Dokumentacja projektu zaliczeniowego

Przedmiot: Inżynieria oprogramowania

Temat: **System zarządzający gabinetem weterynaryjnym**

Autorzy: **Mateusz Małecki, Jakub Wajs**

Grupa: 21B

Kierunek: informatyka

Rok akademicki: 2020/2021

Poziom i semestr: I/4

Tryb studiów: niestacjonarne

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 6](#_Toc1976799)

[5 Specyfikacja wymagań 7](#_Toc1976800)

[5.1 Charakterystyka ogólna 7](#_Toc1976801)

[5.2 Wymagania funkcjonalne 7](#_Toc1976802)

[5.3 Wymagania niefunkcjonalne 8](#_Toc1976803)

[6 Zarządzanie projektem 9](#_Toc1976804)

[6.1 Zasoby ludzkie 9](#_Toc1976805)

[6.2 Harmonogram prac 9](#_Toc1976806)

[6.3 Etapy/kamienie milowe projektu 9](#_Toc1976807)

[7 Zarządzanie ryzykiem 10](#_Toc1976808)

[7.1 Lista czynników ryzyka 10](#_Toc1976809)

[7.2 Ocena ryzyka 10](#_Toc1976810)

[7.3 Plan reakcji na ryzyko 10](#_Toc1976811)

[8 Zarządzanie jakością 11](#_Toc1976812)

[8.1 Scenariusze i przypadki testowe 11](#_Toc1976813)

[9 Projekt techniczny 12](#_Toc1976814)

[9.1 Opis architektury systemu 12](#_Toc1976815)

[9.2 Technologie implementacji systemu 12](#_Toc1976816)

[9.3 Diagramy UML 12](#_Toc1976817)

[9.4 Charakterystyka zastosowanych wzorców projektowych 12](#_Toc1976818)

[9.5 Projekt bazy danych 12](#_Toc1976819)

[9.6 Projekt interfejsu użytkownika 12](#_Toc1976820)

[9.7 Procedura wdrożenia 13](#_Toc1976821)

[10 Dokumentacja dla użytkownika 14](#_Toc1976822)

[11 Podsumowanie 15](#_Toc1976823)

[11.1 Szczegółowe nakłady projektowe członków zespołu 15](#_Toc1976824)

[12 Inne informacje 16](#_Toc1976825)

# Odnośniki do innych źródeł

* Zarządzania projektem – monday.com
* Wersjonowanie kodu – sugerowany Git, Github.com
* System obsługi defektów –  monday.com, Github

# Słownik pojęć

Tabela lub lista z pojęciami, które wymagają wyjaśnienia, wraz z tymi wyjaśnieniami – w szczególności synonimy różnych pojęć używanych w dokumentacji.

|  |  |
| --- | --- |
| Weterynarz | Osoba prowadząca leczenie zwierzęcia |
| Pacjent | Zwierzę przyjęte do weterynarii |
| Właściciel | Właściciel zwierzęcia |
| Administrator | Osoba z pełnym dostępem do systemu, mogąca dokonywać zmian |

# Wprowadzenie

## Cel dokumentacji

Dokumentacja wykonywana jest w celu ustalenia definicji pojęć, oraz przeprowadzenie użytkowników przez strukturę produktu, oraz sposób jego wykorzystywania.

## Przeznaczenie dokumentacji

Dokumentacja skierowana jest do odbiorców produktu, oraz podmiotów zaangażowanych w jego powstawanie.

## Opis organizacji lub analiza rynku

Gabinet weterynaryjny szykujący się do ekspansji tj. utworzenia kolejnej placówki. Zakład zatrudnia 4 lekarzy, którzy przyjmują na wizyty oraz wykonują zabiegi. Gabinet prowadzi kartoteki pacjentów, oraz obsługuje rejestrację wizyt. Od strony biznesowej, dochód przynoszą opłaty za: wizyty, wykonane zabiegi i usługi, sprzedaż środków medycznych( takich jak leki i suplementy), kontrakty z wybranymi koncernami produkującymi karmy, leki i suplementy.

Analiza SWOT organizacji:

|  |  |  |
| --- | --- | --- |
|  | **Pozytywne** | **Negatywne** |
| **Wewnętrzne** | * Stale wzrastające zapotrzebowanie na usługi weterynaryjne * Zmniejszenie marginesu błędu przy obsłudze pacjentów * Przyjazna obsługa programu | * Produkt nie jest uniwersalny, tylko dopasowany do konkretnej kliniki * Konieczność szkolenia kadry z zakresu obsługi oprogramowania (zmieniającego się w czasie) * Dodatkowe koszty |
| **Zewnętrzne** | * Zwiększenie zainteresowania usługami danej kliniki weterynaryjnej ze względu na nowoczesną/innowacyjną aplikację * Deficyt zaawansowanego oprogramowania, spersonalizowanego do potrzeb rynku branży weterynaryjnej | * Niskie zainteresowanie produktem ze względu na przyzwyczajenie do starszych, ale bardziej uniwersalnych systemów * Luka technologiczna w społeczeństwie |

# Specyfikacja wymagań

## Charakterystyka ogólna

### Definicja produktu

System zorientowany w technologii webowej, do obsługi kliniki weterynaryjnej.

### Podstawowe założenia

System ma na celu wspomaganie funkcjonowania kliniki weterynaryjnej. Tworzenie historii leczenia, rejestrowanie nowych zwierząt, zapisywanie terminów, wystawianie i przechowywanie diagnozy, oraz kontakt z weterynarzami.

### Cel biznesowy

System znacząco usprawni i przyśpieszy pracę weterynarzy, tym samym zredukuje koszty jej prowadzenia i zwiększy przychody.

### Użytkownicy

-Weterynarz

-Pacjent

-Właściciel

-Administrator

### Korzyści z systemu

Weterynarze – szybki dostęp do historii leczenia, łatwość przy konstruowaniu recepty, lepsza organizacja pracy.

Pacjenci/Właściciele - brak konieczności podawania szczegółowych danych, możliwość umówienia wizyty, dostęp do recept i wskazówek dotyczących leczenia.

Administratorzy - szczegółowa kontrola działania kliniki.

### Ograniczenia projektowe i wdrożeniowe

Zgodnie z przepisami dotyczącymi ochrony danych osobowych, klienci przekazujący dane do kontaktu muszą zostać poinformowani o fakcie przechowywania ich danych, oraz muszą wyrazić na to zgodę. Ponad to strona internetowa, która będzie stanowić interfejs do zarządzania systemem, w przypadku wykorzystywania plików cookie, powinna o tym fakcie poinformować użytkownika oraz uzyskać od niego na to zgodę.

System od strony serwerowej będzie działał na dzierżawionym VPS z systemem dystrybucji Linux. W implementacji wykorzystana będzie konteneryzacja przy wykorzystaniu otwarto źródłowej usługi Docker. Aplikacja serwerowa zostanie zaimplementowana z wykorzystaniem frameworku PHP Symfony 5 oraz MySQL 8.

Komunikacja serwisu ze stroną internetową, będzie odbywała się zgodnie z architekturą REST, która jest podzbiorem protokołu komunikacyjnego HTTP. Dla zapewnienia bezpieczeństwa, wykorzystane zostanie rozszerzenie HTTPS.

## Wymagania funkcjonalne

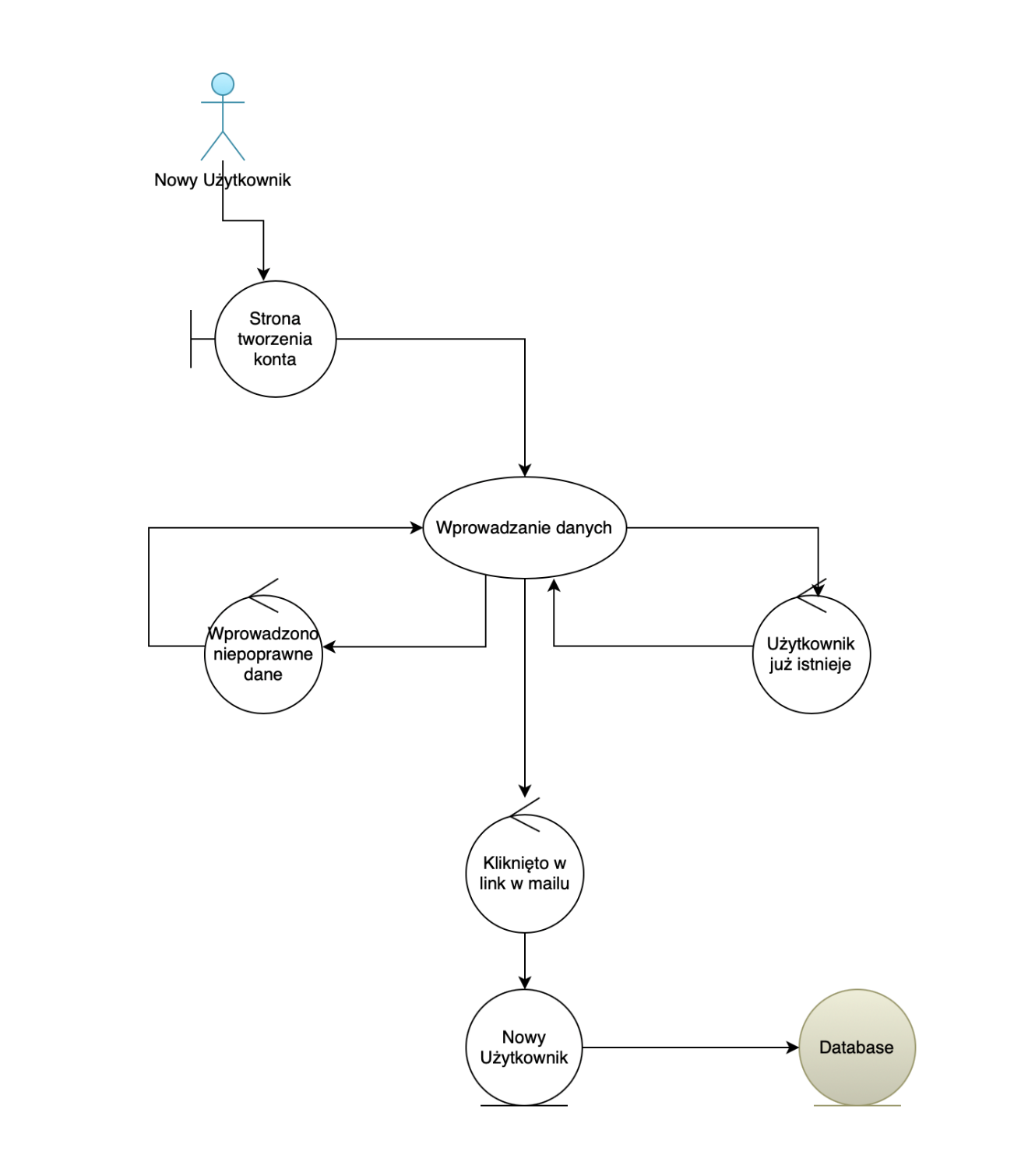
### Lista wymagań

1. **Rejestracja użytkownika**
2. **Rejestracja pacjenta.**
3. **Umówienie wizyty.**
4. **Zmiana terminu wizyty**
5. **Odwołanie wizyty**
6. **Sprawdzenie wolnych terminów wizyt**
7. **Zapisanie i pobranie informacji o pacjencie**
8. **Zapisanie i pobranie historii pacjenta**
9. **Wysyłanie powiadomień o nadchodzącej wizycie**
10. **Wysyłanie powiadomień dla zainteresowanych użytkowników o wolnym wcześniejszym terminie**

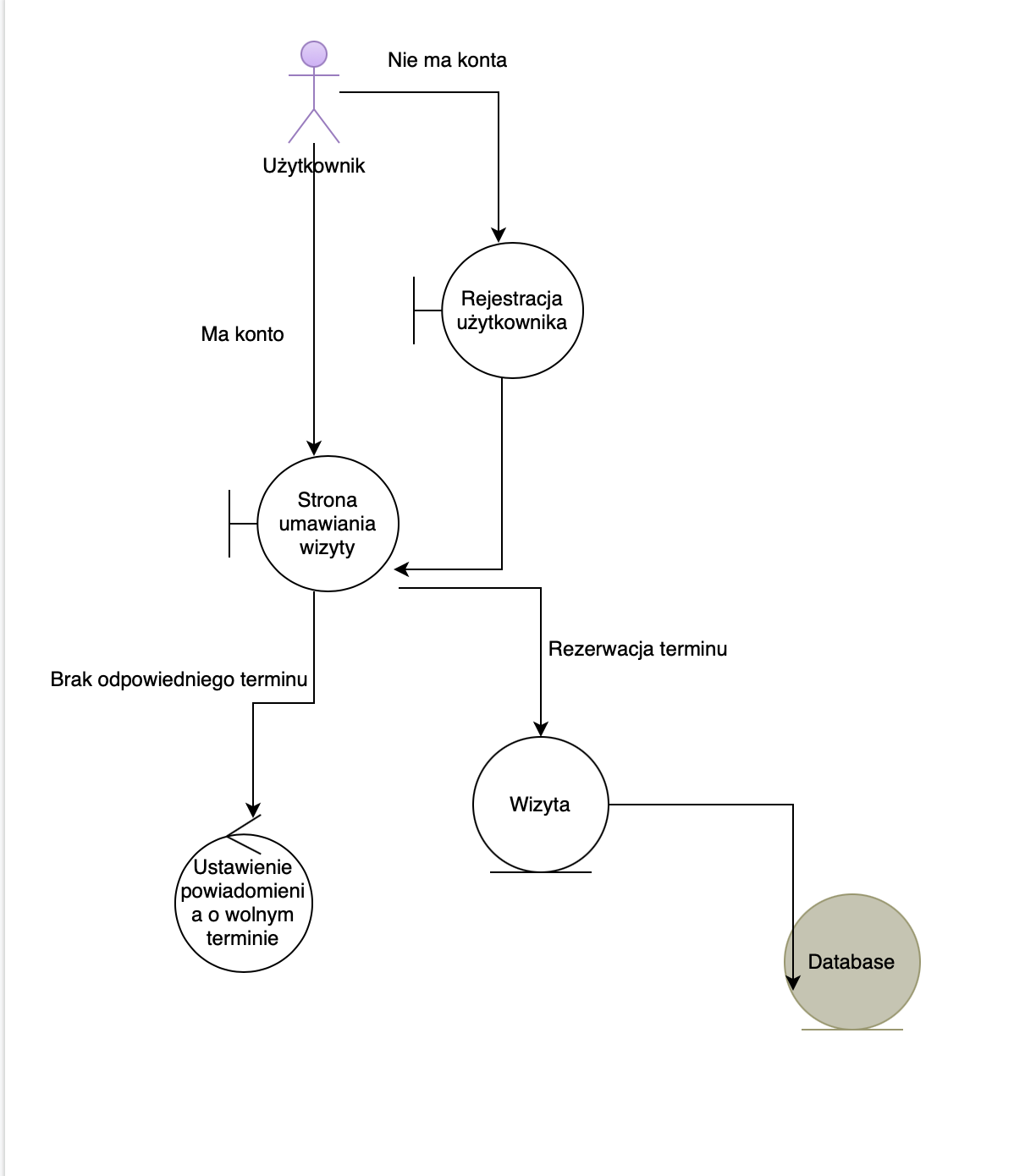
### Diagramy przypadków użycia

Tutaj same diagramy – bez specyfikacji, ale każdy diagram z tytułem i na osobnej stronie

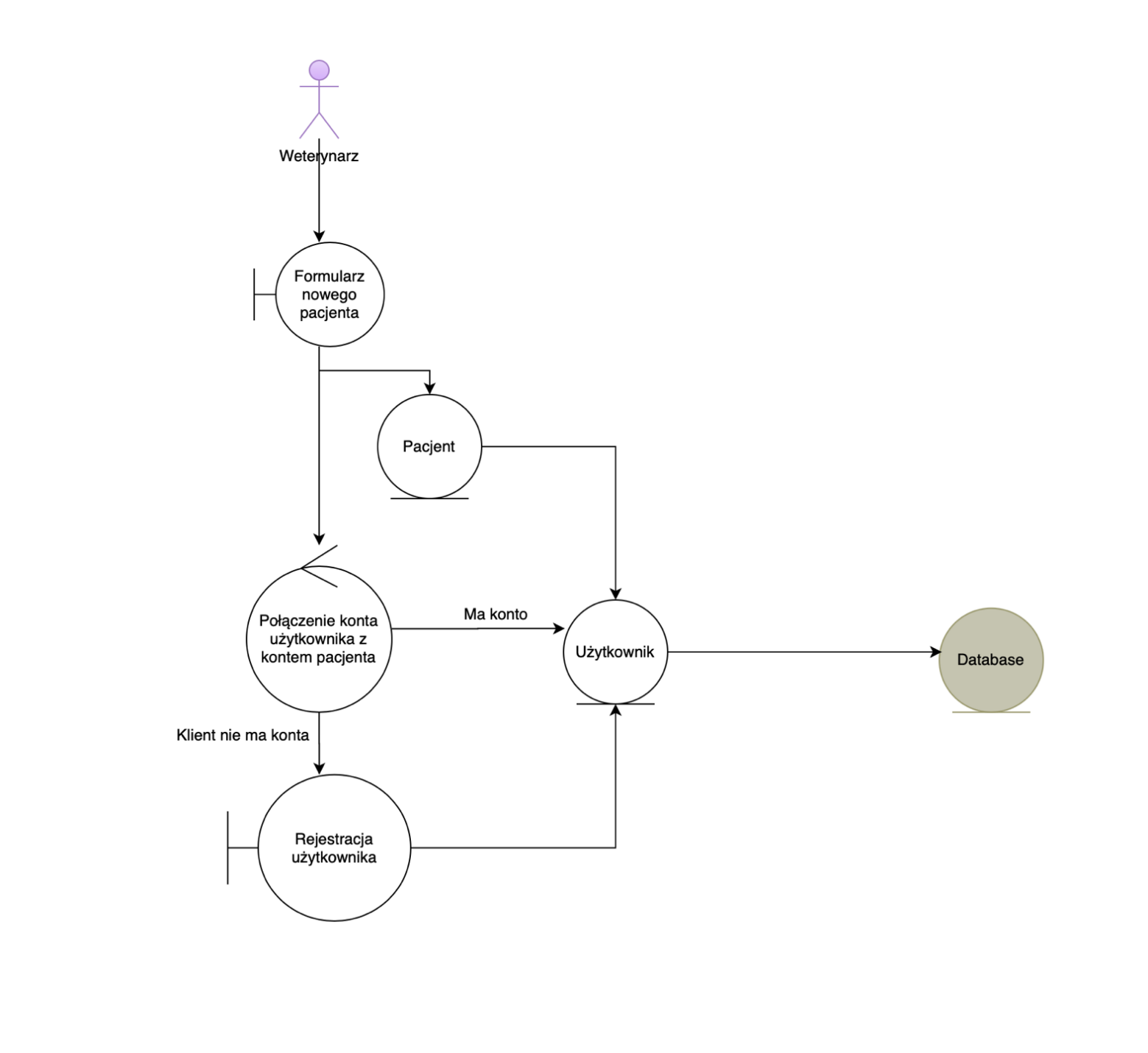
Rejestracja użytkownika



Rejestracja pacjenta



Umówienie wizyty



### Szczegółowy opis wymagań

1. Rejestracja pacjenta

* Pacjent musi posiadać możliwość rejestracji w serwisie, w celu utworzenia spersonalizowanego konta umożliwiającego kontakt z weterynarzami, przegląd informacji i rezerwowanie wizyt.
* Użytkownicy:

-Pacjent

-Administrator

* Scenariusze:

Warunki początkowe – pacjent nie jest zarejestrowany.

Przebieg:

* + 1. Pacjent wchodzi na stronę z rejestracją
    2. Podaje maila, hasło, numer telefonu i dane osobowe
    3. Konieczna jest autoryzacja numeru telefonu
    4. Po udanej autoryzacji administrator zatwierdza konto, oraz generowany jest numer pacjenta

Efekty – pacjent uzyskuje dostęp do wymienionych wcześniej opcji.

Częstotliwość – 5

Istotność – 5

1. Umówienie wizyty

* Pacjent musi posiadać możliwość umówienia wizyty do weterynarza.
* Użytkownicy:

-Pacjent

-weterynarz

* Scenariusze:

Warunki początkowe – pacjent posiada konto.

Przebieg:

* + 1. Pacjent wchodzi na stronę logowania
    2. Podaje maila, hasło
    3. Wchodzi w zakładkę umówione wizyty
    4. Potem używa formularza „Umów wizytę”
    5. Wybiera placówkę, specjalistę, rodzaj wizyty i jej termin
    6. Zatwierdza wprowadzone informację

Efekty – pacjent zostaje umówiony na wizytę i uzyskuje potwierdzenia mailem i sms.

Częstotliwość – 4

Istotność – 4

1. Odwołanie wizyty / Zmiana terminu

* Pacjent musi posiadać możliwość zmiany terminu wizyty.
* Użytkownicy:

-Pacjent

-Weterynarz

* Scenariusze:

Warunki początkowe – pacjent posiada konto.

Przebieg:

* + 1. Pacjent wchodzi na stronę logowania
    2. Podaje maila, hasło
    3. Wchodzi w zakładkę umówione wizyty
    4. Potem używa formularza „Zmień termin / odwołaj”
    5. Wybiera umówioną wcześniej wizytę
    6. Zaznacza „Zmień”, lub „Odwołaj”
    7. Zatwierdza wprowadzone informację (jeżeli wybrał „odwołaj”)
    8. Ustala nowy termin wizyty
    9. Zatwierdza

Efekty – pacjent odwołuje wizytę, lub zostaje mu przydzielony nowy termin

Częstotliwość – 2

Istotność – 3

1. Zapisywanie i pobieranie historii i informacji o pacjencie

* Weterynarz musi posiadać możliwość uzyskania informacji o pacjencie i jego leczeniu.
* Użytkownicy:

-Pacjent

-Weterynarz

* Scenariusze:

Warunki początkowe – pacjent posiada konto.

Przebieg:

* + 1. Weterynarz wchodzi na stronę logowania
    2. Podaje maila, hasło
    3. Wchodzi w zakładkę „pacjenci”
    4. Potem wybiera pacjenta
    5. Wybiera opcję „historia leczenia”, lub „informacje”
    6. Na panelu wyświetlają się dane

Efekty – Weterynarz uzyskuje informacje o pacjencie

Częstotliwość – 4

Istotność – 5

1. Wysyłanie powiadomień o nadchodzącej wizycie

* Pacjent otrzymuje przypomnienie o zbliżającym się terminie wizyty w celu uniknięcia jej przeoczenia.
* Użytkownicy:

-Pacjent

* Scenariusze:

Warunki początkowe – pacjent posiada konto i ma umówioną wizytę.

Przebieg:

* + 1. System wykrywa, że do wizyty zostało 3 dni
    2. Pobiera informację o wizycie i numer telefonu pacjenta
    3. Generuje schemat przypomnienia i wysyła go na numer telefonu podany przez pacjenta

Efekty – Pacjent jest informowany o zbliżającej się wizycie

Częstotliwość – 5

Istotność - 3

## Wymagania niefunkcjonalne

wobec całego systemu

1. Wydajność

- Zewnętrzny dostawca usług serwerowych zapewniający wysoką jakość wydajności podczas obsługiwania i udostępniania danych.

-Oprogramowanie zoptymalizowane pod konkretną weterynarię zapewnia wysoką wydajność

1. Bezpieczeństwo

-Dane pacjentów wraz z informacjami odnośnie historii i przebiegu leczenia zabezpieczone przez zewnętrzną firmę.

-Dane są przetwarzane i przechowywane zgodnie z przepisami RODO

-Każda informacja ma wykonywany backup na dedykowanym do tego serwerze.

1. Zabezpieczenia

-Zewnętrzny dostawca usług serwerowych z certyfikatem ISO/IEC 27001 zapewniający najwyższe standardy bezpieczeństwa.

-Organizacja szkoleń z dziedziny bezpiecznego przetwarzania i magazynowania danych dla kadr obsługujących oprogramowanie.

1. Inne cechy jakości

-Prosty i przejrzysty interfejs użytkownika (łatwa adaptacja pracowników w obsłudze systemu)

-Servicedesk ułatwiający obsługę i eliminację błędów (konserwacja)

-Dostosowanie oprogramowania pod konkretnego klienta zmniejsza ryzyko

awarji

# Zarządzanie projektem

## Zasoby ludzkie

* Zespół programistów rozwijających główne funkcjonalności serwisu liczący minimum 2 osoby.
* Minimum jedna osoba zajmująca się rozwijaniem strony internetowej.
* Tester, sprawdzający wdrażane funkcjonalności oraz poprawne działanie strony internetowej
* Firma zewnętrzna, która zajmie się pozycjonowaniem serwisu.
* Firma zewnętrzna, która stworzy projekt UI/UX strony internetowej oraz zajmie się identyfikacją wizualną i marketingiem w sieci.

## Harmonogram prac

***Numer tygodnia***

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Zadania | 1-2 | 3-4 | 5-6 | 7-8 | 9-10 | 11-12 | 13-14 | 15-16 | 17-18 |
| Spotkanie i założenia wstępne | x |  |  |  |  |  |  |  |  |
| wstępna dokumentacja |  | x | x |  |  |  |  |  |  |
| podział prac |  |  |  | x |  |  |  |  |  |
| tworzenie prototypu |  |  |  |  | x | x | x |  |  |
| testy |  |  |  |  |  |  | x | x |  |
| wdrożenie |  |  |  |  |  |  |  |  | x |

## Etapy/kamienie milowe projektu

dla głównych etapów projektu

1. Szczegółowa dokumentacja
2. Prototyp projektu
3. Testy
4. Gotowy produkt
5. Wdrożenie

# Zarządzanie ryzykiem

## Lista czynników ryzyka

Głównym ryzykiem jest niechęć korzystania z aplikacji, przez  osoby nie będące przekonane do nowych technologii.    
Może to skutkować:

1) większym nakładem pracy przy rejestracji klientów (zakładając, że 60% klientów będzie chciało korzystać z aplikacji, a 40% będzie wolało umówić się w 'tradycyjny' sposób np. telefonicznie, to będzie trzeba optymalnie to skonfigurować, żeby nie dochodziło do błędów, takich jak nakładające się na siebie terminy wizyt.

2) zmniejszeniem popytu na usługi kliniki (rezygnacja z usługi, ze względu na przyzwyczajenie do bezpośredniego kontaktu z rejestracją, brak zaufania do aplikacji)

## Ocena ryzyka

Bardziej prawdopodobnym podpunktem ryzyka jest zwiększenie nakładów pracy przy rejestracji pacjentów. Nie wpływa to na pracę aplikacji, a na pracę samej kliniki, która może mieć przez to dodatkową ilość danych do wprowadzenia do aplikacji.

## Plan reakcji na ryzyko

Rekomendowane strategie:

1. Zamieszczenie na stronie kliniki instrukcji w formie video/prostej grafiki jak korzystać z aplikacji
2. Przeszkolenie pracowników kliniki, pod kątem znajomości aplikacji, tak aby umożliwić im zachęcanie i przyuczanie klientów jak prawidłowo z niej korzystać
3. Informacja audio o aplikacji internetowej przy łączeniu się z infolinią kliniki

# 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):

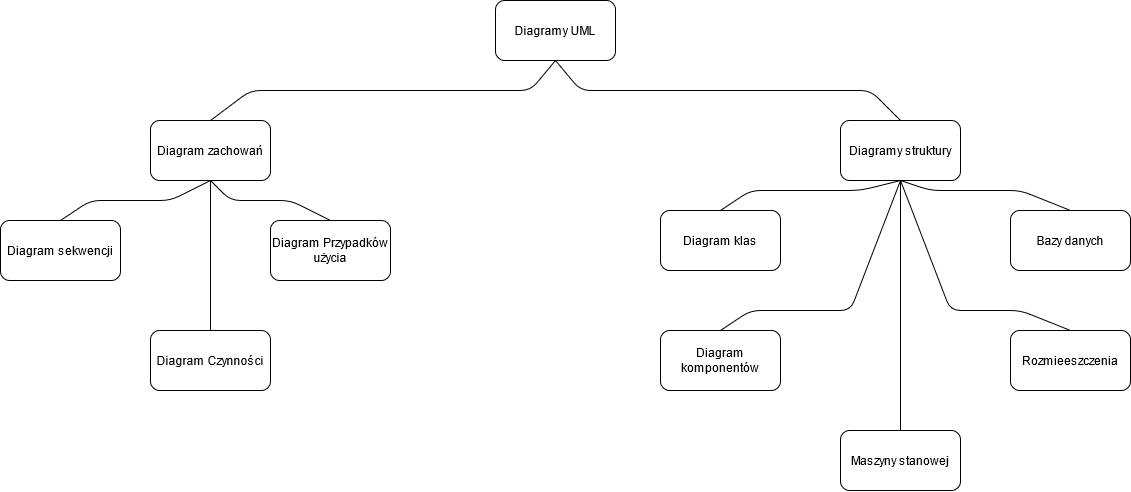
* **Numer 1**
* Testy klas i komponentów
* Testy weryfikacyjne
* Testowanie poprawności budowy klas, wraz z ich funkcjonalnością
* Testowanie wykonuje członek dwuosobowego zespołu programistów
* Test rozpoczyna się w 13-14 tygodniu prac
* Do wykonywania testu zostaną użyte testy jednostkowe
* Klasy powinny być usystematyzowane, nie powinny duplikować informacji, nie powinny zawierać luk, być odporne na wprowadzanie niepoprawnych danych.

Do testów zostanie wykorzystane testy jednostkowe, dane wejściowe poza typowymi danymi wprowadzanymi podczas tworzenia użytkownika, rezerwacji wizyty i konstruowania historii leczenia, będą wprowadzane również takie zawierające niepoprawne dane i typy.

* Zestaw danych:
* **Numer 2**
* Testy czarnoskrzynkowe
* Testy funkcjonalne
* Testowanie funkcjonalności systemu przez osobę nie znającą struktury systemu
* Testowanie wykonuje członek nie związany z zespołem programistów
* Test rozpoczyna się w 13-14 tygodniu prac
* Testy wykonywane są bezpośrednio na kompletnym systemie w środowisku testowym
* Dokładna znajomość kodu programu, struktury wewnętrznej i ogólnej wiedzy programistycznej nie jest wymagana. Osoba testująca jest świadoma tego, co oprogramowanie powinno robić, ale nie wie, jak to się odbywa. Zdaje sobie sprawę, że dane wejściowe zwracają pewne dane wyjściowe, ale nie skupia się na tym, w jaki sposób oprogramowanie generuje dane wyjściowe, ponieważ to nie jest przedmiotem jej testów
* Przypadki testowe oparte są na specyfikacjach i wymaganiach, tzn. na tym co program ma robić. Zazwyczaj pochodzą one z zewnętrznych opisów oprogramowania, specyfikacji wymagań lub ustaleń projektowych. Tester wybiera zarówno poprawne, jak i nieprawidłowe dane wejściowe i ocenia dane wyjściowe, zazwyczaj na podstawie wyniku, o którym wiadomo że jest poprawny
* Zestaw danych:
* 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

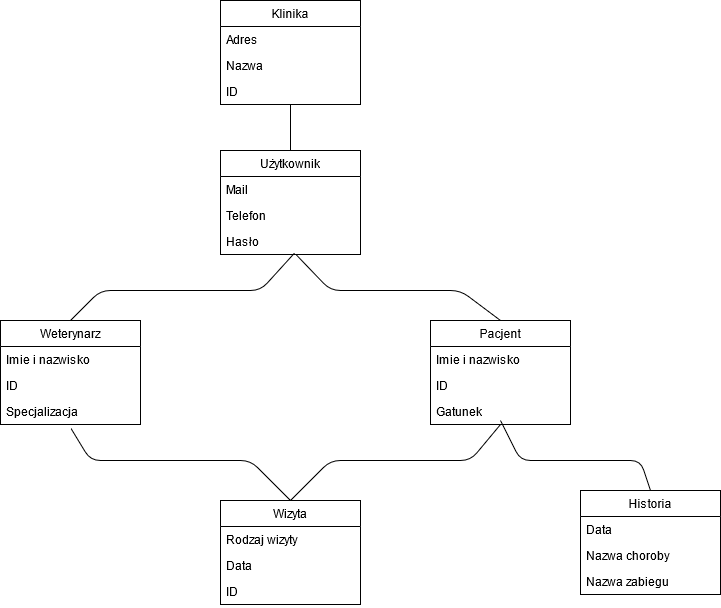


## Technologie implementacji systemu

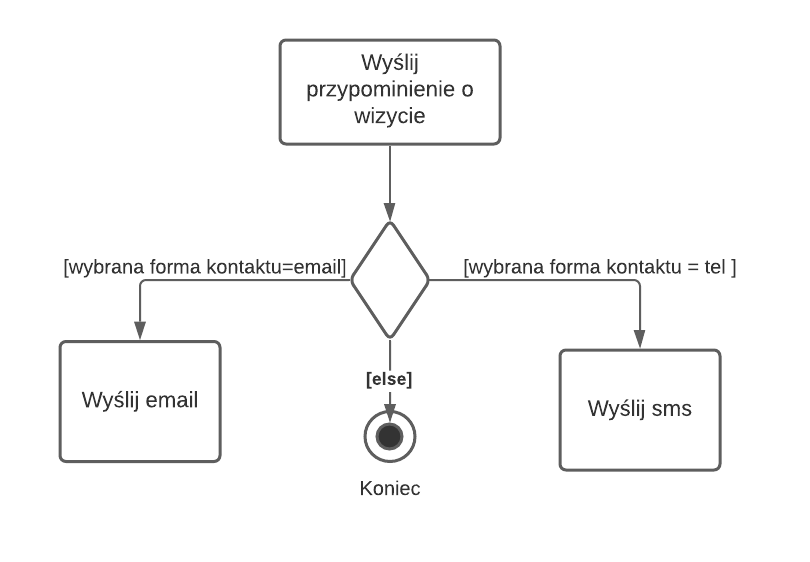
|  |  |
| --- | --- |
| Django | Podstawowa technologia wykorzystywana do tworzenia witryny internetowej z widokami i obsługą bazy danych |
| HTML | Niezbędne do budowania strony internetowej |
| Bootstrap | Framework ułatwiający i optymalizujący prace przy tworzeniu widoku strony |
| SQL | Język wykorzystywany w obsłudze baz danych |

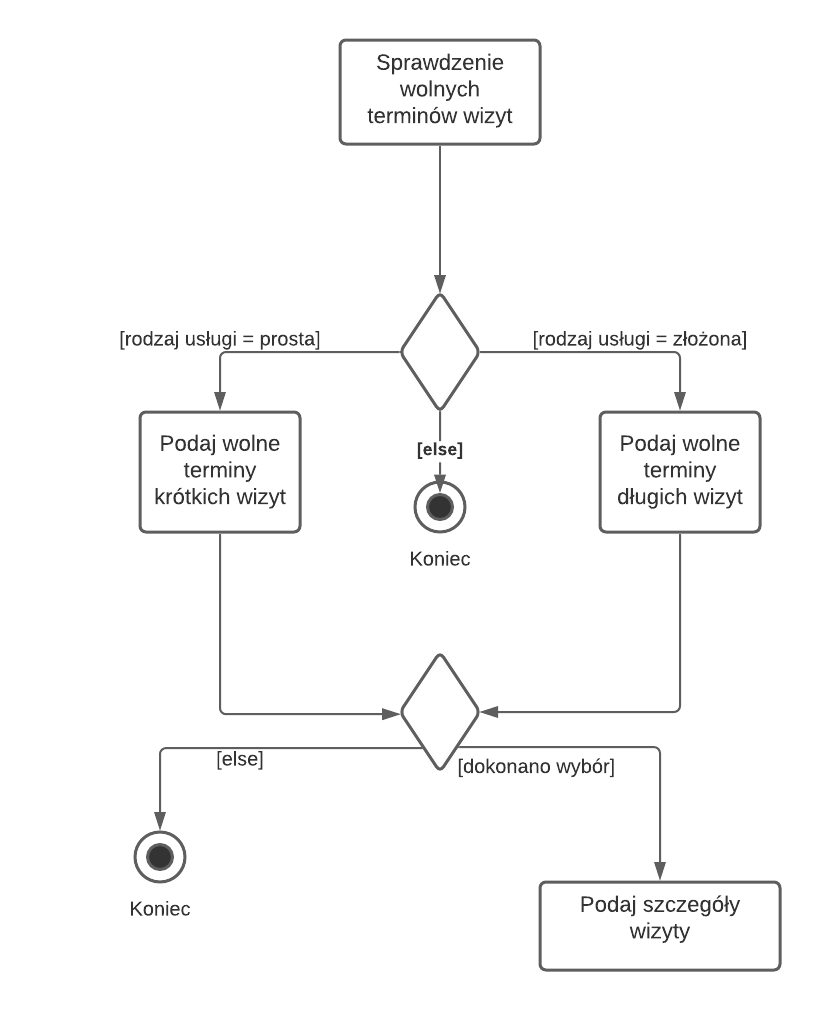
## Diagramy UML

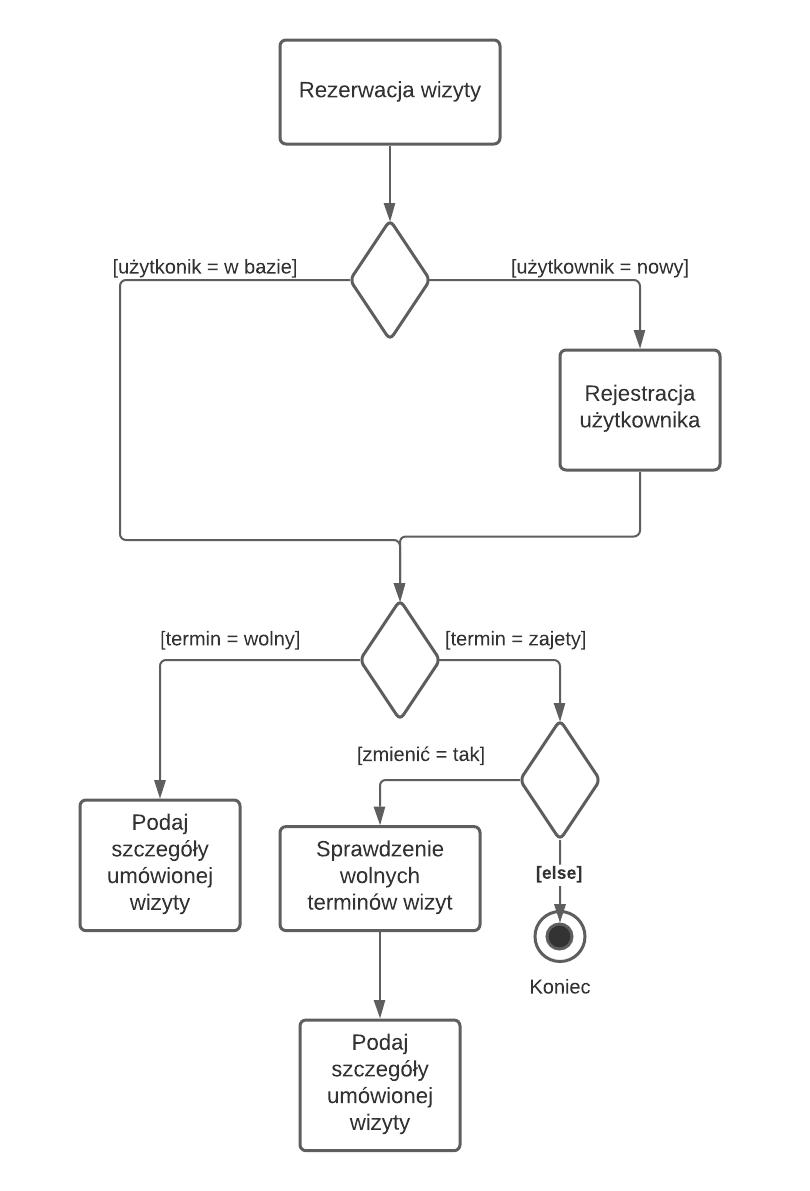
### Diagram(-y) klas



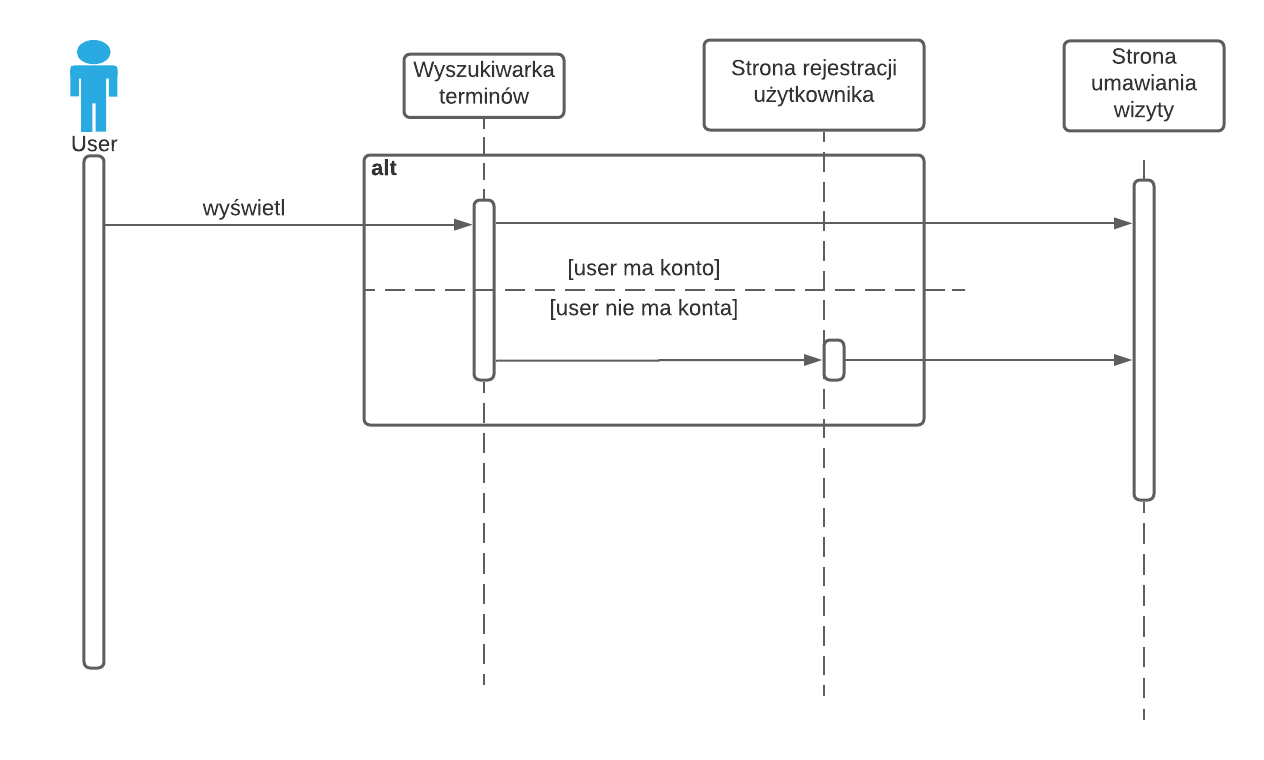
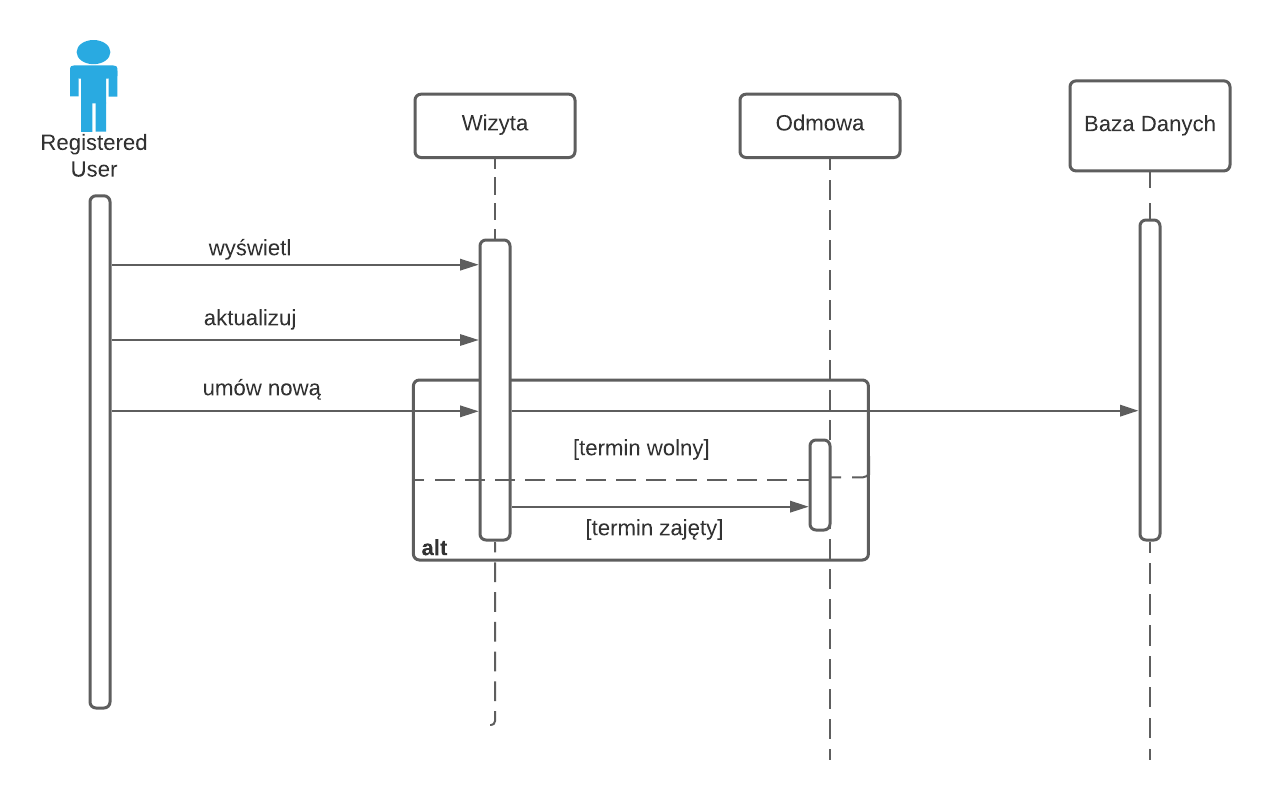
### Diagram(-y) czynności

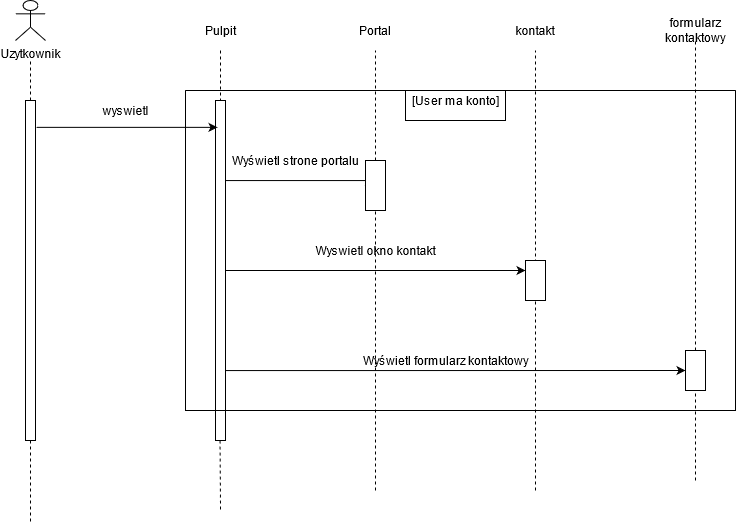
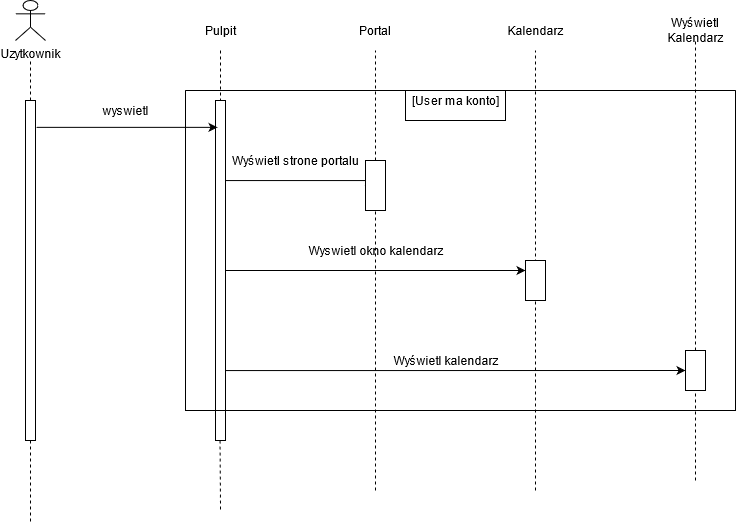
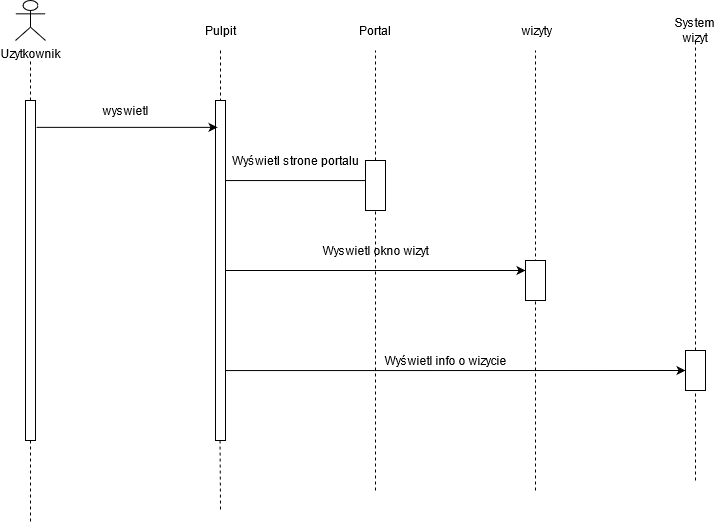




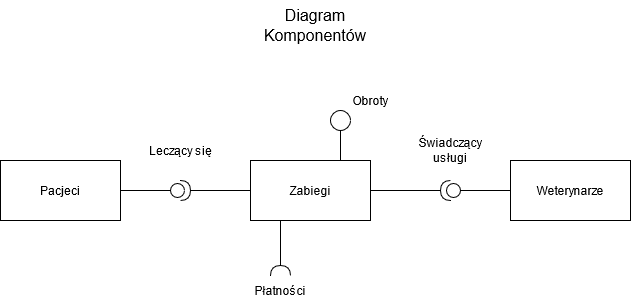


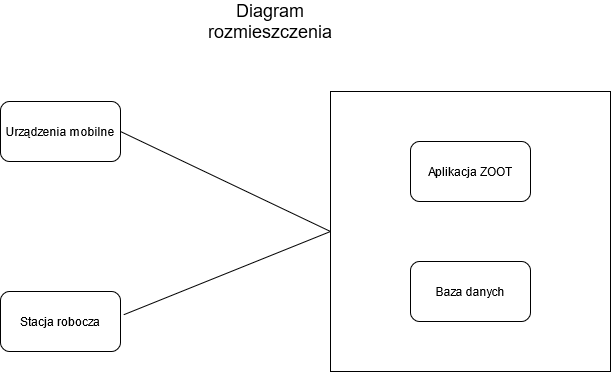
### Diagramy sekwencji

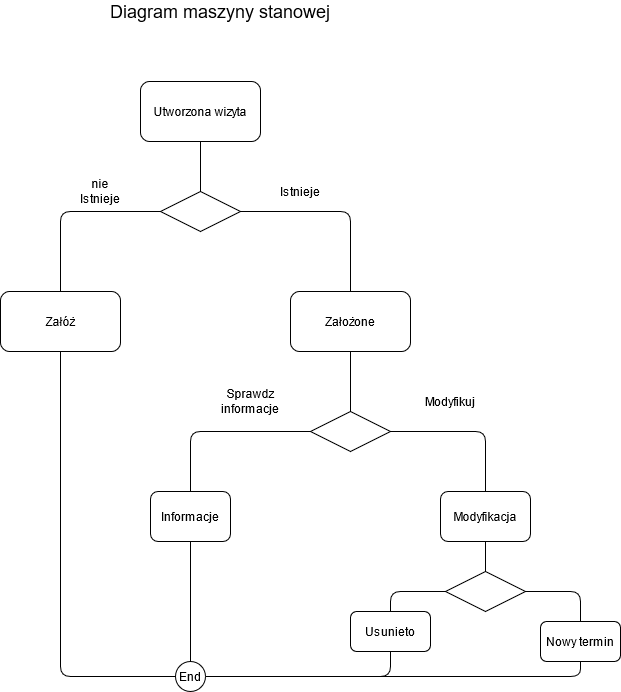




### Inne diagramy





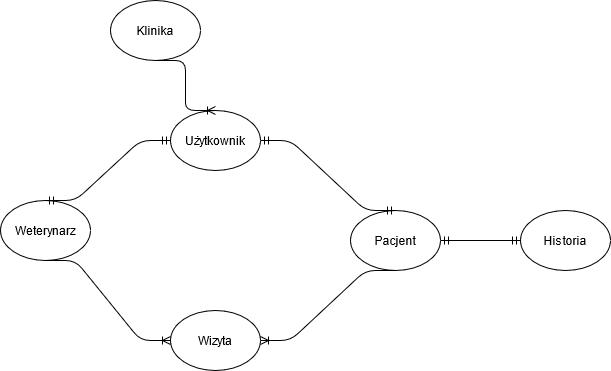


## Charakterystyka zastosowanych wzorców projektowych

* Wzorzec Wrapper – wzorzec strukturalny projektowy jego celem jest umożliwienie współpracy klasom (Diagram maszyny stanowej)
* Wzorzec Builder – wzorzec kreacyjny projektowy jego celem jest rozdzielenie sposobu tworzenia obiektów od ich reprezentacji (Diagram umówienie wizyty)

## Projekt bazy danych

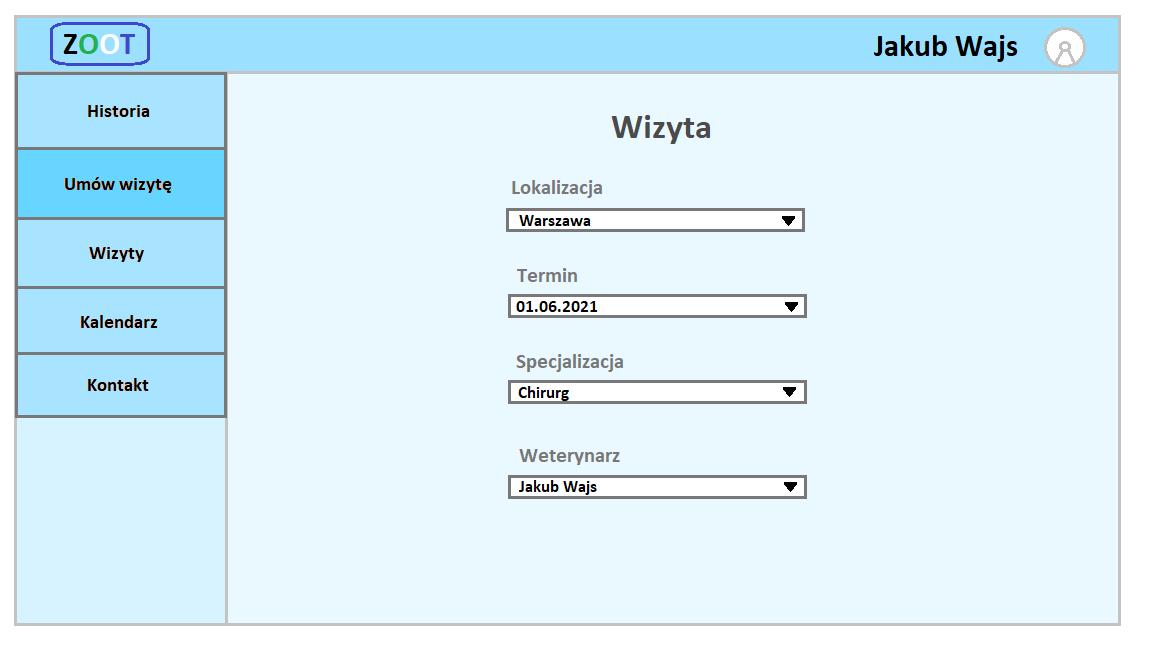
### Schemat



### Projekty szczegółowe tabel



## Projekt interfejsu użytkownika



### Lista głównych elementów interfejsu

* Okno historii wizyt
* Okno do umawiania wizyty
* Okno odbytych wizyt
* Okno kalendarza
* Okno kontaktowe
* Ikona ustawień konta

### Przejścia między głównymi elementami

Ekran logowania -> Pulpit użytkownika -> zakładka opcji(Umów wizytę) -> formularz informacji do wizyty -> zatwierdzenie enterem

### Projekty szczegółowe poszczególnych elementów

1. Logowanie



* wykorzystywane są dane do weryfikacji maila i hasła
* po wpisaniu maila i hasła należy wcisnąć przycisk zaloguj.

2. Historia wizyt

Obraz zawierający tekst

Opis wygenerowany automatycznie

* po kliknięciu w zakładkę historia w centralnej części pojawia się historia odbytych wizyt

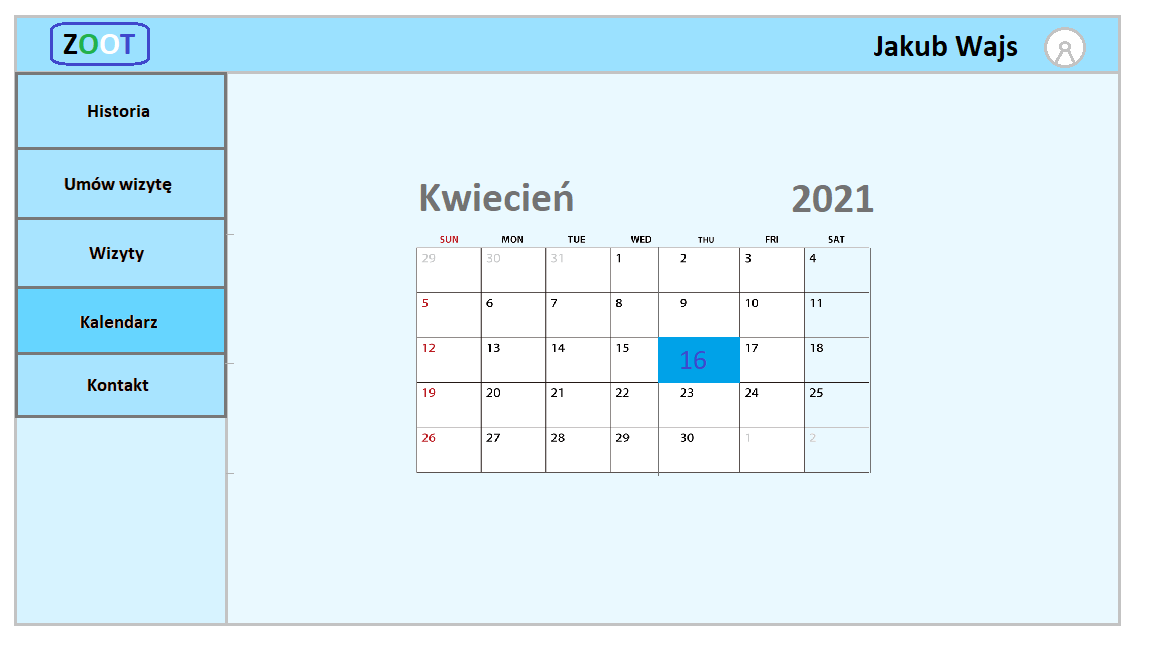
1. Wizyty

Obraz zawierający tekst

Opis wygenerowany automatycznie

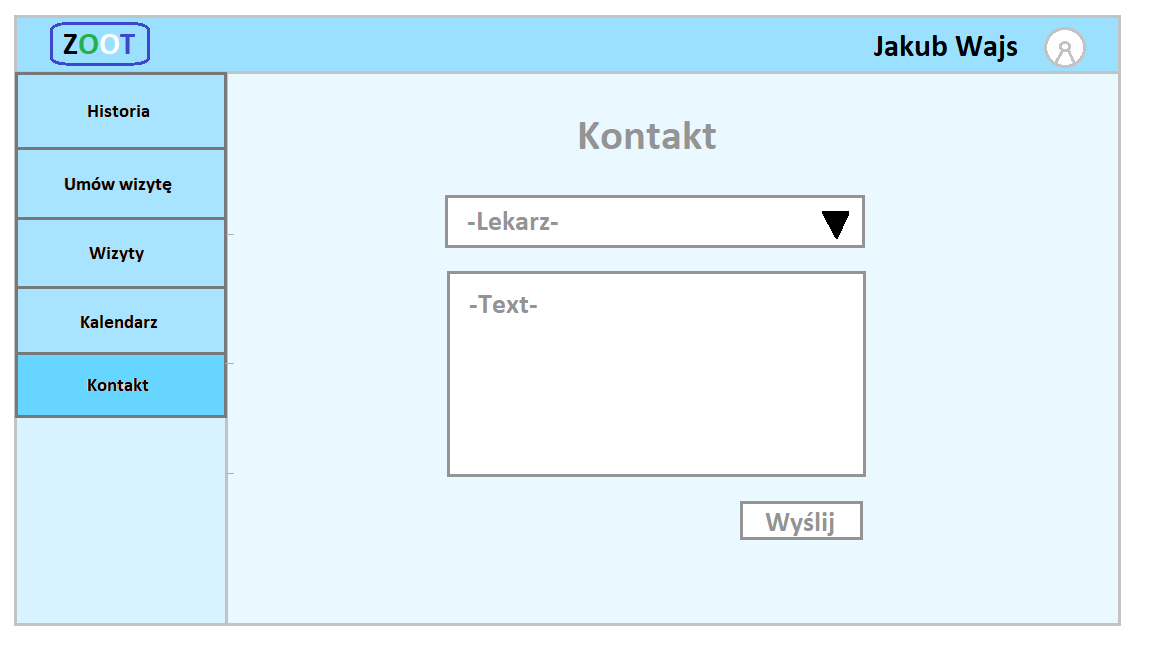
* po kliknięciu w wizyty wyświetla się lista nadchodzących wizyt

5. Kalendarz



* Po kliknięciu w zakładkę kalendarz wyświetla się kalendarz wizyt

6. Kontakt



* Po kliknięciu w zakładkę kontakt wyświetla formularz do kontaktu z weterynarzem

## Procedura wdrożenia

 Przygotowanie dokumentacji powykonawczej.

 Przygotowanie i skonfigurowanie infrastruktury technicznej.

 Zainstalowanie i skonfigurowanie systemu informatycznego do testów.

 Testowanie systemu.

 Testowa migracja danych.

 Testy migracji oraz systemu w zakresie zmigrowanych danych

# 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 – udział każdej osoby w skali całego projektu

# Inne informacje

przydatne informacje, które nie zostały ujęte we wcześniejszych punktach