

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

1 Spis treści

2	Odnosniki do innych źródeł.....	4
3	Słownik pojęć	4
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 нефunkcjonalne.....	17
6	Zarządzanie projektem	18
6.1	Zasoby ludzkie	18
6.2	Harmonogram prac.....	18
6.3	Etapy/kamienie milowe projektu	18
7	Zarządzanie ryzykiem.....	19
7.1	Lista czynników ryzyka	19
7.2	Ocena ryzyka.....	19
7.3	Plan reakcji na ryzyko	19
8	Zarządzanie jakością.....	19
8.1	Scenariusze i przypadki testowe	20
9	Projekt techniczny	22
9.1	Opis architektury systemu.....	22
9.2	Technologie implementacji systemu.....	22
9.3	Diagramy UML	22
9.4	Charakterystyka zastosowanych wzorców projektowych.....	30
9.5	Projekt bazy danych	30
9.6	Projekt interfejsu użytkownika.....	31
9.7	Procedura wdrożenia	36
10	Dokumentacja dla użytkownika.....	37
11	Podsumowanie	38
11.1	Szczegółowe nakłady projektowe członków zespołu	38
12	Inne informacje	39

2 Odnośniki do innych źródeł

- Zarządzania projektem – monday.com
- Wersjonowanie kodu – sugerowany Git, [Github.com](https://github.com)
- System obsługi defektów – monday.com, Github

3 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

4 Wprowadzenie

4.1 Cel dokumentacji

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

4.2 Przeznaczenie dokumentacji

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

4.3 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	<ul style="list-style-type: none">• Stale wzrastające zapotrzebowanie na usługi weterynaryjne• Zmniejszenie marginesu błędu przy obsłudze pacjentów• Przyjazna obsługa programu	<ul style="list-style-type: none">• Produkt nie jest uniwersalny, tylko dopasowany do konkretnej kliniki• Konieczność szkolenia kadry z zakresu obsługi oprogramowania (zmieniającego się w czasie)• Dodatkowe koszty
Zewnętrzne	<ul style="list-style-type: none">• Zwiększenie zainteresowania usługami danej kliniki weterynaryjnej ze względu na nowoczesną/innovacyjną aplikację• Deficyt zaawansowanego oprogramowania, spersonalizowanego do potrzeb rynku branży weterynaryjnej	<ul style="list-style-type: none">• Niskie zainteresowanie produktem ze względu na przyzwyczajenie do starszych, ale bardziej uniwersalnych systemów• Luka technologiczna w społeczeństwie

5 Specyfikacja wymagań

5.1 Charakterystyka ogólna

5.1.1 Definicja produktu

System zorientowany w technologii webowej, do obsługi kliniki weterynaryjnej.

5.1.2 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.

5.1.3 Cel biznesowy

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

5.1.4 Użytkownicy

- Weterynarz
- Pacjent
- Właściciel
- Administrator

5.1.5 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.

5.1.6 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 otwartoźródłowej 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.

5.2 Wymagania funkcjonalne

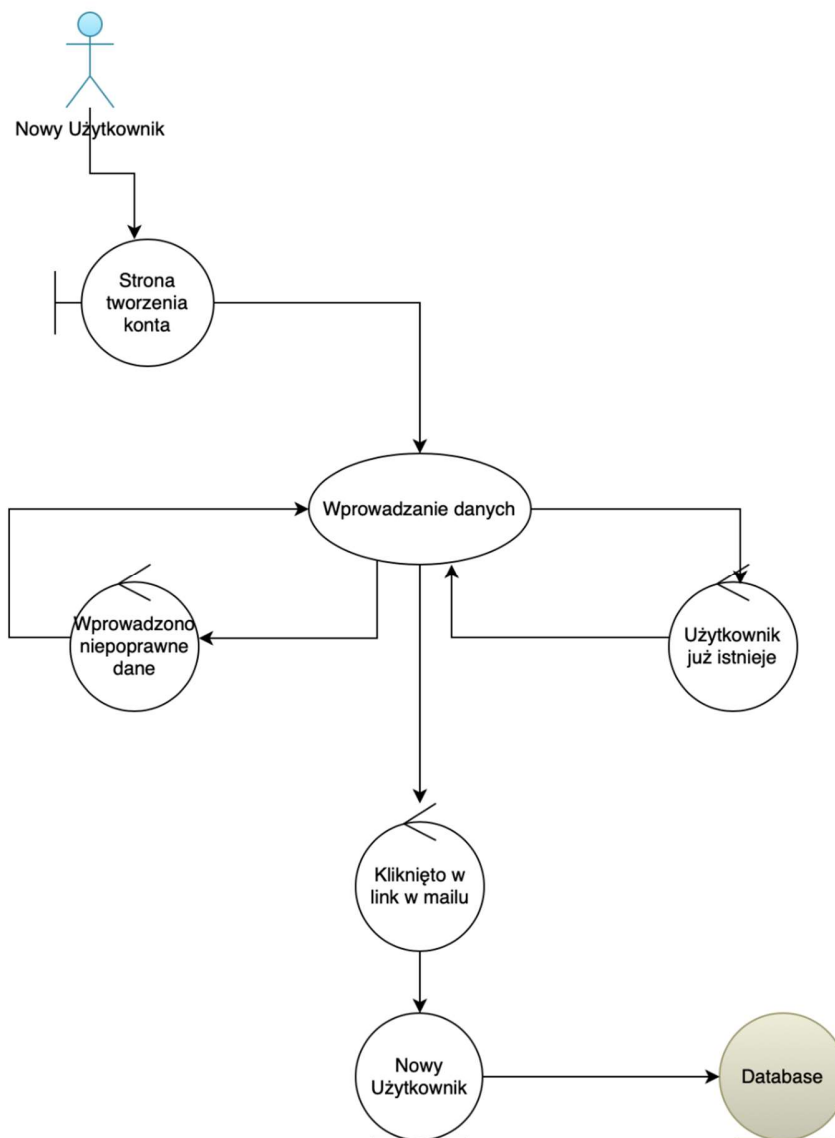
5.2.1 Lista wymagań

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

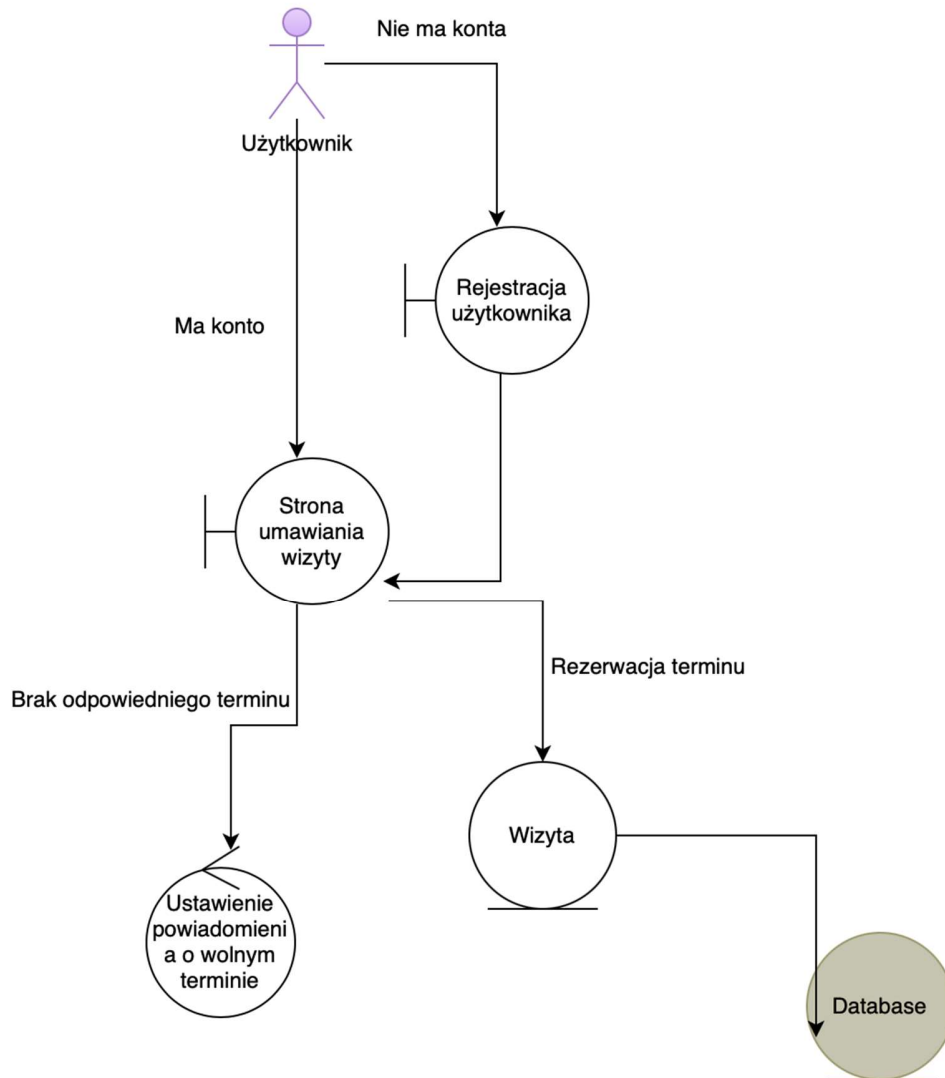
5.2.2 Diagramy przypadków użycia

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

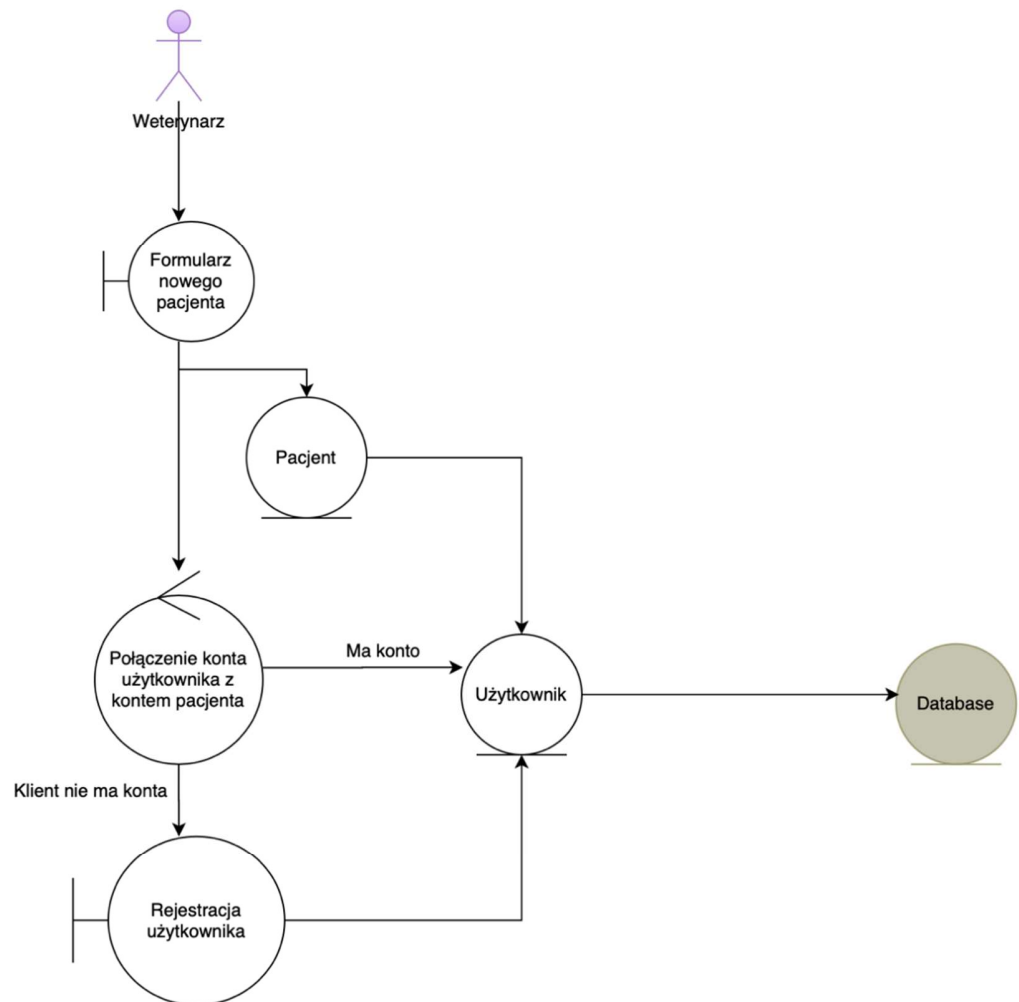
Rejestracja użytkownika



Rejestracja pacjenta



Umówienie wizyty



5.2.3 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

2. 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 informacje

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

Częstotliwość – 4

Istotność – 4

3. 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 informacje (jeżeli wybrał „odwołaj”)
8. Ustala nowy termin wizyty
9. Zatwierdza

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

Częstotliwość – 2

Istotność – 3

4. 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

5. 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

5.3 Wymagania niefunkcjonalne

wobec całego systemu

1. Wydajność

- Zewnętrzny dostawca usług serwerowych zapewniający wysoką jakość wydajności podczas obsługi i udostępniania danych.
- Oprogramowanie zoptymalizowane pod konkretną weterynarię zapewnia wysoką wydajność

2. 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.

3. 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.

4. 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 awarii

6 Zarządzanie projektem

6.1 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.

6.2 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

6.3 Etapy/kamienie milowe projektu

dla głównych etapów projektu

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

7 Zarządzanie ryzykiem

7.1 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)

7.2 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.

7.3 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

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):

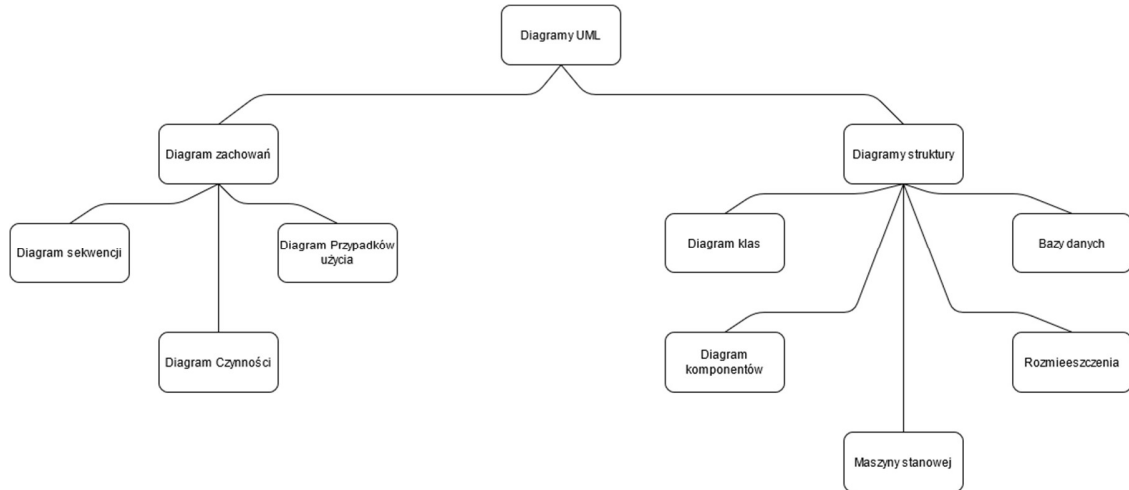
- **Numer 1**
- Testy klas i komponentów
- Testy weryfikacyjne
- Testowanie poprawności budowy klas, wraz z ich funkcjonalnością
- Testowanie wykonuje członek dwuosobowego zespołu programistów
- Test rozpoczyna się w 13-14 tygodniu prac
- Do wykonywania testu zostaną użyte testy jednostkowe
- Klasy powinny 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*

9 Projekt techniczny

9.1 Opis architektury systemu

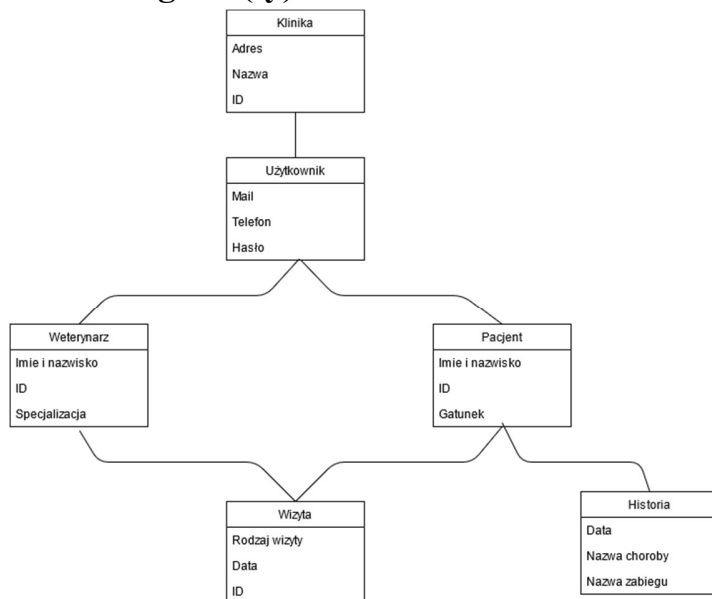


9.2 Technologie implementacji systemu

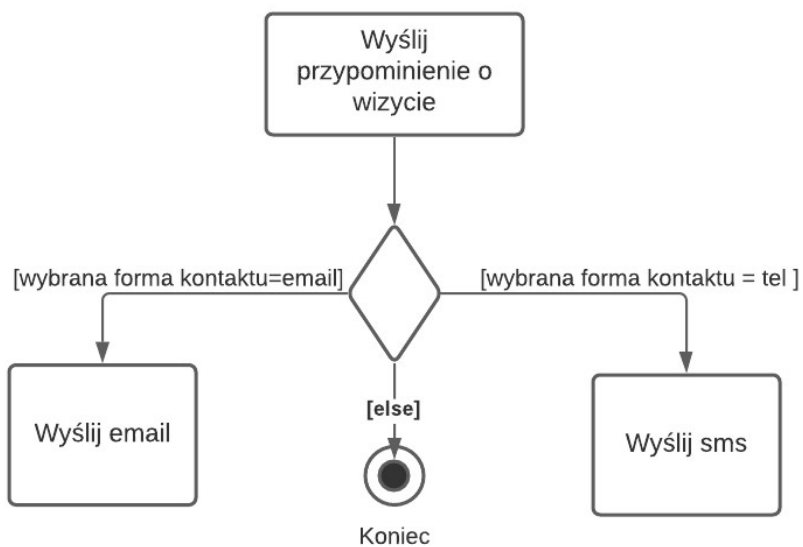
Django	Podstawowa technologia wykorzystywana do tworzenia witryny internetowej z widokami i obsługą bazy danych
HTML	Niezbędne do budowania strony internetowej
Bootstrap	Framework ułatwiający i optymalizujący prace przy tworzeniu widoku strony
SQL	Język wykorzystywany w obsłudze baz danych

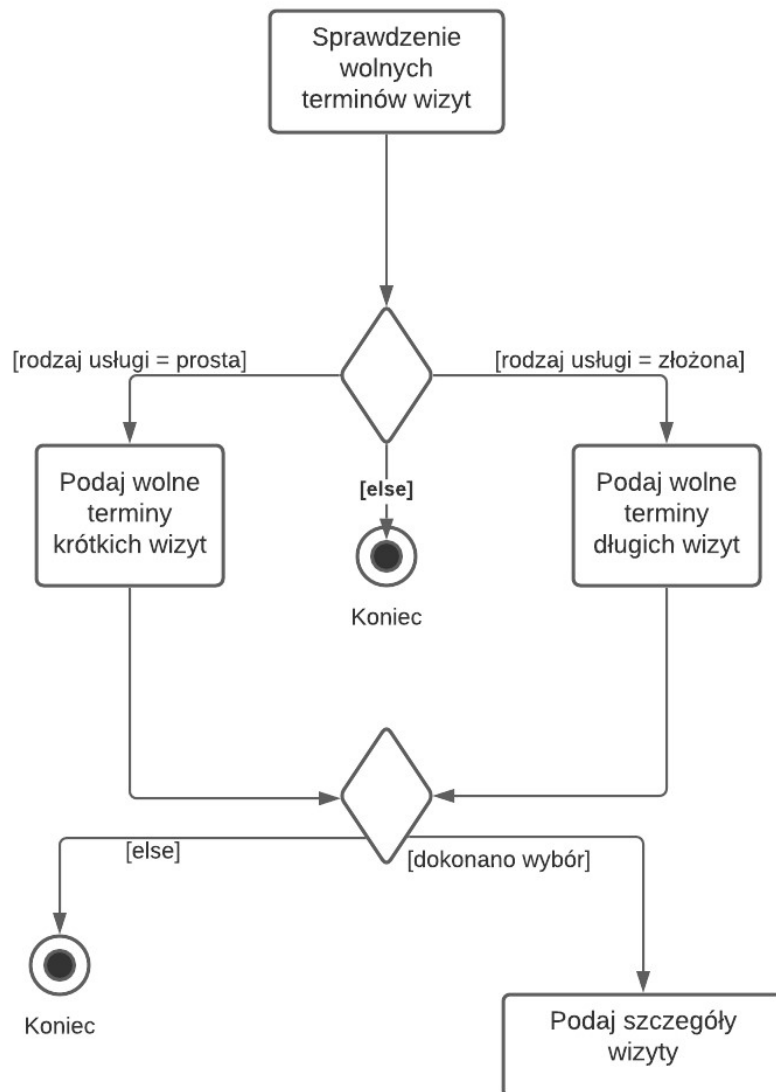
9.3 Diagramy UML

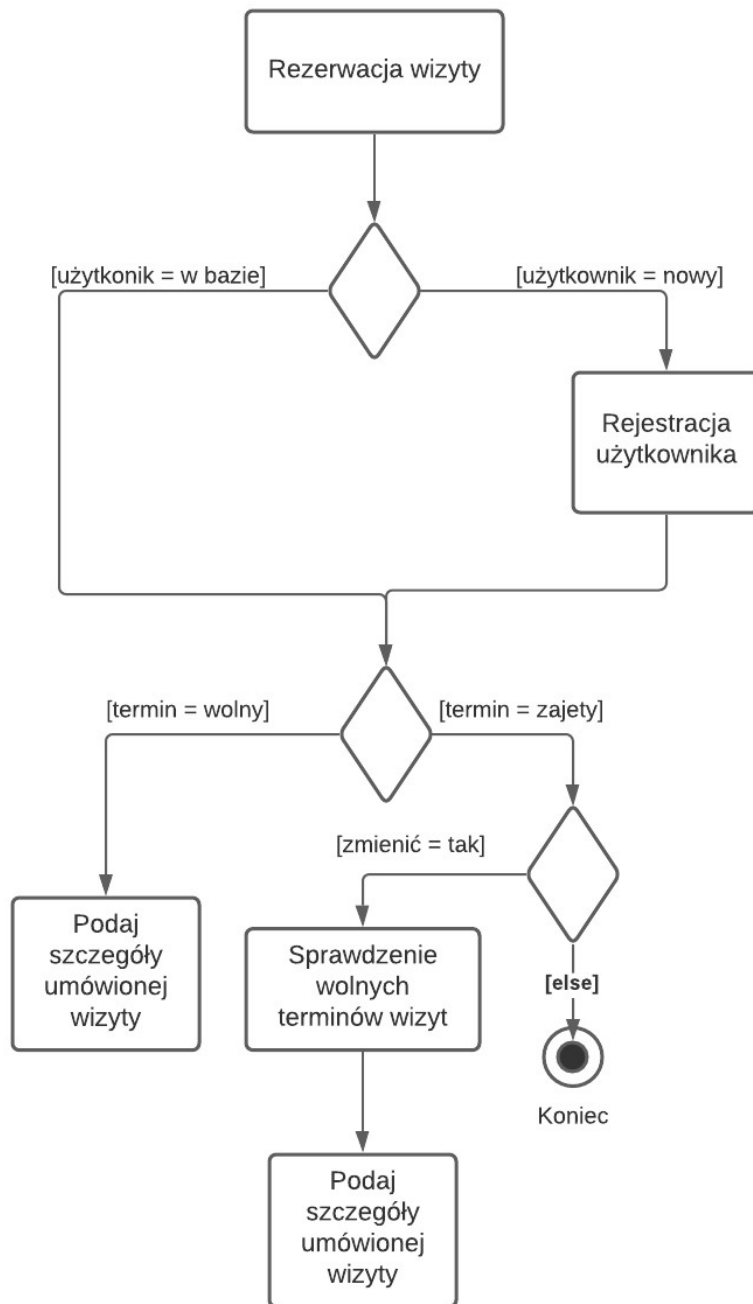
9.3.1 Diagram(-y) klas



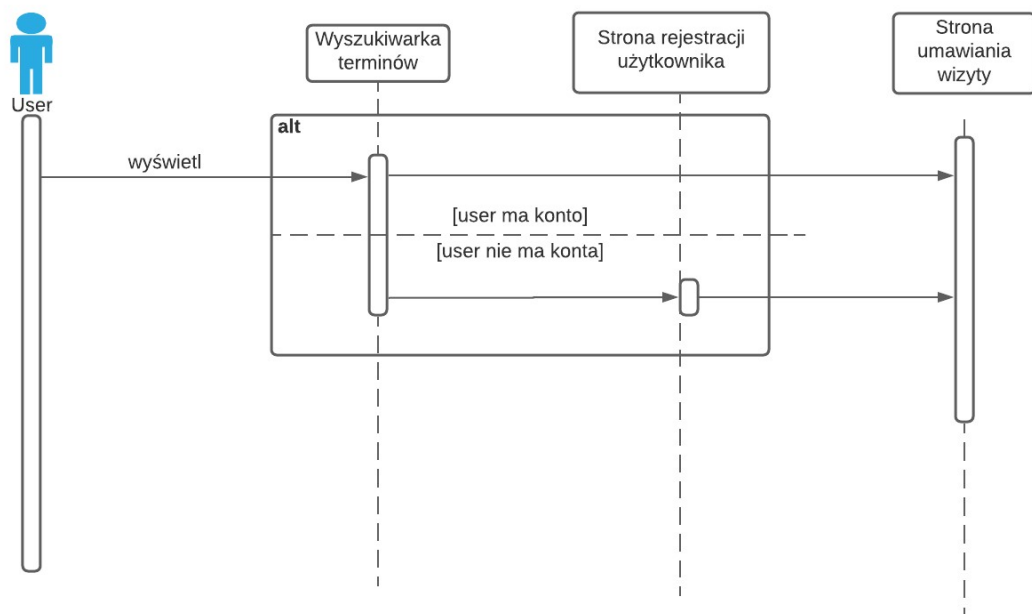
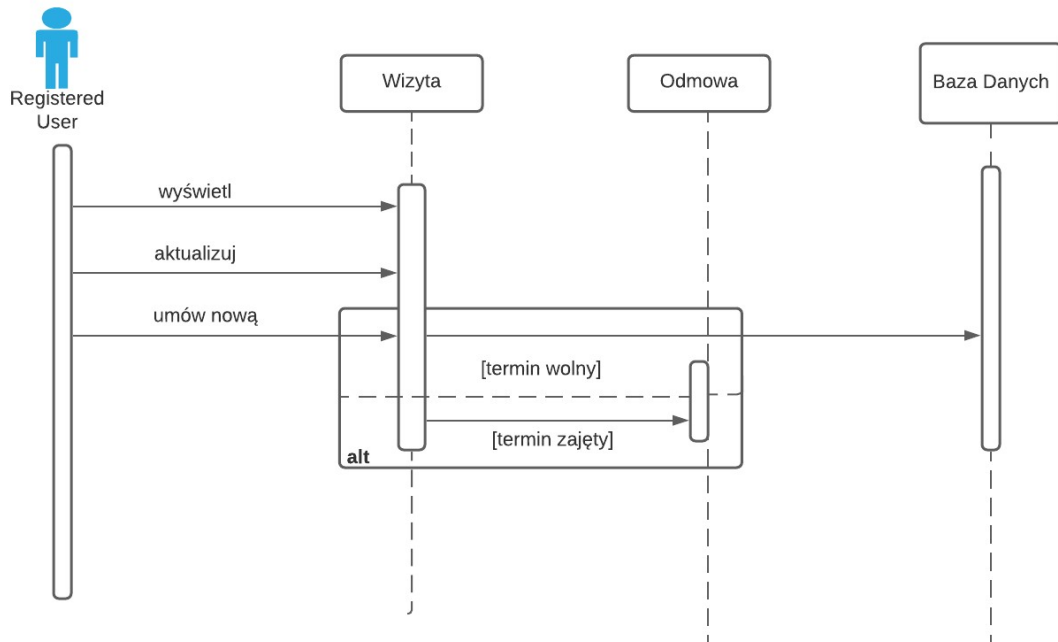
9.3.2 Diagram(-y) czynności

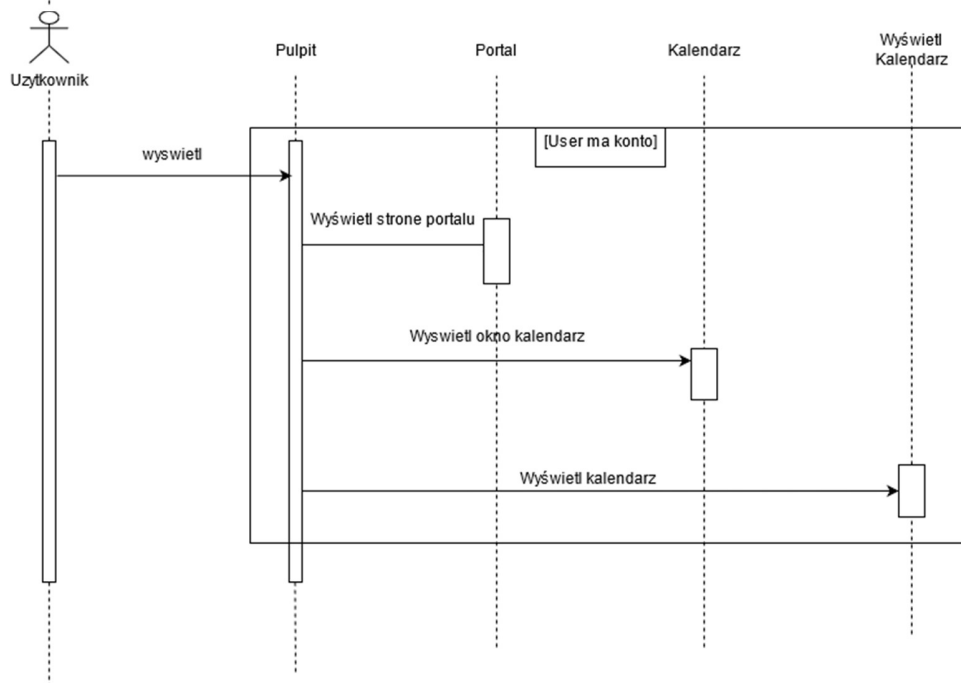
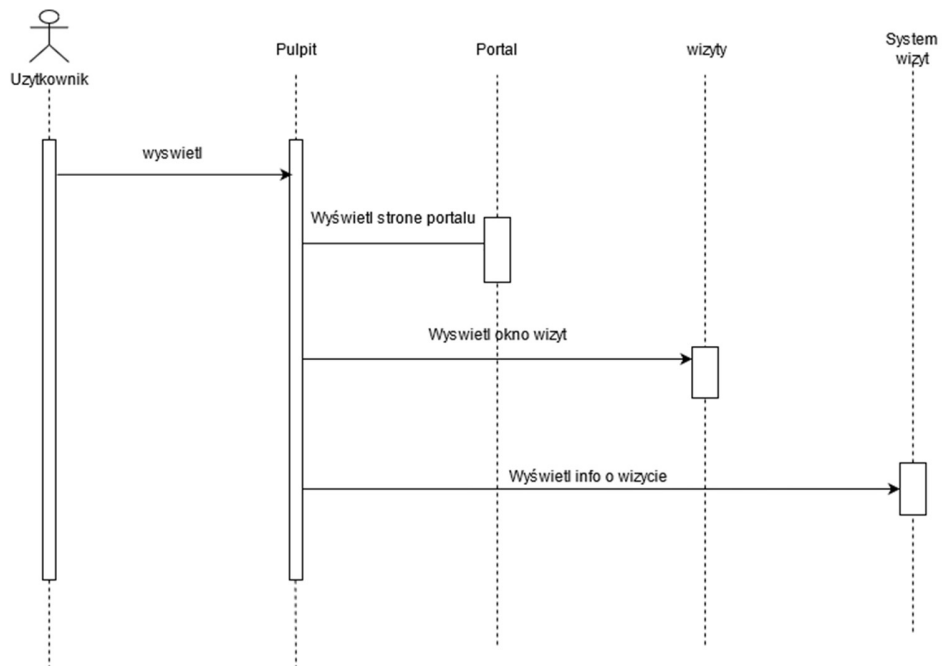


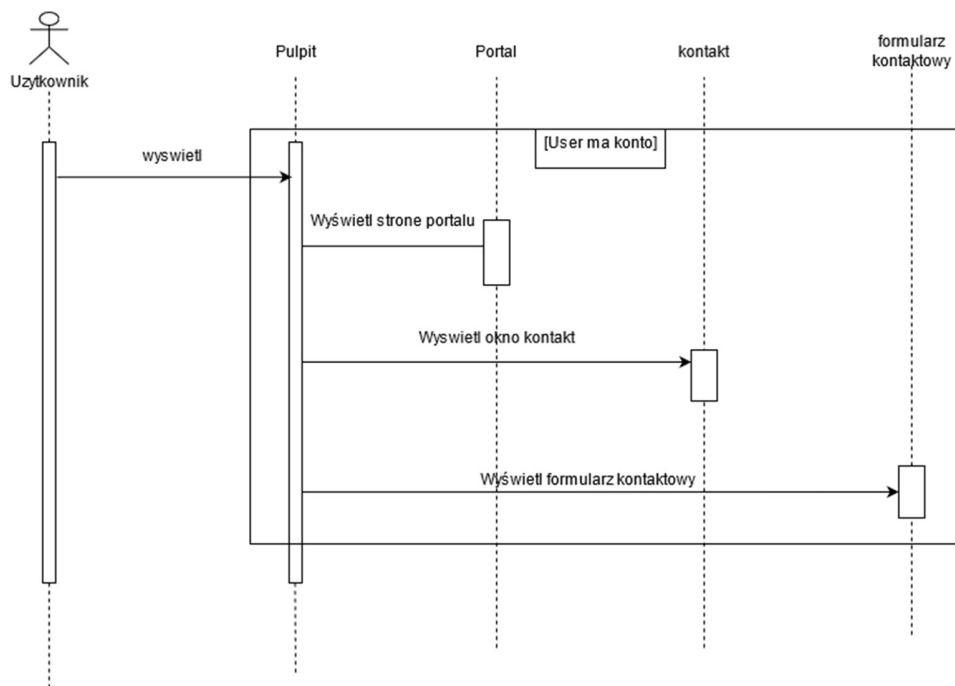




9.3.3 Diagramy sekwencji







9.3.4 Inne diagramy

Diagram
Komponentów

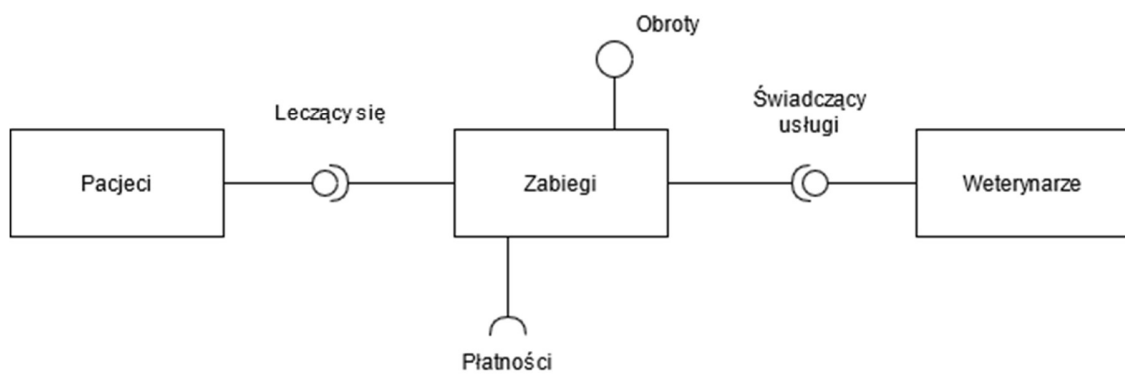


Diagram
rozmieszczenia

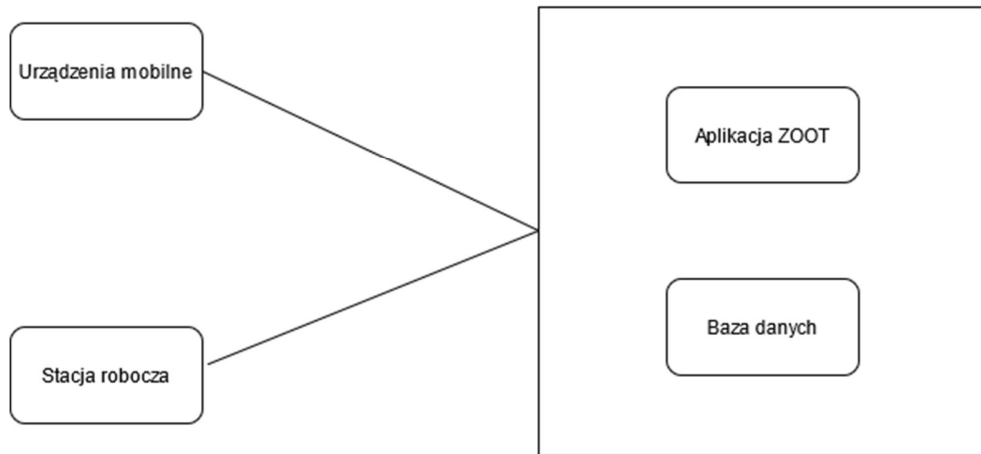
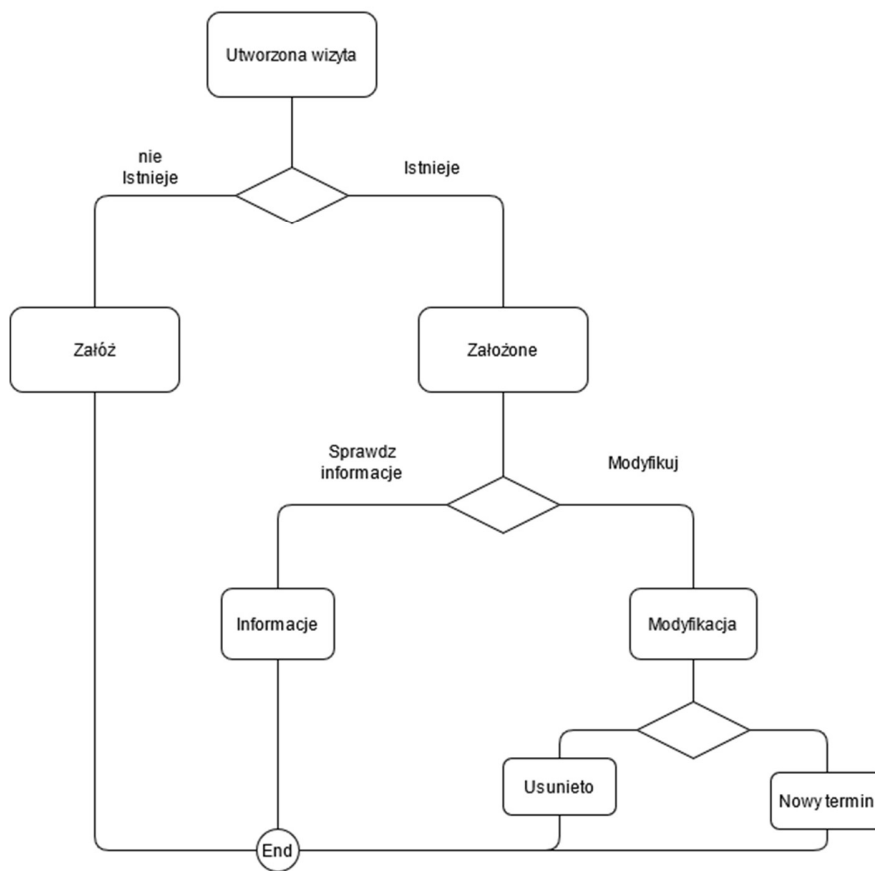


Diagram maszyny stanowej

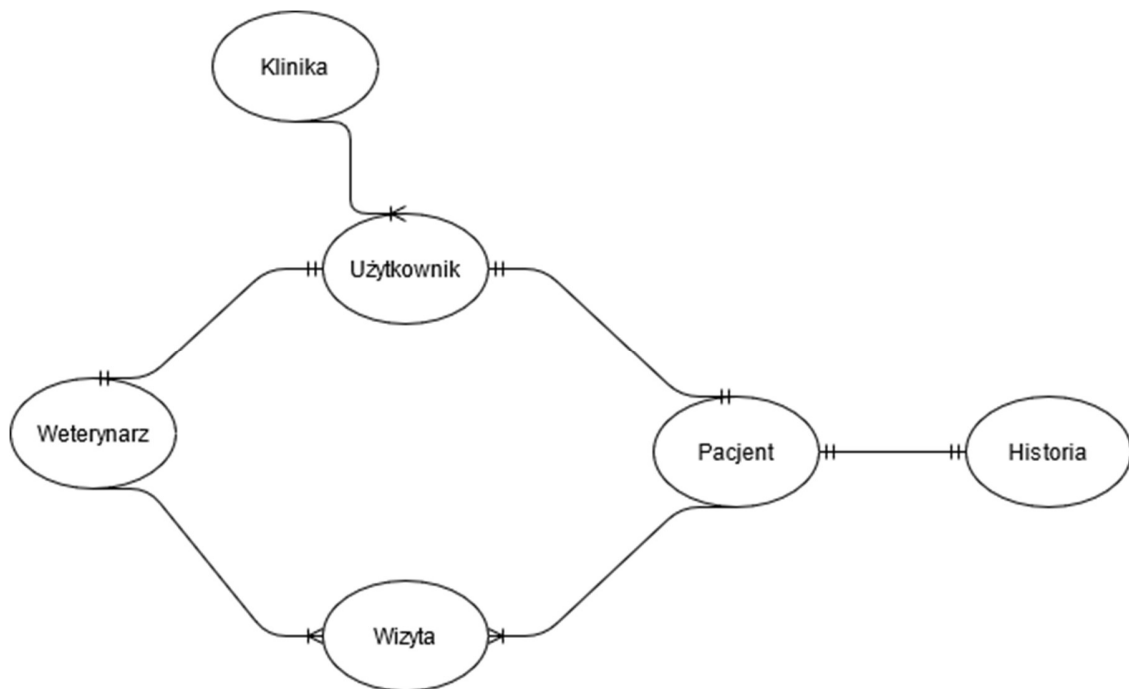


9.4 Charakterystyka zastosowanych wzorców projektowych

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

9.5 Projekt bazy danych

9.5.1 Schemat



9.5.2 Projekty szczegółowe tabel

Klinika
klínika_PK
użytkownik_FK
lokalizacja
nazwa
nr_kontaktowy

Użytkownik
użytkownik_PK
weterynarz_FK
pacjent_FK
email
nr_telefonu
hasło
stan

Użytkownik
użytkownik_PK
weterynarz_FK
pacjent_FK
email
nr_telefonu
hasło
stan

Weterynarz
weterynarz_PK
wizyta_FK
imie
nazwisko
specjalizacja

Pacjent
pacjent_PK
historia_FK
wizyta_FK
imie
nazwisko
Gatunek
imie_zwierzęcia

Historia
historia_PK
długość_wizyty
rodzaj_wizyty
data
stan

Wizyta
wizyta_PK
rodzaj_wizyty
nazwa_zabiegu
data
kwota
stan

9.6 Projekt interfejsu użytkownika

The screenshot displays the ZOOT application interface. At the top, a light blue header bar contains the 'ZOOT' logo on the left and the user's name 'Jakub Wajs' with a profile icon on the right. A vertical sidebar on the left lists navigation options: 'Historia', 'Umów wizytę' (highlighted in a darker blue), 'Wizyty', 'Kalendarz', and 'Kontakt'. The main content area, titled 'Wizyta', contains a form with four dropdown menus: 'Lokalizacja' (set to 'Warszawa'), 'Termin' (set to '01.06.2021'), 'Specjalizacja' (set to 'Chirurg'), and 'Weterynarz' (set to 'Jakub Wajs').

9.6.1 Lista głównych elementów interfejsu

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

9.6.2 Przejścia między głównymi elementami

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

9.6.3 Projekty szczegółowe poszczególnych elementów

1. Logowanie

A login form for the ZOOT system. It features a light blue background. At the top center is the ZOOT logo, with 'Z' in black, 'O' in green, and 'OT' in blue. Below the logo are two white input fields with black borders. The first field is labeled 'Adres e-mail' and the second is labeled 'Hasło'. Below these fields is a light blue button with rounded corners and a black border, labeled 'Zaloguj'.


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

2. Historia wizyt

ZOOT		Jakub Wajs		
Historia	Historia			
Umów wizytę				
Wizyty	01.04.2021 - Dermatologia Lekarz: Michał Weterynarski Zabieg: badania skóry			
Kalendarz				
Kontakt	25.06.2021 - Chirurgia Lekarz: Adam Kot Zabieg: kastracja			
	30.06.2021 - Laryngologia Lekarz: Beata Zutowska Zabieg: Badanie uszu			

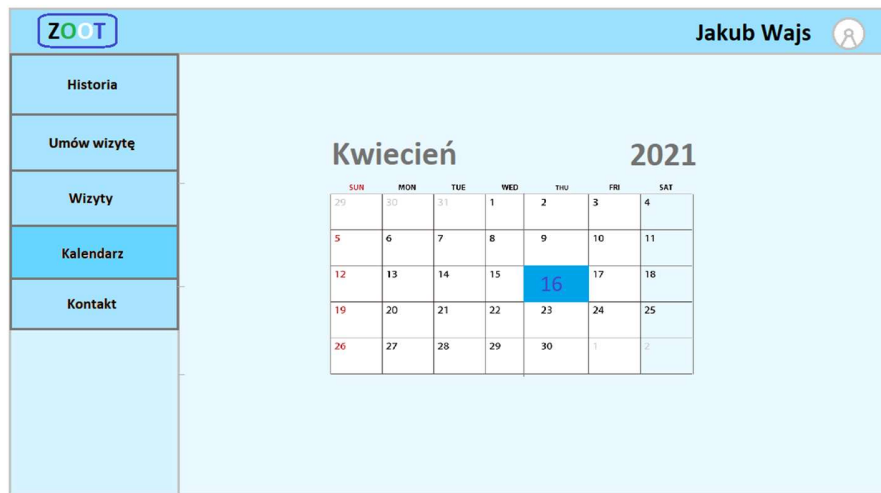
- po kliknięciu w zakładkę historia w centralnej części pojawia się historia odbytych wizyt

4. Wizyty

ZOOT		Jakub Wajs	
Historia	<div>Wizyty</div> <div><div>03.04.2021 - Dermatologia (3 dni)</div><div>Lekarz: Michał Wererynarski</div></div> <div><div>06.04.2021 - Chirurgia (6 dni)</div><div>Lekarz: Adam Kot</div></div>		
Umów wizytę			
Wizyty			
Kalendarz			
Kontakt			

- po kliknięciu w wizyty wyświetla się lista nadchodzących wizyt

5. Kalendarz



- Po kliknięciu w zakładkę kalendarz wyświetla się kalendarz wizyt

6. Kontakt

The screenshot displays the ZOOT application interface. At the top, there is a blue header bar containing the 'ZOOT' logo on the left and the user name 'Jakub Wajs' with a profile icon on the right. A vertical sidebar on the left contains five menu items: 'Historia', 'Umów wizytę', 'Wizyty', 'Kalendarz', and 'Kontakt'. The 'Kontakt' item is highlighted in a darker blue. The main content area has a light blue background and is titled 'Kontakt'. It features a form with a dropdown menu labeled '-Lekarz-' with a downward arrow, a large text input field labeled '-Text-', and a 'Wyślij' (Send) button at the bottom right.

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

9.7 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

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

Mateusz Małecki – 30h

Jakub Wajs – 30h

Mateusz Małecki punkty: 2, 4, 6, 8

Jakub Wajs punkty: 1, 3, 5, 7, 9

12 Inne informacje

przydatne informacje, które nie zostały ujęte we wcześniejszych punktach