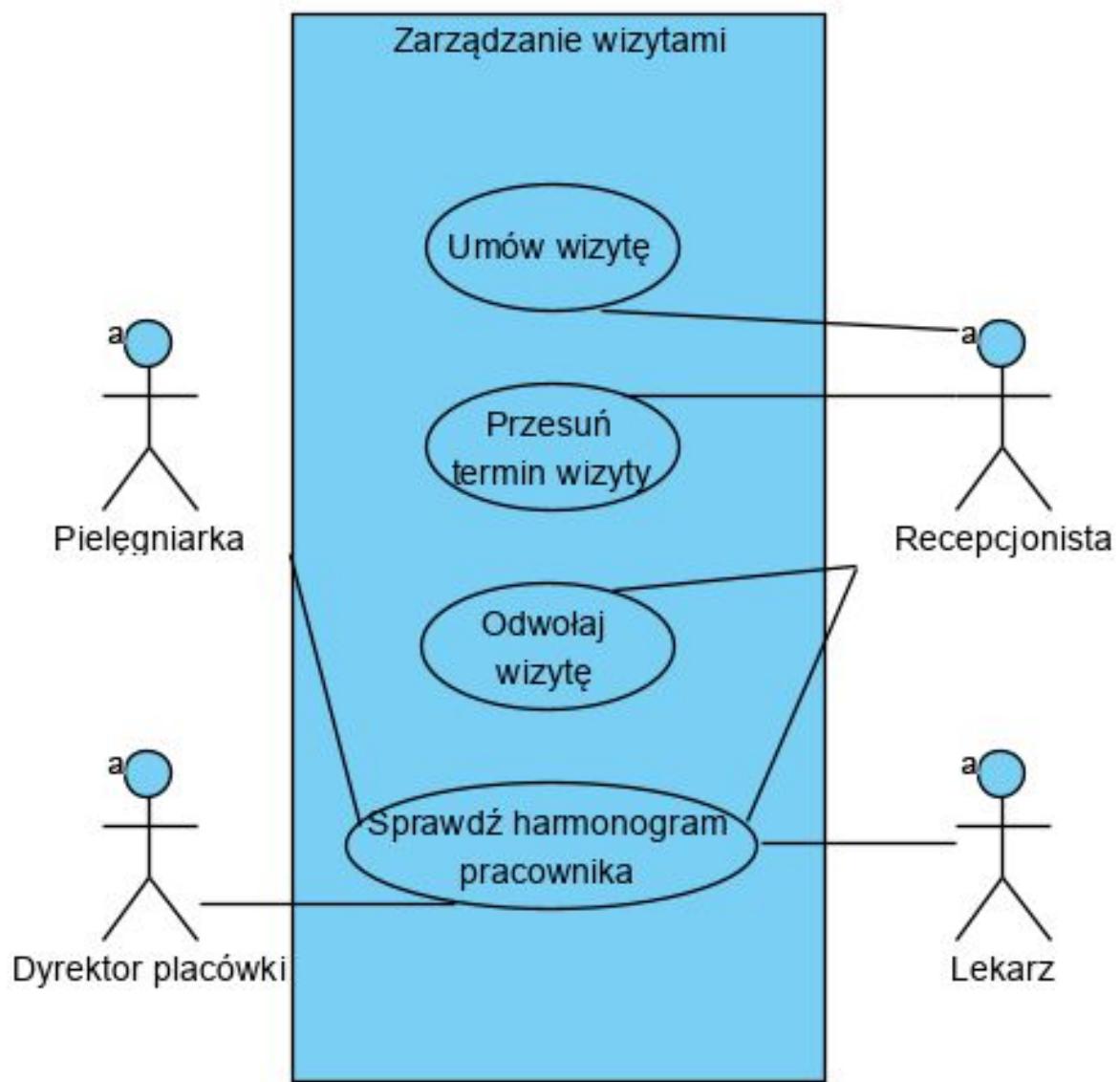
1. System ma umożliwić rejestrację klienta w bazie danych.
2. System ma umożliwić modyfikację danych klienta.
3. System ma umożliwić usunięcie klienta z bazy danych.
4. System ma umożliwić wgląd w historię chorób pacjenta.
5. System ma umożliwić wpis w historię chorób pacjenta.
6. System ma umożliwić na elektroniczne wypisanie recepty.
7. System ma umożliwić umówienie wizyty u lekarza.
8. System ma umożliwić przesunięcie terminu wizyty jak i również jej odwołanie.
9. System ma umożliwić zarządzanie harmonogramem pracy personelu.
10. System ma umożliwić rejestrację pracownika w bazie danych.
11. System ma umożliwić modyfikację danych pracownika.
12. System ma umożliwić usunięcie pracownika z bazy danych.
13. System ma umożliwić wgląd w wyniki badań.
14. System ma umożliwić wysyłanie powiadomień e-mail oraz sms
15. System ma umożliwić wypisanie elektronicznego skierowania na badania.
16. System ma umożliwić dostęp do historycznych wyników badań pacjenta.
17. System ma umożliwić wgląd w harmonogram pracy personelu.

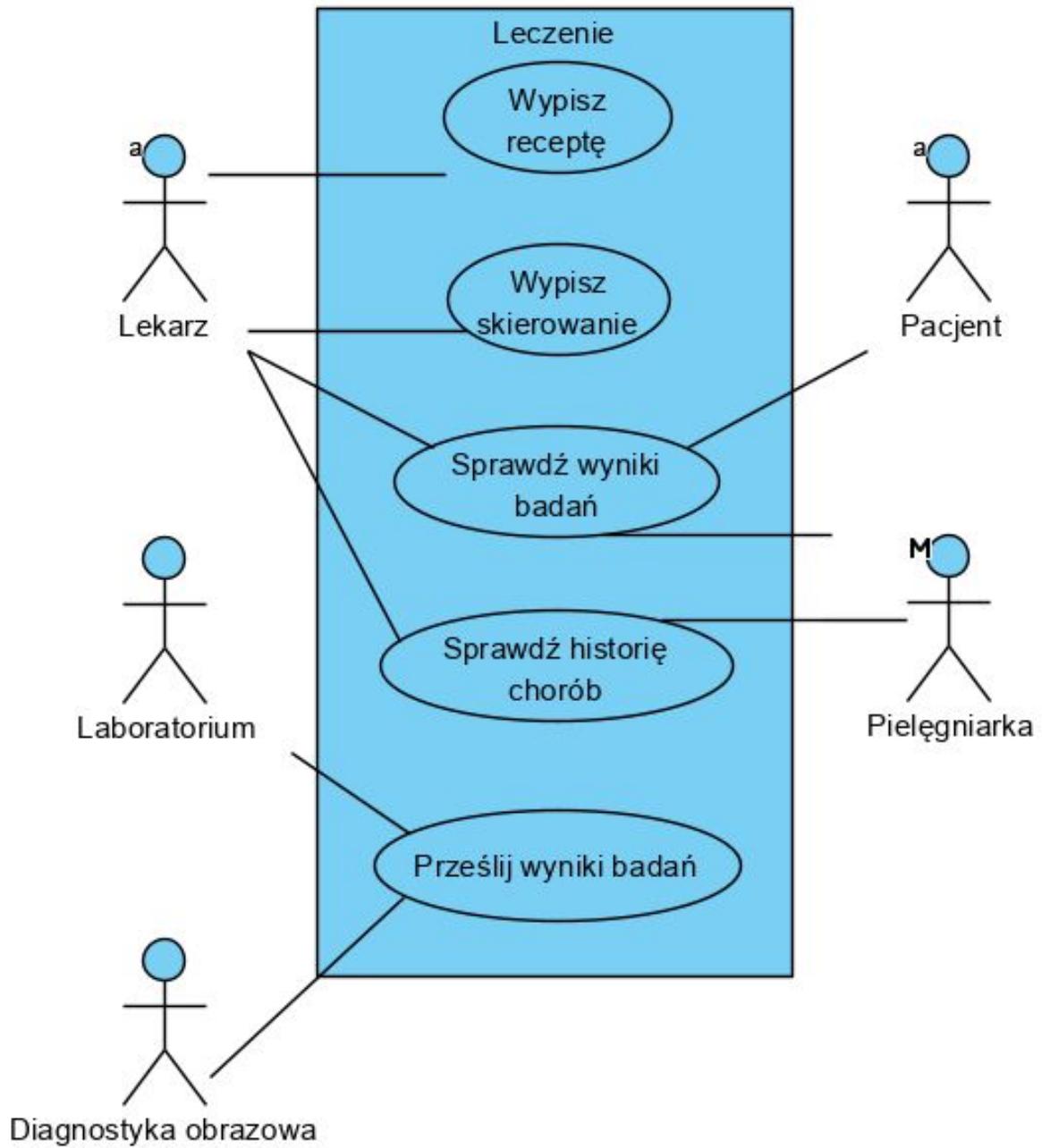
### 5.2.2 Diagramy przypadków użycia











## **5.2.3 Szczegółowy opis wymagań**

### **1. Rejestracja pacjenta**

#### **1.1. Uzasadnienie biznesowe**

1b

#### **1.2. Użytkownicy**

Recepcjonista

#### **1.3. Scenariusz**

##### **1.3.1. Warunki początkowe**

Osoba, która ma być dodana do bazy danych, nie może istnieć już w bazie danych.

##### **1.3.2. Przebieg działań**

1. Recepcjonista wybiera opcję dodania nowego pacjenta.
2. System wyświetla formularz do rejestracji pacjenta.
3. Recepcjonista wypełnia formularz danymi podanymi przez pacjenta (tj. imię, nazwisko, adres zamieszkania, kod pocztowy, numer telefonu, e mail, dane osobowe osoby upoważnionej do odbioru wyników).
4. Recepcjonista zatwierdza wykonanie operacji.
5. System sprawdza, czy obowiązkowe pola zostały wypełnione.
6. System sprawdza czy dany pacjent nie istnieje już w bazie danych.
7. System dodaje profil pacjenta do bazy danych.
8. System generuje unikalny kod dla pacjenta, który znajduje się również w profilu pacjenta, do wglądu lekarza.

##### **1.3.3. Warunki końcowe**

1. Pacjent zostaje dodany do bazy danych

#### **1.4. Scenariusz negatywny - duplikat pacjenta**

##### **1.4.1. Warunki początkowe**

Osoba, która ma zostać dodana do bazy danych, znajduje się już w bazie danych.

##### **1.4.2. Przebieg działań**

1. Recepcjonista wybiera opcję dodania nowego pacjenta.
2. System wyświetla formularz do rejestracji pacjenta.
3. Recepcjonista wypełnia formularz danymi podanymi przez pacjenta (tj. imię, nazwisko, adres zamieszkania, kod pocztowy, numer telefonu, e mail, dane osobowe osoby upoważnionej do odbioru wyników).
4. Recepcjonista zatwierdza wykonanie operacji.
5. System sprawdza, czy obowiązkowe pola zostały wypełnione.
6. System sprawdza czy dany pacjent nie istnieje już w bazie danych.
7. System wyświetla komunikat, że dany pacjent już znajduje się w bazie danych.

#### **1.4.3. Warunki końcowe**

1. Pacjent nie zostaje ponownie dodany do systemu.

#### **1.5. Wymagania niefunkcjonalne**

Brak

#### **1.6. Częstotliwość i istotność**

Po uruchomieniu systemu częstotliwość korzystania z tej funkcji będzie na poziomie 5, z czasem zredukuje się ona do poziomu 3. Istotność - 5.

## **2. Wyświetl historię chorób pacjenta**

### **2.1. Uzasadnienie biznesowe**

2b,3b

### **2.2. Użytkownicy**

(Lekarz, pielęgniarka) jako medyk

### **2.3. Scenariusz**

#### **2.3.1. Warunki początkowe**

Pacjent, którego historia chorób jest sprawdzana, musi istnieć w bazie danych.

#### **2.3.2. Przebieg działań**

1. Medyk wybiera opcję wyszukania profilu pacjenta.
2. System wyświetla okno wyszukiwania pacjenta.
3. Medyk wpisuje dane szukanego pacjenta.
4. System wyświetla wyniki pasujące do wprowadzanych danych.
5. Medyk wybiera profil szukanego pacjenta.
6. System wyświetla profil szukanego pacjenta.
7. Medyk wybiera opcję wyświetlenia historii chorób pacjenta.
8. System wyświetla historię chorób pacjenta.

#### **2.3.3. Warunki końcowe**

1. Historia chorób pacjenta zostaje wyświetlona.

### **2.4. Scenariusz negatywny - brak pacjenta w bazie danych**

#### **2.4.1. Warunki początkowe**

Pacjent, którego historia chorób jest sprawdzana, nie istnieje w bazie danych.

#### **2.4.2. Przebieg działań**

1. Medyk wybiera opcję wyszukania profilu pacjenta.
2. System wyświetla okno wyszukiwania pacjenta.
3. Medyk wpisuje dane szukanego pacjenta.
4. System wyświetla komunikat, że osoba o wprowadzonych danych nie istnieje w bazie danych

#### **2.4.3. Warunki końcowe**

1. Brak możliwości wyświetlenia historii chorób pacjenta.

### **2.5. Wymagania niefunkcjonalne**

Brak

### **2.6. Częstotliwość i istotność**

Częstotliwość - 5

Istotność -5

### **3. Wypisanie elektronicznej recepty**

#### **3.1. Uzasadnienie biznesowe**

5d

#### **3.2. Użytkownicy**

Lekarz

#### **3.3. Scenariusz**

##### **3.3.1. Warunki początkowe**

Pacjent, dla którego wypisywana będzie recepta, musi znajdować się w bazie danych.

##### **3.3.2. Przebieg działań**

1. Lekarz wybiera opcję wypisania recepty.
2. System wyświetla odpowiedni formularz do wypisania recepty.
3. Lekarz uzupełnia formularz odpowiednimi danymi (tj. imię i nazwisko osoby otrzymującej receptę, nazwę leku, dawkowanie).
4. Lekarz zatwierdza wykonanie operacji.
5. System sprawdza, czy obowiązkowe pola zostały wypełnione.
6. System przesyła dane do systemu P1.
7. System MediApp otrzymuje informację zwrotną od systemu P1 w postaci wydruku informacyjnego z kodem.

##### **3.3.3. Warunki końcowe**

1. Wypisanie elektronicznej recepty.

#### **3.4. Scenariusz negatywny - niewypełnione pola**

##### **3.4.1. Warunki początkowe**

Lekarz nie wypełnia obowiązkowych pól w formularzu.

##### **3.4.2. Przebieg działań**

1. Lekarz wybiera opcję wypisania recepty.
2. System wyświetla odpowiedni formularz do wypisania recepty.
3. Lekarz uzupełnia formularz odpowiednimi danymi (tj. imię i nazwisko osoby otrzymującej receptę, nazwę leku, dawkowanie).
4. Lekarz zatwierdza wykonanie operacji.
5. System sprawdza, czy obowiązkowe pola zostały wypełnione.
6. System informuje lekarza, że nie wprowadzono wszystkich obowiązkowych pól.

##### **3.4.3. Warunki końcowe**

1. Recepta nie zostaje wypisana.

#### **3.5. Wymagania niefunkcjonalne**

Brak

#### **3.6. Częstotliwość i istotność**

Częstotliwość - 5

Istotność -5

## **4. Umówienie wizyty u lekarza**

### **4.1. Uzasadnienie biznesowe**

1a

### **4.2. Użytkownicy**

Recepcjonista

### **4.3. Scenariusz**

#### **4.3.1. Warunki początkowe**

Pacjent, który umawia wizytę u lekarza, znajduje się w bazie danych.

#### **4.3.2. Przebieg działań**

1. Recepcjonista wybiera opcję wyszukania lekarza.
2. System wyświetla okno wyszukiwania lekarza.
3. Recepcjonista wpisuje dane szukanego lekarza.
4. System wyświetla wyniki pasujące do wprowadzonych danych.
5. Recepcjonista wybiera profil szukanego lekarza.
6. System wyświetla profil szukanego lekarza.
7. Recepcjonista wybiera opcję wyświetlania harmonogramu pracy danego lekarza.
8. System wyświetla harmonogram pracy danego lekarza.
9. System sprawdza czy dany lekarz wymaga skierowania.
10. System informuje, że nie potrzeba skierowania do danego lekarza.
11. Recepcjonista wybiera w oknie wizyty opcję wyszukania pacjenta.
12. System wyświetla okno wyszukiwania pacjenta
13. Recepcjonista wpisuje dane pacjenta
14. System wyświetla wyniki pasujące do wprowadzonych danych.
15. Recepcjonista wybiera profil szukanego pacjenta.
16. Recepcjonista wybiera termin wizyty.
17. System sprawdza czy dany termin wizyty nie jest zajęty.
18. Recepcjonista dodaje dane pacjenta do danego terminu wizyty, tj. jego profil z bazy danych.
19. Recepcjonista zatwierdza wykonanie operacji.
20. System zapisuje wizytę w danym terminie oraz ustawia przypomnienie sms oraz e mail na dzień przed wizytą.

#### **4.3.3. Warunki końcowe**

1. Wizyta zostaje zapisana na wybrany termin.

### **4.4. Scenariusz negatywny - brak pacjenta w bazie danych**

#### **4.4.1. Warunki początkowe**

Pacjent, który umawia wizytę u lekarza, znajduje się w bazie danych.

#### **4.4.2. Przebieg działań**

1. Recepjonista wybiera opcję wyszukania lekarza.
2. System wyświetla okno wyszukiwania lekarza.
3. Recepjonista wpisuje dane szukanego lekarza.
4. System wyświetla wyniki pasujące do wprowadzonych danych.
5. Recepjonista wybiera profil szukanego lekarza.
6. System wyświetla profil szukanego lekarza.
7. Recepjonista wybiera opcję wyświetlania harmonogramu pracy danego lekarza.
8. System wyświetla harmonogram pracy danego lekarza.
9. System sprawdza czy dany lekarz wymaga skierowania.
10. System informuje, że nie potrzeba skierowania do danego lekarza.
11. Recepjonista wybiera w oknie wizyty opcję wyszukania pacjenta.
12. System wyświetla okno wyszukiwania pacjenta.
13. Recepjonista wpisuje dane pacjenta
15. System wyświetla komunikat, że osoba o wprowadzonych danych nie istnieje w bazie danych

#### **4.4.3. Warunki końcowe**

1. Wizyta nie zostaje zapisana.

#### **4.5. Wymagania niefunkcjonalne**

Brak

#### **4.6. Częstotliwość i istotność**

Częstotliwość - 5

Istotność -5

## **5. Wyświetl historię wyników badań**

### **5.1. Uzasadnienie biznesowe**

5c

### **5.2. Użytkownicy**

Lekarz, pacjent.

### **5.3. Scenariusz główny dla lekarza**

#### **5.3.1. Warunki początkowe**

Pacjent, którego historię wyników badań chcemy wyświetlić, znajduje się w bazie danych.

#### **5.3.2. Przebieg działań**

1. Lekarz wybiera opcja wyszukania profilu pacjenta.
2. System wyświetla okno wyszukiwania pacjenta.
3. Lekarz wpisuje dane szukanego pacjenta.
4. System wyświetla wyniki pasujące do wprowadzonych danych.
5. Lekarz wybiera profil szukanego pacjenta
6. System wyświetla profil szukanego pacjenta.
7. Lekarz wybiera opcję wyświetlenia historii wyników badań pacjenta.
8. System wyświetla historię wyników badań pacjenta.

#### **5.3.3. Warunki końcowe**

1. Historia wyników badań zostaje wyświetlona.

### **5.4. Scenariusz główny dla pacjenta**

#### **5.4.1. Warunki początkowe**

Pacjent, który chce sprawdzić historię wyników swoich badań, musi istnieć w bazie danych.

#### **5.4.2. Przebieg działań**

1. Pacjent wybiera na stronie zakładkę wyniki badań.
2. System wyświetla okno z polem na kod pacjenta
3. Pacjent wpisuje swój kod w wyznaczone pole
4. System sprawdza czy pacjent o wprowadzonym kodzie istnieje w bazie danych.
5. System wyświetla historię wyników badań pacjenta.

#### **5.4.3. Warunki końcowe**

1. Historia wyników badań zostaje wyświetlona.

### **5.5. Scenariusz negatywny dla lekarza - brak pacjenta w bazie danych**

#### **5.5.1. Warunki początkowe**

Pacjent, którego historia wyników badań ma zostać wyświetlona, nie istnieje w bazie danych.

#### **5.5.2. Przebieg działań**

1. Lekarz wybiera opcja wyszukania profilu pacjenta.
2. System wyświetla okno wyszukiwania pacjenta.

3. Lekarz wpisuje dane szukanego pacjenta.
4. System wyświetla komunikat, że osoba o wprowadzonych danych nie istnieje w bazie danych

#### **5.5.3. Warunki końcowe**

1. System nie wyświetla wyników badań.

### **5.6. Scenariusz negatywny dla lekarza - brak pacjenta w bazie danych**

#### **5.6.1. Warunki początkowe**

Pacjent, który chce wyświetlić swoją historię wyników badań, nie posiada profilu w bazie danych.

#### **5.5.2. Przebieg działań**

1. Pacjent wybiera na stronie zakładkę wyniki badań.
2. System wyświetla okno z polem na kod pacjenta
3. Pacjent wpisuje swój kod w wyznaczone pole
4. System sprawdza czy pacjent o wprowadzonym kodzie istnieje w bazie danych.
5. System wyświetla komunikat, że pacjent o wprowadzonym kodzie nie istnieje w bazie danych.

#### **5.5.3. Warunki końcowe**

1. System nie wyświetla historii wyników badań

### **5.5. Wymagania niefunkcjonalne**

Brak

### **5.6. Częstotliwość i istotność**

Częstotliwość - 3

Istotność - 4

### **5.3 Wymagania niefunkcjonalne**

1. Wydajność:
  - a) Wszelkie operacje wykonywane na systemie przez pracowników ośrodka zdrowia mają wykonywać się w czasie nie większym niż 1 sekunda.
2. Pojemność:
  - a) System ma mieć możliwość przechowywania danych sześciu tysięcy pacjentów.
3. Bezpieczeństwo:
  - a) dane przechowywane przez system mają mieć swoją kopię zapasową, przechowywaną w chmurze oraz na lokalnym dysku twardym. Mają się one aktualizować co godzinę. W przypadku awarii lub na żądanie klienta, dane mają zostać odtworzone z kopii zapasowej.
4. Zabezpieczenia:
  - a) Należy zastosować szyfrowaną transmisję
  - b) System powinien być zabezpieczony przed atakami DDOS.
5. Inne cechy jakości:
  - a) system ma być przystosowany do zewnętrznych programów RIS i PACS.

## **6 Zarządzanie projektem**

### **6.1 Zasoby ludzkie**

(rzeczywiste lub hipotetyczne) – przy realizacji projektu

Należy założyć, że projekt byłby realizowany w całości jako projekt komercyjny a nie tylko częściowo w ramach zajęć na uczelni

### **6.2 Harmonogram prac**

Etapy mogą się składać z zadań.

Wskazać czasy trwania poszczególnych etapów i zadań – wykres Gantta.

obejmuje również harmonogram wdrożenia projektu – np. szkolenie, rozruch, konfiguracja, serwis – może obejmować różne wydania (tj. o różnej funkcjonalności – personal, professional, enterprise) i wersje (1.0, 1.5, itd.)

### **6.3 Etapy/kamienie milowe projektu**

dla głównych etapów projektu

## 7 Zarządzanie ryzykiem

### 7.1 Lista czynników ryzyka

Wypełniona lista kontrolna

### 7.2 Ocena ryzyka

prawdopodobieństwo i wpływ

### 7.3 Plan reakcji na ryzyko

Działania w odniesieniu do poszczególnych ryzyk.

Mogą być wg różnych strategii, tj. kilka strategii dla pojedynczego czynnika ryzyka

## 8 Zarządzanie jakością

### 8.1 Scenariusze i przypadki testowe

szczegółowy plan testowania systemu – głównie testowanie funkcjonalności; każdy scenariusz od nowej strony, musi zawierać co najmniej następujące informacje (sugerowany układ tabelaryczny, np. wg szablonu podanego w osobnym pliku lub na wykładzie):

- numer – jako ID
- nazwa scenariusza – co test w nim testowane (max kilka wyrazów)
- kategoria – poziom/kategoria testów
- opis – dodatkowe opcjonalne informacje, które nie zmieściły się w nazwie
- tester - konkretna osoba lub klient/pracownik,
- termin – kiedy testowanie ma być przeprowadzane,
- narzędzia wspomagające – jeśli jakieś są używane przy danym scenariuszu
- przebieg działań – tabela z trzema kolumnami: lp. oraz opisującymi działania testera i systemu
- założenia, środowisko, warunki wstępne, dane wejściowe – przygotowanie przed uruchomieniem testów
- zestaw danych testowych – najlepiej w formie tabelarycznej – jakie konkretnie dane mają być użyte przez testera i zwrócone przez system w poszczególnych krokach przebiegu działań
  - o *przebieg lub zestaw danych testowych musi zawierać jawną informację o warunku zaliczenia testu*

## 9 Projekt techniczny

### 9.1 Opis architektury systemu

System będzie składał się z następujących składowych:

1. Podsystem Zarządzanie Kontem - klasy z tego pakietu umożliwiają tworzenie instancji klienta, modyfikacji danych oraz usuwanie ich.
2. Podsystem Zarządzanie Pracownikami - klasy z tego pakietu umożliwiają tworzenie instancji pracownika, modyfikacji danych oraz usuwanie ich.
3. Podsystem Wizytami - klasy z tego pakietu umożliwiają tworzenie instancji wizyty, przesunięcie i odwołanie danego terminu oraz sprawdzenie harmonogramu pracownika.
4. Podsystem Leczenia - klasy z tego pakietu umożliwiają tworzenie instancji recepty oraz skierowania. Umożliwia też sprawdzić wyniki badań, historię chorób oraz wyników badań.

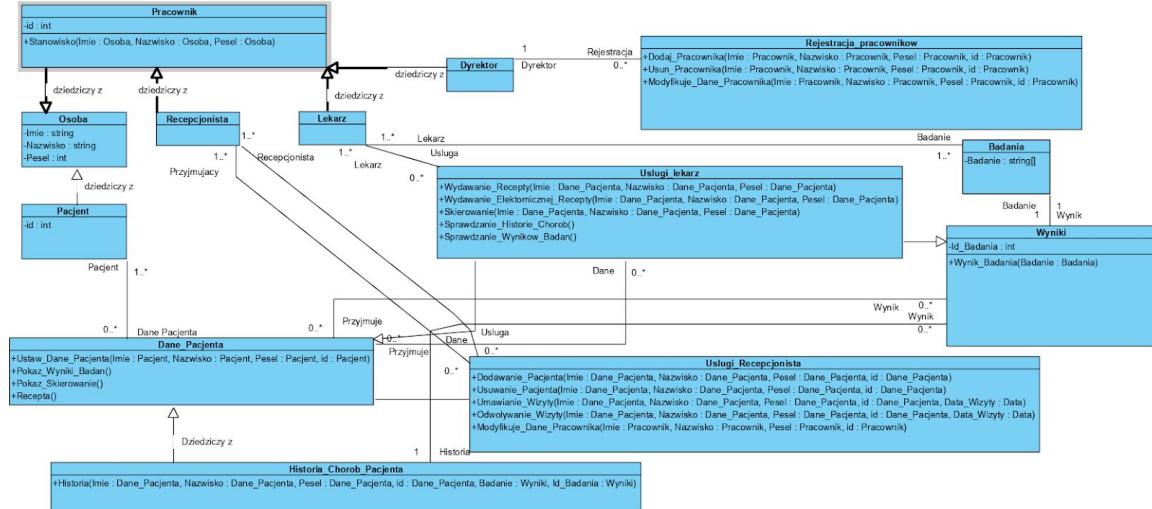
### 9.2 Technologie implementacji systemu

tabela z listą wykorzystanych technologii, każda z uzasadnieniem

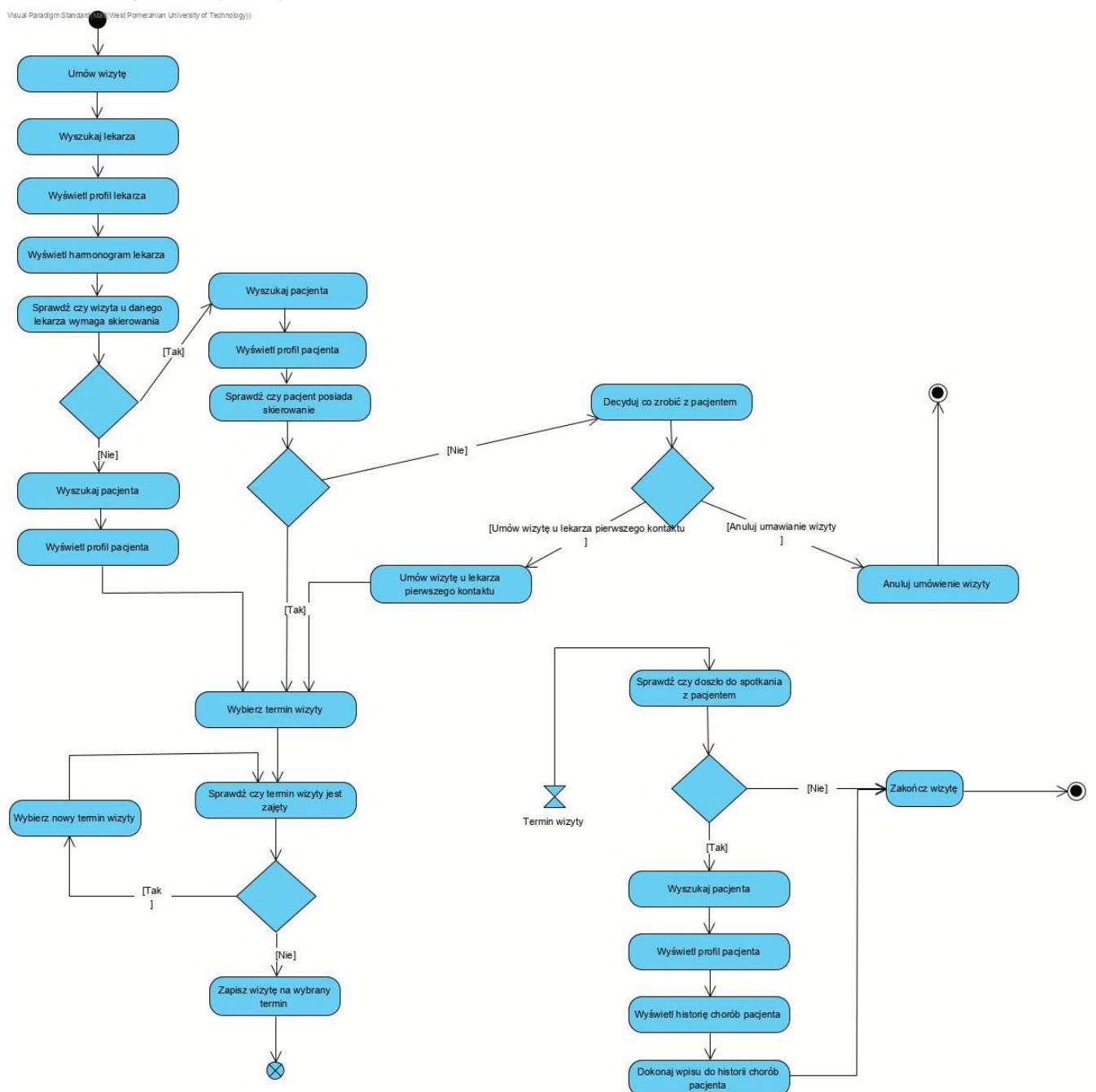
Język/narzędzie	Uzasadnienie
Język Java	Język obiektowy ogólnego przeznaczenia, o ogromnej ilości bibliotek przyspieszających tworzenie aplikacji.
phpMyAdmin	Narzędzie służące do łatwego zarządzania bazą danych MySQL
Baza danych MySQL Mariadb	System zarządzania relacyjną bazą danych ułatwia modelowanie danych przetwarzanych przez system.
Github	Najpopularniejszy system kontroli wersji, który ułatwia współpracę między osobami wchodząymi w skład grupy projektowej.
Intelij IDEA	To najbardziej inteligentny IDE dla Javy. Zawiera zestaw gotowych funkcji i narzędzi do integracji nowoczesnych technologii framework i tworzenia stron internetowych.

## 9.3 Diagramy UML

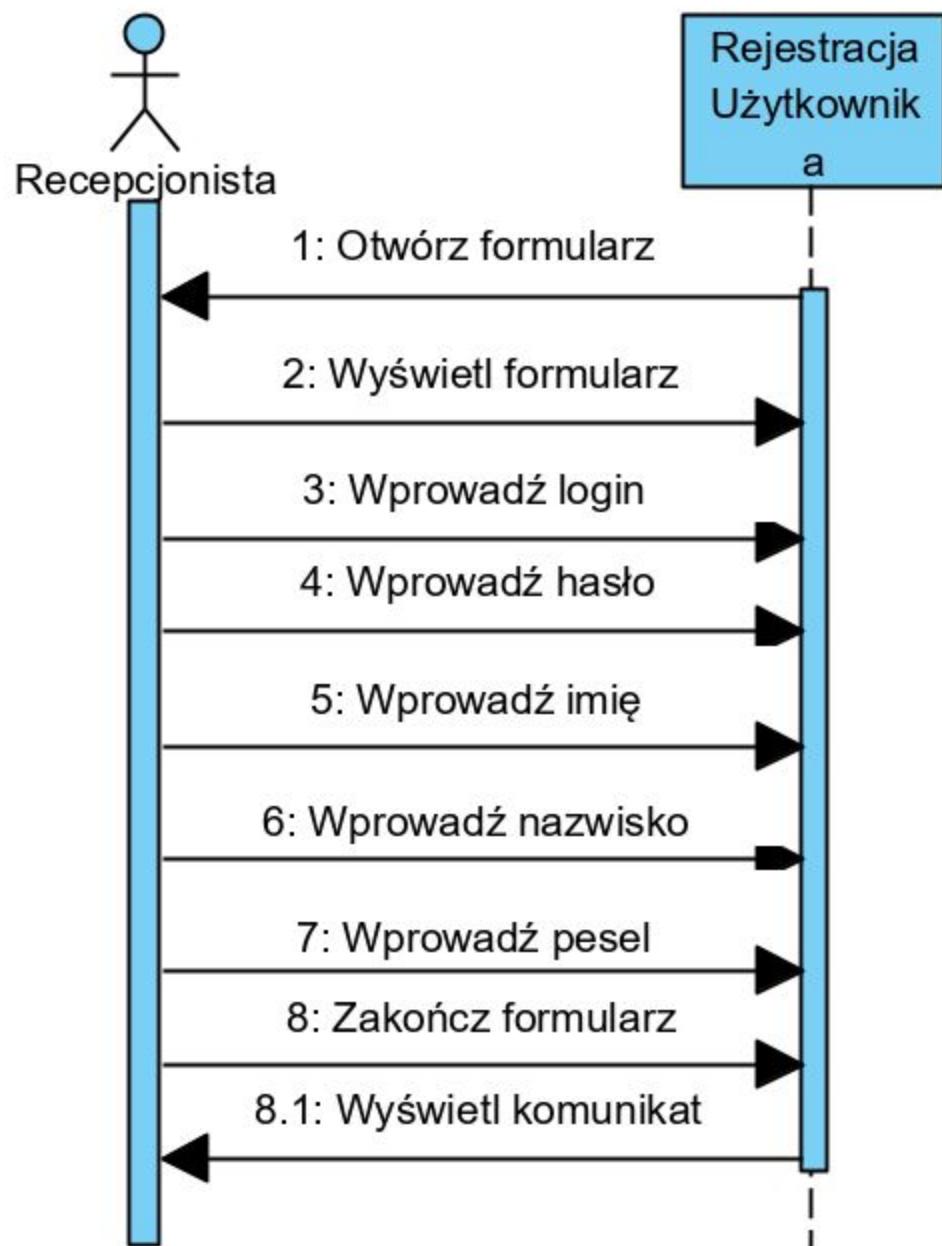
### 9.3.1 Diagram(-y) klas

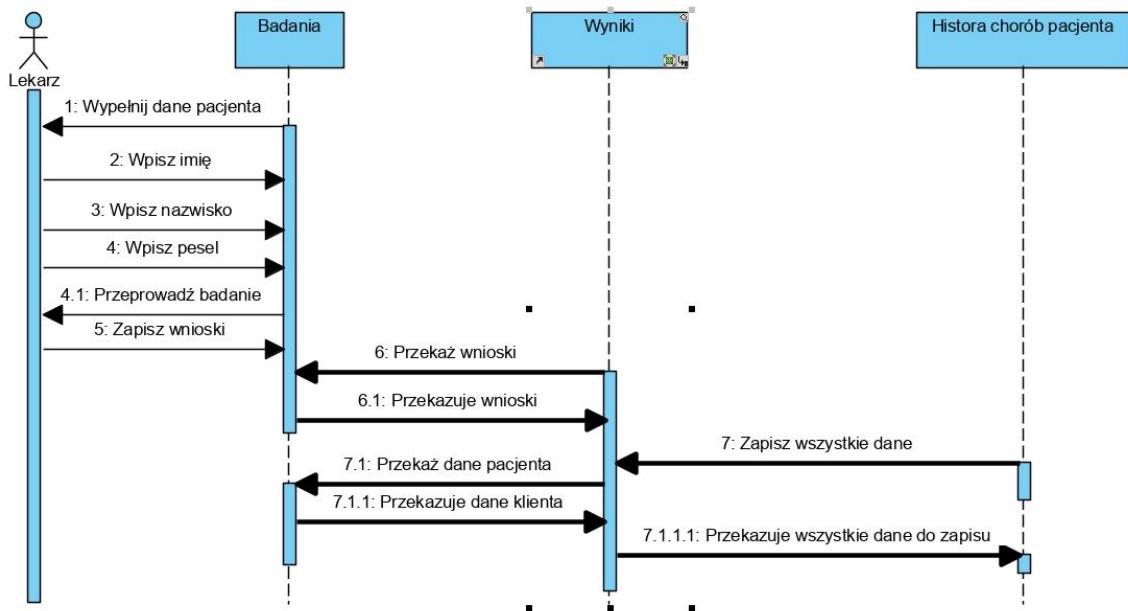


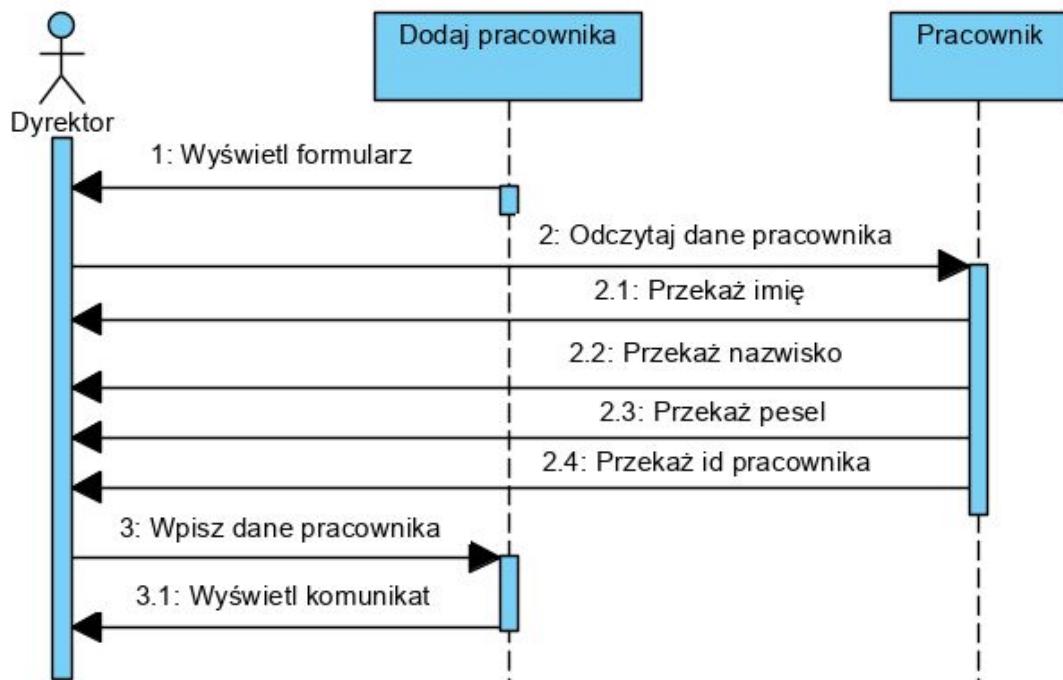
### 9.3.2 Diagram(-y) czynności

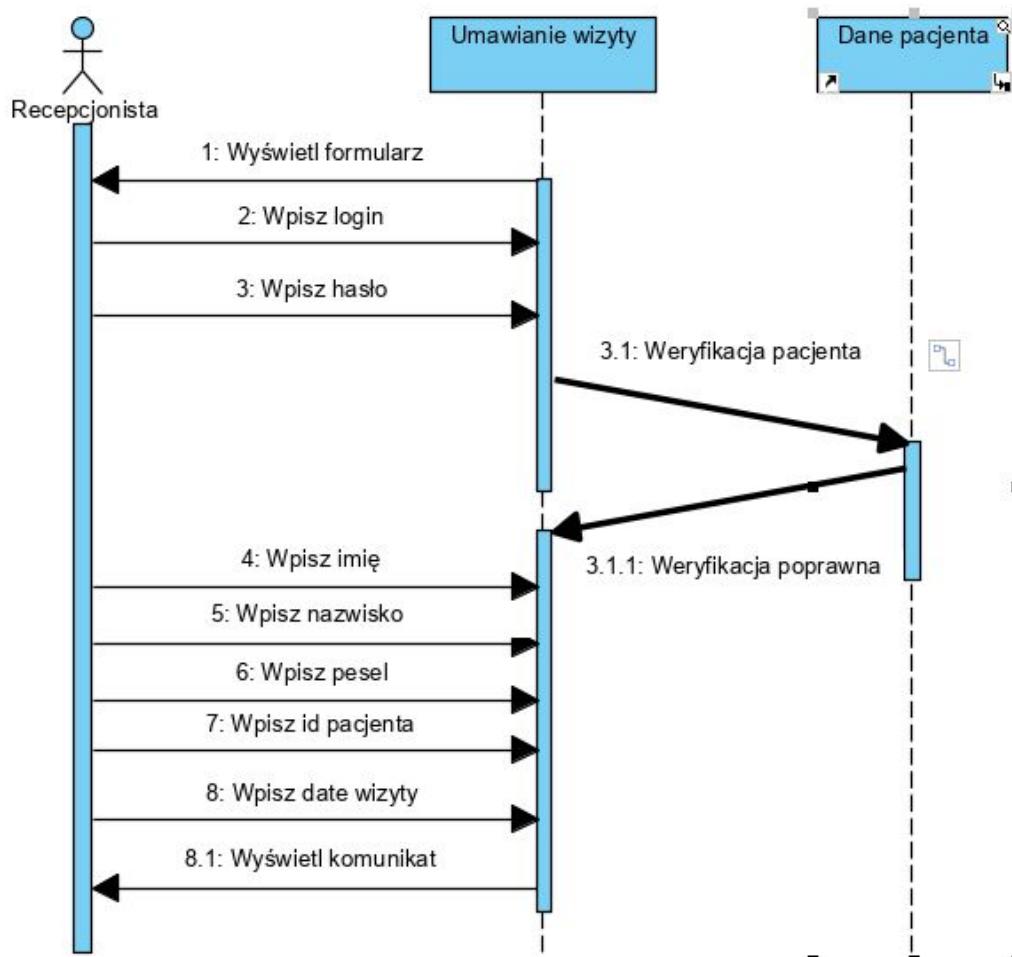


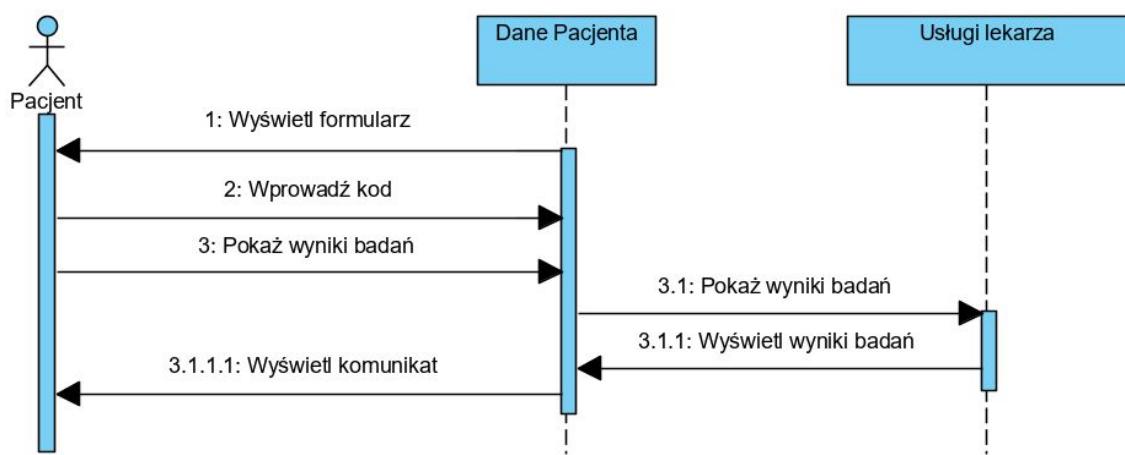
### 9.3.3 Diagramy sekwencji



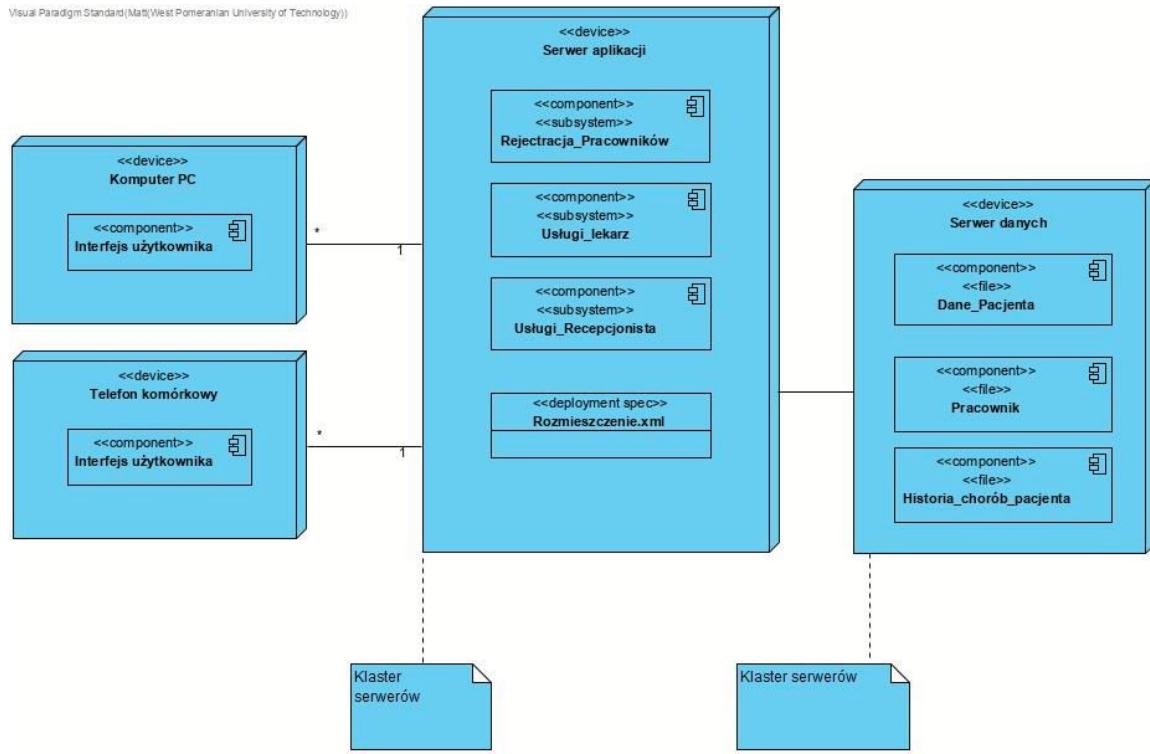


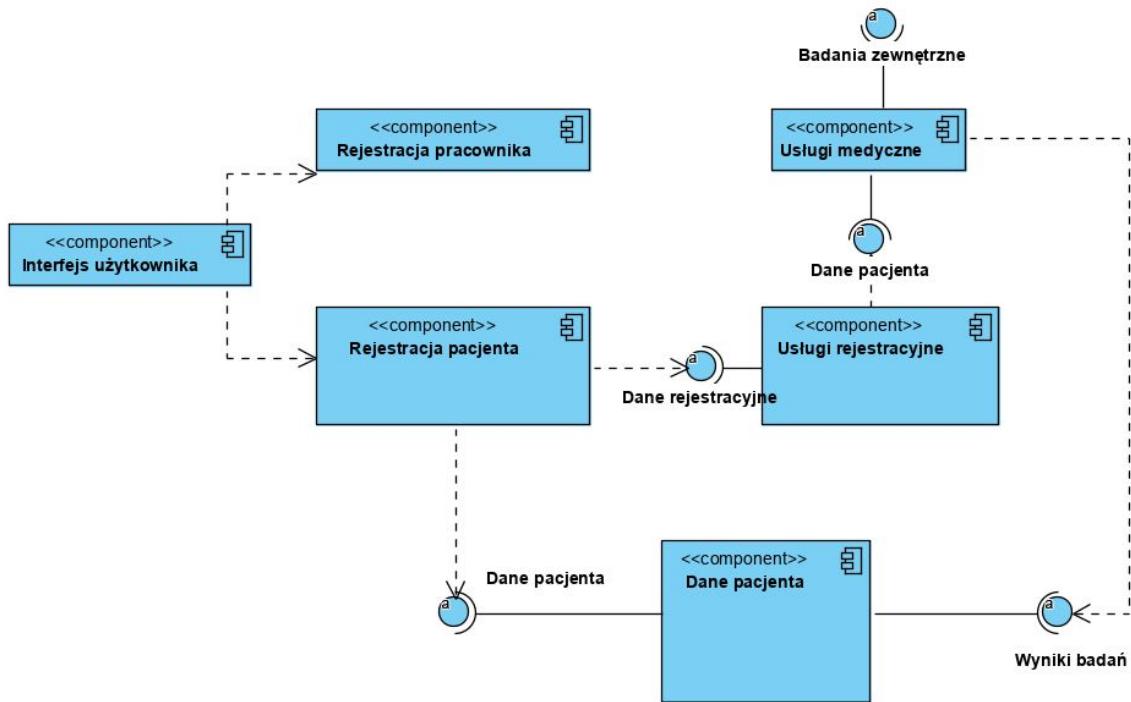


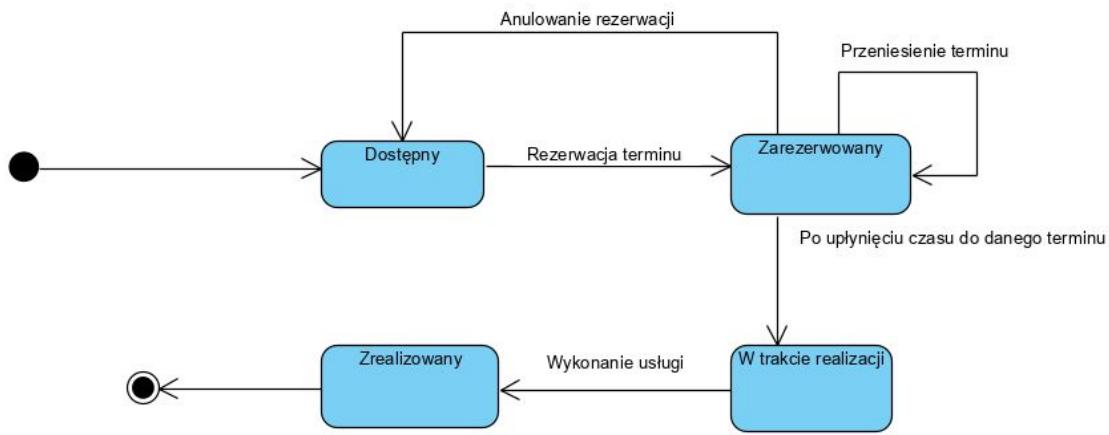




### 9.3.4 Inne diagramy







## **9.4 Charakterystyka zastosowanych wzorców projektowych**

Brak

## **9.5 Projekt bazy danych**

### **9.5.1 Schemat**

w trzeciej formie normalnej; jeśli w innej to umieć uzasadnić wybór

### **9.5.2 Projekty szczegółowe tabel**

## **9.6 Projekt interfejsu użytkownika**

Co najmniej dla głównej funkcjonalności programu – w razie wątpliwości, uzgodnić z prowadzącym zajęcia

### **9.6.1 Lista głównych elementów interfejsu**

okien, stron, aktywności (Android)

### **9.6.2 Przejścia między głównymi elementami**

### **9.6.3 Projekty szczegółowe poszczególnych elementów**

każdy element od nowej strony z następującą minimalną zawartością:

- numer – ID elementu
- nazwa – np. formularz danych produktu
- projekt graficzny – wystarczy schemat w narzędziu graficznym lub zrzut ekranu – z przykładowymi informacjami (nie pusty!!!)
  
- opcjonalnie:
- opis – dodatkowe opcjonalne informacje o przeznaczeniu, obsłudze – jeśli nazwa nie będzie wystarczająco czytelna
- wykorzystane dane – jakie dane z bazy danych są wykorzystywane
- opis działania – tabela pokazująca m.in. co się dzieje po kliknięciu przycisku, wybraniu opcji z menu itp.

## **9.7 Procedura wdrożenia**

jeśli informacje w harmonogramie nie są wystarczające (a zapewne nie są)

## **10 Dokumentacja dla użytkownika**

Opcjonalnie – dla chętnych

Na podstawie projektu docelowej aplikacji, a nie zaimplementowanego prototypu architektury

4-6 stron z obrazkami (np. zrzuty ekranowe, polecenia do wpisania na konsoli, itp.)

- pisana językiem odpowiednim do grupy odbiorców – czyli najczęściej nie do informatyków
- może to być przebieg krok po kroku obsługi jednej głównej funkcji systemu, kilku mniejszych, instrukcja instalacji lub innej pomocniczej czynności.

## 11 Podsumowanie

### ***11.1 Szczegółowe nakłady projektowe członków zespołu***

tabela (kolumny to osoby, wiersze to działania) pokazująca, kto ile czasu poświęcił na projekt oraz procentowy udział każdej osoby w danym zadaniu oraz wiersz podsumowania – udział każdej osoby w skali całego projektu

## 12 Inne informacje

przydatne informacje, które nie zostały ujęte we wcześniejszych punktach

- Numer – jako ID
- Nazwa
- Uzasadnienie biznesowe – odwołanie (-a) do elementów wymienionych w 5.1.5. (id i treść elementu, do którego się odwołujemy)
- Użytkownicy
- Scenariusze, dla każdego z nich:
  - Warunki początkowe
  - **Przebieg działań** – numerowana lista kroków, ze wskazaniem, kto realizuje dany krok
  - Efekty – warunki końcowe
  - Wymagania niefunkcjonalne – szczegółowe wobec poszczególnych wymagań funkcjonalnych
  - Częstotliwość - na skali 1-5 lub BN-BW
  - Istotność – inaczej: zależność krytyczna, znaczenie - na skali 1-5 lub BN-BW

### **Ważne!**

*Elementy od warunków początkowych do końca mogą być grupowane, tj. specyfikacja pojedynczego przypadku użycia może zawierać:*

- pojedynczy przebieg działań (scenariusz główny) oraz ew. scenariusze alternatywne, albo
- wiele przebiegów głównych wraz z ew. scenariuszami alternatywnymi – wtedy każdy z przebiegów głównych powinien być opisany wg tych punktów (od warunków początkowych do końca).