Dokumentacja projektu zaliczeniowego

Przedmiot: Inżynieria oprogramowania

Temat: **Oprogramowanie do wspierania placówki medycznej.**

Autorzy: **Sanocki Maciej, Deluga Bartosz**

Grupa: I1-211B

Kierunek: informatyka

Rok akademicki: 2

Poziom i semestr: I/4

Tryb studiów: stacjonarne

Należy pozostawić wszelkie nagłówki tego dokumentu, a umieszczać treść w odpowiednich miejscach zamiast obecnych objaśnień.

Stronę tytułową można sformatować w dowolny sposób, ale należy pozostawić zawartość informacyjną w układzie pokazanym powyżej.

Praca powinna zostać złożona wyłącznie w formacie pdf. Przed wygenerowaniem ostatecznej wersji należy zaktualizować spis treści – wyświetlane dwa poziomy.

Niniejszą informację należy również usunąć z wersji końcowej.

Spis treści

[2 Odnośniki do innych źródeł 4](#_Toc1976793)

[3 Słownik pojęć 5](#_Toc1976794)

[4 Wprowadzenie 6](#_Toc1976795)

[4.1 Cel dokumentacji 6](#_Toc1976796)

[4.2 Przeznaczenie dokumentacji 6](#_Toc1976797)

[4.3 Opis organizacji lub analiza rynku 6](#_Toc1976798)

[4.4 Analiza SWOT organizacji 7](#_Toc1976799)

[5 Specyfikacja wymagań 8](#_Toc1976800)

[5.1 Charakterystyka ogólna 8](#_Toc1976801)

[5.2 Wymagania funkcjonalne 9](#_Toc1976802)

[5.3 Wymagania niefunkcjonalne 14](#_Toc1976803)

[6 Zarządzanie projektem 16](#_Toc1976804)

[6.1 Zasoby ludzkie 16](#_Toc1976805)

[6.2 Harmonogram prac 16](#_Toc1976806)

[6.3 Etapy/kamienie milowe projektu 16](#_Toc1976807)

[7 Zarządzanie ryzykiem 17](#_Toc1976808)

[7.1 Lista czynników ryzyka 17](#_Toc1976809)

[7.2 Ocena ryzyka 17](#_Toc1976810)

[7.3 Plan reakcji na ryzyko 17](#_Toc1976811)

[8 Zarządzanie jakością 18](#_Toc1976812)

[8.1 Scenariusze i przypadki testowe 18](#_Toc1976813)

[9 Projekt techniczny 19](#_Toc1976814)

[9.1 Opis architektury systemu 19](#_Toc1976815)

[9.2 Technologie implementacji systemu 19](#_Toc1976816)

[9.3 Diagramy UML 19](#_Toc1976817)

[9.4 Charakterystyka zastosowanych wzorców projektowych 23](#_Toc1976818)

[9.5 Projekt bazy danych 23](#_Toc1976819)

[9.6 Projekt interfejsu użytkownika 23](#_Toc1976820)

[9.7 Procedura wdrożenia 24](#_Toc1976821)

[10 Dokumentacja dla użytkownika 25](#_Toc1976822)

[11 Podsumowanie 26](#_Toc1976823)

[11.1 Szczegółowe nakłady projektowe członków zespołu 26](#_Toc1976824)

[12 Inne informacje 27](#_Toc1976825)

# Odnośniki do innych źródeł

tj. do wykorzystywanych narzędzi / projektów w tych narzędziach

* + Zarządzania projektem – [Trello](https://trello.com/b/HswUPBwO/inżynieria-oprogramowania)
  + Wersjonowanie kodu – [Github](https://github.com/Xewol/IO)
  + System obsługi defektów – [GitHub](https://github.com/Xewol/IO)

# Słownik pojęć

UUID – Unique User ID – unikatowy identyfikator użytkownika.

UX – User Experience – doświadczenia użytkownika

UI – User Interface – interfejs użytkownika

# Wprowadzenie

## Cel dokumentacji

Dokumentacja zawiera informacje dotyczące placówek medycznych. Zaczynając od Analizy SWOT, aż przez opis organizacji. Oprócz tego dokumentacja posiada również stronę techniczną, która przedstawia zasadę działania oraz opis aplikacji, która jest głównym celem całej pracy. Sama dokumentacja ma na celu zobrazowanie zasady działania takiego programu oraz przedstawienie możliwości jakie placówki medyczne będą mogły osiągnąć używając tej właśnie aplikacji.

## Przeznaczenie dokumentacji

Dokumentacja przeznaczona jest dla pracowników, jednostki medycznej używających oprogramowania.

## Opis organizacji lub analiza rynku

Oprogramowanie jest zorientowane na rynek masowy.

Oprogramowanie jest przeznaczone dla placówek wielopowierzchniowych zatrudniających powyżej 50 lekarzy i posiadającą bazę pacjentów nie mniejszą niż 10 tyś.

Organizacje te posiadają zróżnicowane oddziały wraz z wyspecjalizowanym personelem.

Celem działań organizacji jest udzielanie świadczeń zdrowotnych w warunkach zamkniętych, a samo przyjęcie pacjenta następuje po wcześniejszym otrzymaniu skierowania od lekarza rodzinnego. Następnie pacjent kierowany jest na wybrany oddział, gdzie otrzymuje zakwaterowanie, a także lekarza z danego oddziału. Określana jest metoda leczenia, następnie po stwierdzeniu, że pacjent jest gotowy do wykwaterowania – otrzymuje on wypis, natomiast historia leczenia jest archiwizowana przez 20 lat.

Organizacje takie są rozproszone po całej Polsce, przy czym nie powinno to stanowić problemu dla samego wdrożenia oprogramowania, które jest zorientowane na uniwersalność.

## Analiza SWOT organizacji

|  |  |
| --- | --- |
| **SZANSE** | **ZAGROŻENIA** |
| * Inwestycje w najnowszą technologie za sprawą dofinansowań ze strony państwa lub prywatnych jednostek * Promocja na skalę światową * Napływ propozycji współpracy z uczelniami medycznymi * Promocja zdrowia | * Zagrożenia biologiczne powodujące przeciążenie danych jednostek szpitalnych * Rezygnacja z profilaktyki oraz leczenia publicznego, spowodowana długim czasem oczekiwania (korzystanie z prywatnych placówek medycznych) * Redukcja poziomu zatrudnienia spowodowana szpitalną siatką płac |

# Specyfikacja wymagań

## Charakterystyka ogólna

### Definicja produktu

AID System ™ – System wspomagania zarządzania.

### Podstawowe założenia

AID System jest to oprogramowanie do wspomagania pracy szpitala.

System jest odpowiedzialny za:

* Ewidencję godzin pracy,
* Zarządzanie bazą pacjentów,
* Zarządzaniem bazą pracowników,
* Zarządzaniem asortymentem szpitala,
* Umożliwienie komunikacji w ramach jednego oddziału jak i komunikacji międzyoddziałowej.

Oprogramowanie jest formą nakładki na system Windows 10, która wymaga zalogowania się swoim unikalnym kodem UUID i hasłem. Po zalogowaniu się dany użytkownik ma dostęp do konta, które posiada tylko przyznane mu uprawnienia w ramach, których może się on poruszać po systemie. Historia leczenia pacjentów jest utrzymywana przez 20 lat i może być ona udostępniona tylko osobom do tego upoważnionym ( np. Policja).

### Cel biznesowy

Decydując się na wdrożenie systemu, organizacja szuka sposobu na usprawnienie pracy całej jednostki, a tym samym stać się bardziej atrakcyjną dla swoich pacjentów.

### Użytkownicy

* Kierownik Szpitala
* Lekarz
* Pielęgniarka
* Woźny
* Administrator Systemu

### Korzyści z systemu

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Korzyść | Kierownik Szpitala | Lekarz | Pielęgniarka | Woźny | Administrator Systemu |
| 1 | Ewidencja Pracy | X | X | X | X | X |
| 2 | Dostęp do bazy pacjentów | X | X | X |  | X |
| 3 | Dostęp do bazy asortymentów | X | X | X | X | X |
| 4 | Dostęp do bazy pracowników | X |  |  |  | X |
| 5 | Dostęp do komunikacji | X | X | X |  | X |

### Ograniczenia projektowe i wdrożeniowe

W razie potrzeby udostępnienia danych dla innego wewnętrznego systemu,

AID System umożliwia udostępnienie:

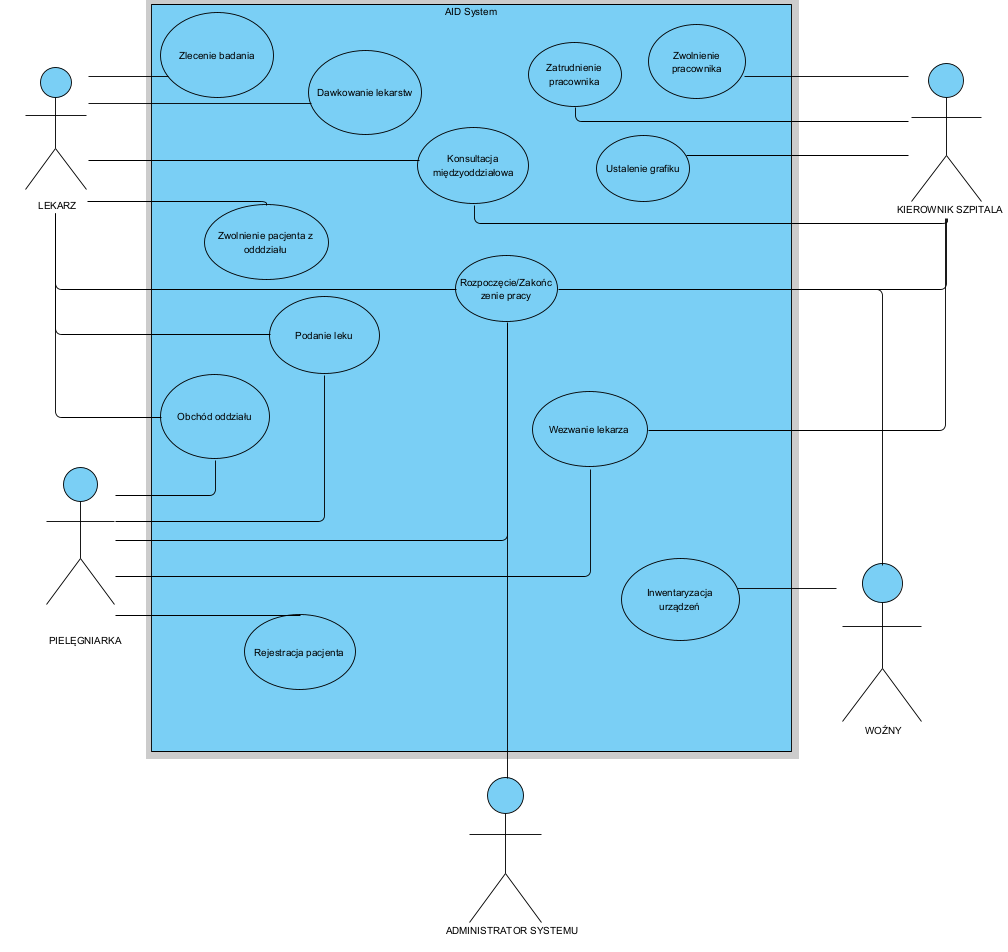
* Ewidencji pracy
* Bazy użytkowników
* Dostępu do asortymentów

## Wymagania funkcjonalne

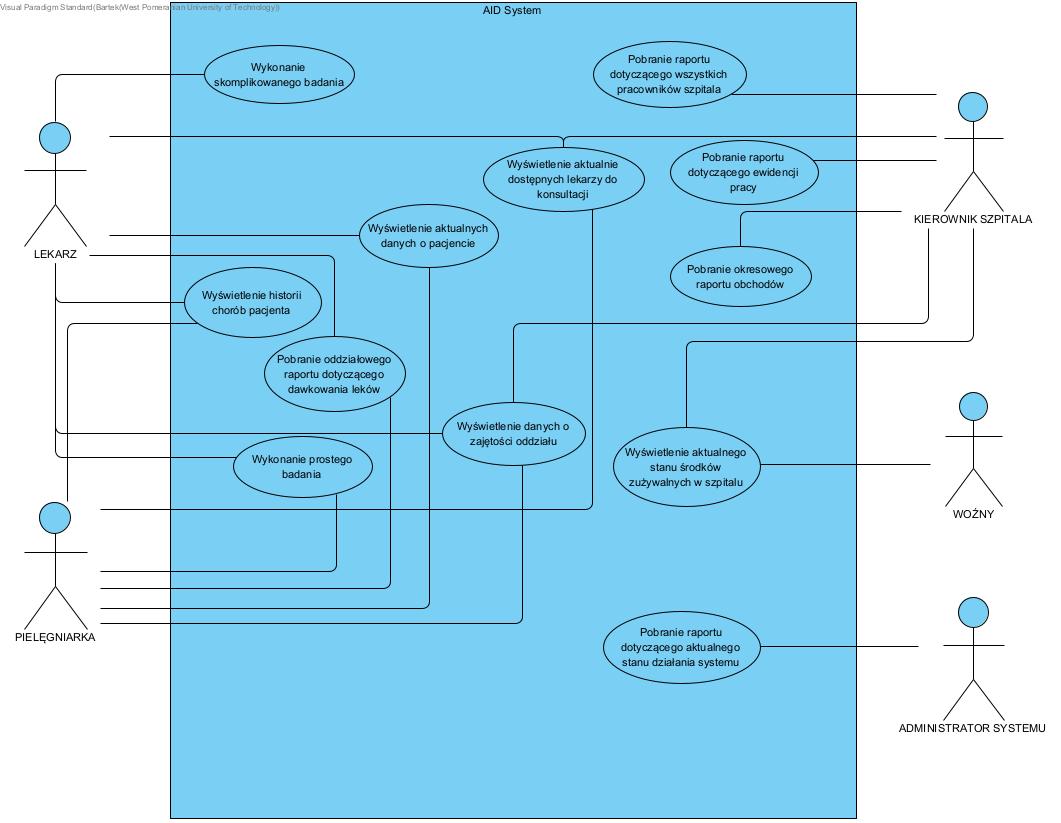
### Lista wymagań

1. Rozpoczęcie/Zakończenie pracy.
2. Rejestracja pacjenta.
3. Zlecenie badania.
4. Zwolnienie pacjenta z oddziału.
5. Inwentaryzacja urządzeń.
6. Wykonanie zabiegu.
7. Wykonanie badania skomplikowanego
8. Wykonanie badania prostego
9. Wyświetlanie historii chorób pacjenta.
10. Wyświetlanie danych o zajętości oddziału.
11. Pobrania różnych raportów (ewidencja pracy, obchodu, dostępnych lekarzy, pracowników, pacjentów)
12. Dawkowanie lekarstw.
13. Podanie leku.
14. Pobranie raportu o dawkowaniu lekarstw
15. Konsultacja międzyoddziałowa.
16. Wezwanie lekarza.
17. Zwolnienie pracownika.
18. Zatrudnienie pracownika.
19. Ustalanie grafiku.
20. Obchód oddziału.

### Diagramy przypadków użycia



Rysunek 1. Diagram UML I



Rysunek 2. Diagram UML II

### Szczegółowy opis wymagań

* Id. 1.
* Nazwa: Rozpoczęcie pracy.
* Ad. 1. Ewidencja pracy
* Lekarz, Administrator systemu, Woźny, Pielęgniarka, Kierownik Szpitala
* Scenariusz:
  + **Nazwa**: Meldunek rozpoczęcia pracy
  + **Warunki początkowe**: Pracownik nie jest jeszcze zalogowany do systemu
  + Przebieg:

1. Pracownik loguje się do systemu swoimi danymi
2. System sprawdza czy dane są prawidłowe.
3. Jeżeli dane są poprawne to zwracane jest konto pracownika, w przeciwnym przypadku należy spróbować ponownie.
4. Pracownik uzyskuje dostęp do odpowiednich sektorów aplikacji
5. System tworzy w bazie danych log użytkownika.
   * Efektem jest ewidencja pracy pracownika.
   * Częstotliwość: Bardzo wysoka
   * Istotność: Bardzo wysoka

* Id. 2.
* Nazwa: Zakończenie pracy.
* Ad. 1. Ewidencja pracy
* Lekarz, Administrator systemu, Woźny, Pielęgniarka, Kierownik Szpitala
* Scenariusz:
  + **Nazwa**: Meldunek zakończenia pracy
  + **Warunki początkowe**: Pracownik jest zalogowany do systemu
  + **Przebieg**:

1. Pracownik wylogowuje się z systemu.
2. System wyszukuje log, wcześniej stworzony i aktualizuje go dopisując godzinę zakończenia pracy.
   * Efektem jest ewidencja pracy pracownika.
   * Częstotliwość: Bardzo wysoka
   * Istotność: Bardzo wysoka

* Id. 3.
* Nazwa: Obchód oddziału.
* Ad. 2. Dostęp do bazy pacjentów, Ad. 3. Dostęp do bazy asortymentu
* Lekarz, Pielęgniarka
* Scenariusz:
  + **Nazwa**: Obchód wybranego oddziału
  + **Warunki początkowe**: Wszyscy użytkownicy muszą być zalogowani do systemu, wybrany oddział musi mieć przynajmniej jednego pacjenta
  + **Przebieg**:

1. Lekarz wysyła powiadomienie o zbliżającym się obchodzie
2. Zebrany oddział pracowników przeprowadza obchód
3. Wszystkie braki i nowości związane z pacjentami lub asortymentem są od razu zgłaszane w systemie i wysyłane do bazy danych.
4. System zapisuje podsumowanie obchodu.
   * Efektem jest obchód danego oddziału.
   * Częstotliwość: Wysoka
   * Istotność: Bardzo wysoka

* Id. 4.
* Nazwa: Przyjęcie pacjenta.
* Ad. 4. Dostęp do bazy pacjentów
* Pielęgniarka
* Scenariusz:
  + **Nazwa**: Przyjęcie pacjenta wraz z jego rejestracją
  + **Warunki początkowe**: Pacjent z ważnym skierowaniem, a także nie zarejestrowany na żadnym oddziale
  + **Przebieg**:

1. Pacjent wraz ze skierowaniem przychodzi do placówki medycznej.
2. Pielęgniarka po numerze PESEL wyszukuje pacjenta
3. System zwraca rekord, w razie niepowodzenia przekierowuje do formularza tworzenia karty pacjenta.
4. Pielęgniarka przydziela pacjentowi oddział, wysyłając przy tym dane do systemu o numerze oddziału.
5. System aktualizuje kartę pacjenta w bazie pacjentów.
   * Efektem jest przyjęcie pacjenta.
   * Częstotliwość: Wysoka
   * Istotność: Bardzo wysoka

* Id. 5.
* Nazwa: Stworzenie grafiku.
* Ad. 2. Dostęp do bazy pracowników
* Kierownik szpitala
* Scenariusz:
  + **Nazwa**: Stworzenie grafiku na kolejny miesiąc.
  + **Warunki początkowe**: Grafik na przyszły miesiąc nie jest wdrożony.
  + **Przebieg**:

1. System wczytuje z bazy danych wszystkie osoby, które mają jakieś dyspozycyjności w sprawie grafiku na kolejny miesiąc.
2. Kierownik uwzględnia, bądź nie osób z dyspozycyjnościami.
3. Kierownik wysyła grafik.
4. System sprawdza czy grafik jest możliwy do ustalenia.
5. Jeżeli system przyjął grafik, zostaje on opublikowany, w przeciwnym przypadku należy poprawić błędy.
   * Efektem jest nowy grafik.
   * Częstotliwość: Średnia
   * Istotność: Bardzo wysoka

## Wymagania niefunkcjonalne

W odniesieniu do całego systemu, modułów lub innych składowych systemu

1. Wydajność – w odniesieniu do konkretnych sytuacji – funkcji systemu
2. Bezpieczeństwo – dane przetrzymywane w technologii RAID 1 - ze względu na ich wrażliwość danych w bazie.
3. Zabezpieczenia – oparte o najnowsze technologie.

# Zarządzanie projektem

## 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

## 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.)

## Etapy/kamienie milowe projektu

dla głównych etapów projektu

# Zarządzanie ryzykiem

## Lista czynników ryzyka

Wypełniona lista kontrolna

## Ocena ryzyka

prawdopodobieństwo i wpływ

## Plan reakcji na ryzyko

Działania w odniesieniu do poszczególnych ryzyk.

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

# Zarządzanie jakością

## 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
* *przebieg lub zestaw danych testowych musi zawierać jawną informację o warunku zaliczenia testu*

# Projekt techniczny

## Opis architektury systemu

Architektura systemu wykorzystuje system operacyjny Windows, którego jest ona nakładką. Dzięki temu UX / UI będzie taki sam jak przy korzystaniu ze zwykłego systemu. Proste do rozwiązania programistycznie, gdyż wiele bibliotek graficznych dla systemu jest oparte o rozwiązania Microsoftu.

## Technologie implementacji systemu

|  |  |
| --- | --- |
| Język systemu | C++ |
| System Bazy Danych | MySQL |
| Baza Danych | SQL |

**C++** - Najbardziej znany język niskiego poziomu zaraz po C, wybrany ze względu na obiektowość. Daje duże możliwości zarządzania pamięcią i jest bardzo bezpieczny.

**MySQL** – Najpowszechniejszy system do zarządzania bazami danych. Używany ze względu

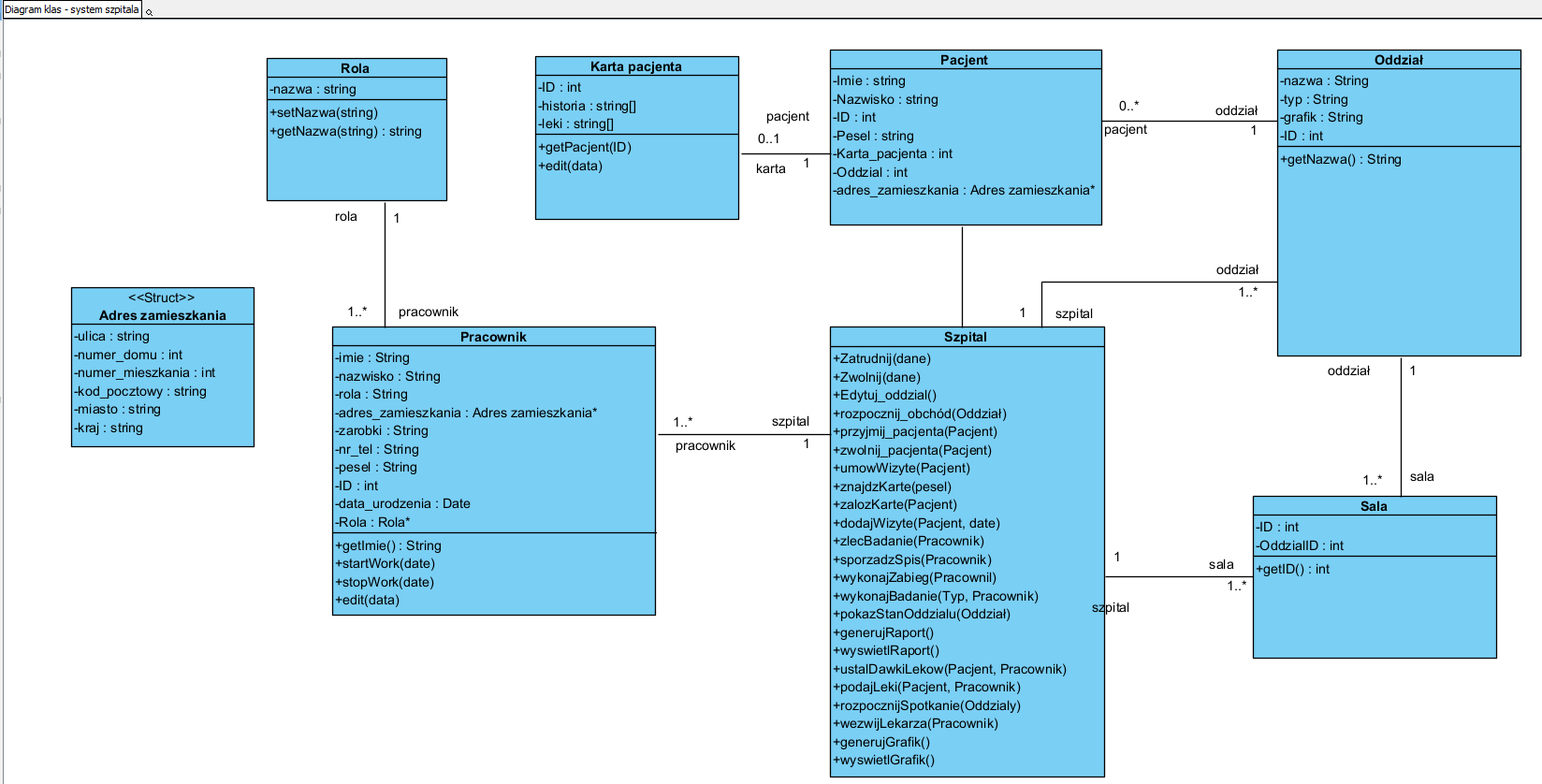
**SQL –** Relacyjna baza danych przeznaczona dla dużej ilości danych, w naszym przypadku– karty pacjentów.

## Diagramy UML

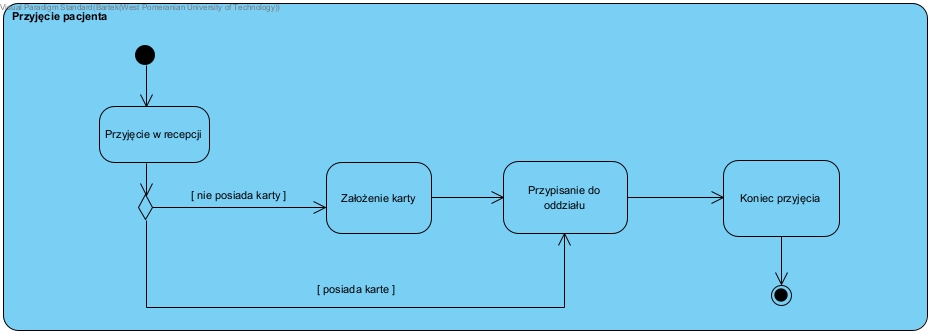
[Rysunek 1. Diagram UML I 10](#_Toc102995872)

[Rysunek 2. Diagram UML II 11](#_Toc102995873)

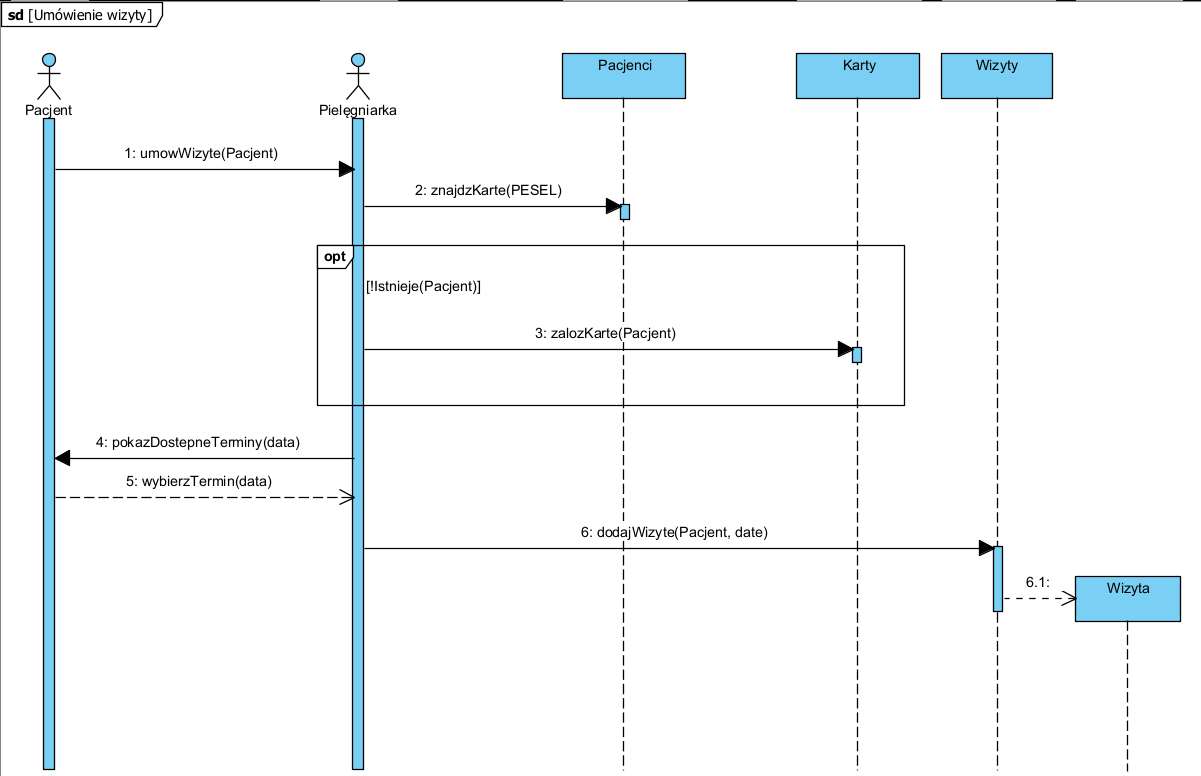
### Diagram(-y) klas

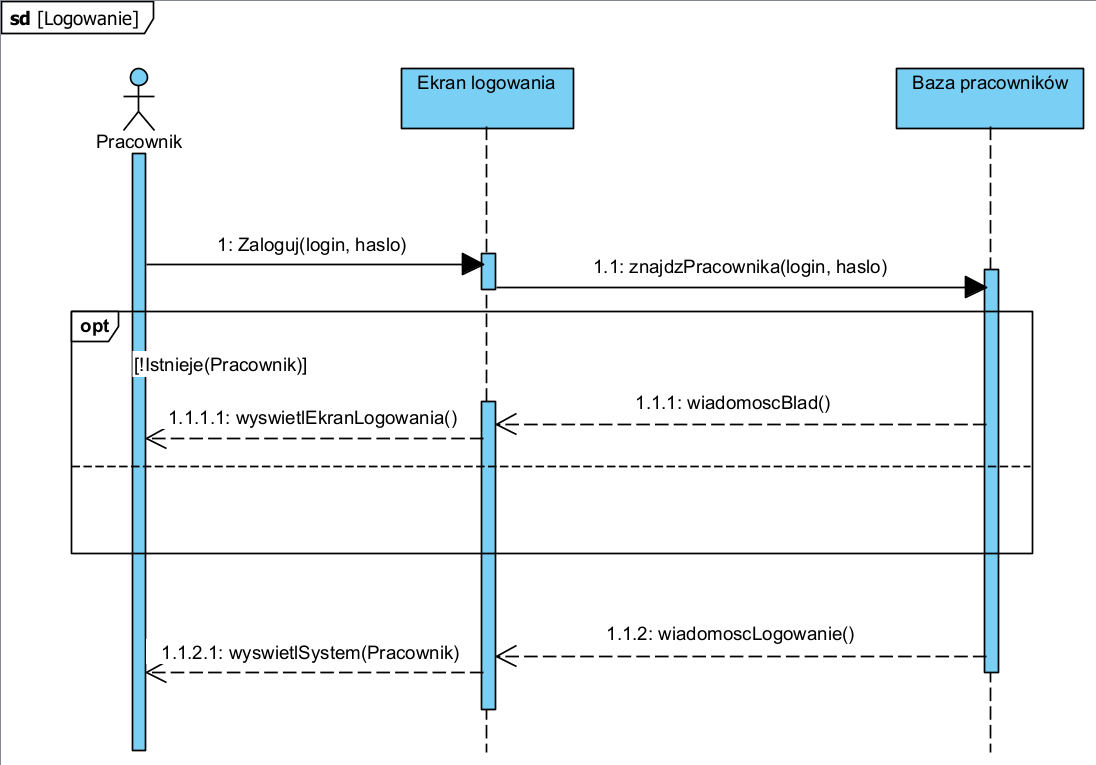


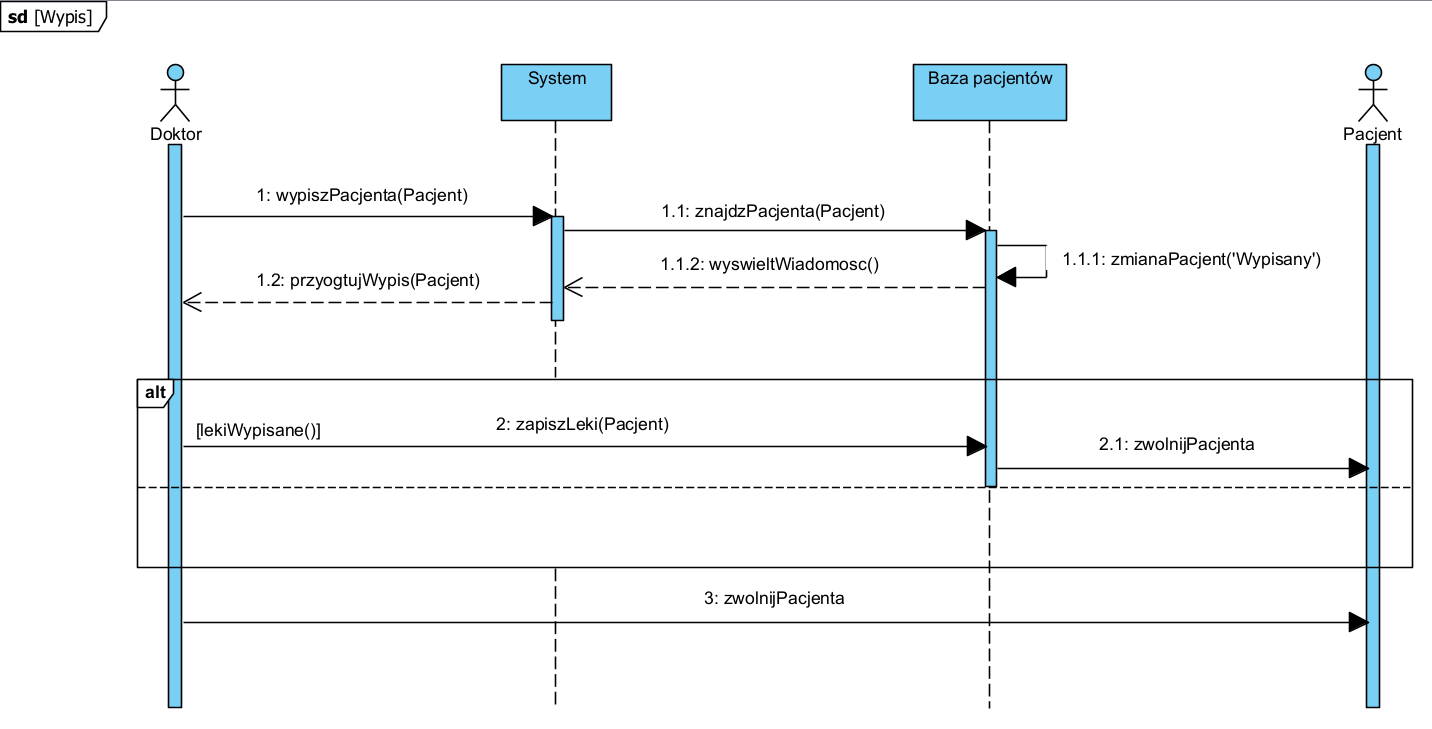
### Diagram(-y) czynności



### Diagramy sekwencji

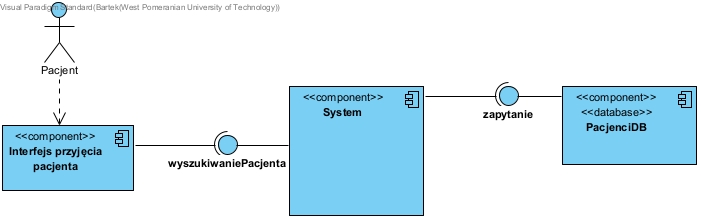


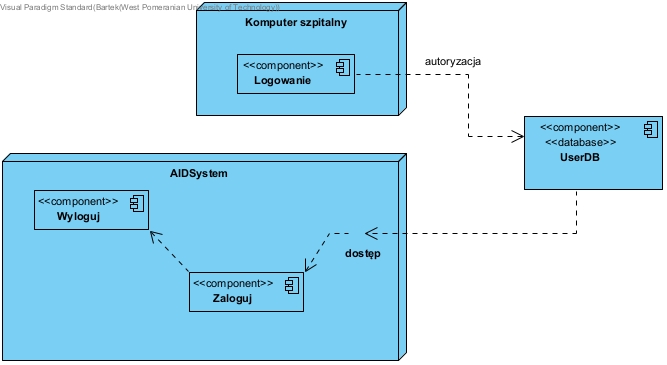


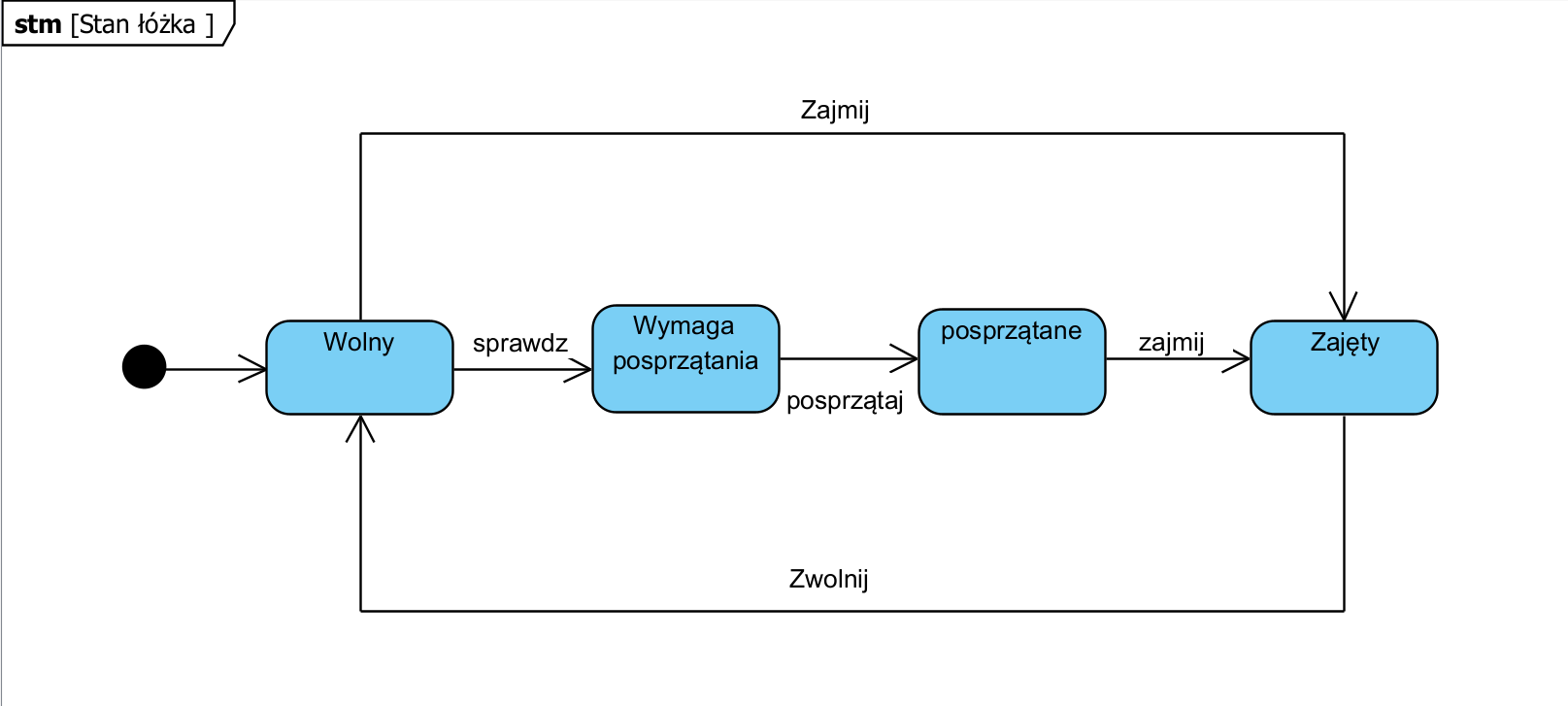


### Inne diagramy

co najmniej trzy – komponentów, rozmieszczenia, maszyny stanowej itp.







## Charakterystyka zastosowanych wzorców projektowych

informacja opisowa wspomagana diagramami (odsyłaczami do diagramów UML); jeśli wykorzystano wzorce projektowe, to należy wykazać dwa z nich

## Projekt bazy danych

### Schemat

### Projekty szczegółowe tabel

w zależności, czy następujące elementy są widoczne na schemacie b.d.: nazwa tabeli, nazwy pól, typ danych, wartości NULL, klucz główny, klucz obcy –

- jeśli TAK: i nie ma potrzeby pokazania dodatkowych elementów b.d., to ten punkt może być pusty,

- jeśli NIE: to podać te elementy, których nie widać na schemacie.

dodatkowymi elementami mogą być np. triggery, procedury, funkcje, indeksy, użytkownicy, role.

## Projekt interfejsu użytkownika

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

### Lista głównych elementów interfejsu

okien, stron, aktywności (Android)

### Przejścia między głównymi elementami

np. storyboard, schemat blokowy lub inna notacja

### Projekty szczegółowe poszczególnych elementów

dla 5-7 głównych 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.

## Procedura wdrożenia

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

# 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.

# Podsumowanie

## 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 – procentowy udział każdej osoby w skali całego projektu

# Inne informacje

przydatne informacje, które nie zostały ujęte we wcześniejszych punktach