Analiza specyfikacji wymagań

Psi Hotel

Spis treści

[1. Wstęp 3](#_Toc17439)

[2. Cele analizy specyfikacji 3](#_Toc15688)

[3. Zakres specyfikacji 3](#_Toc29687)

[4. Analiza wymagań funkcjonalnych - model use case'ów 3](#_Toc27866)

[4.1 Specyfikacja aktorów 3](#_Toc24843)

[4.2 Lista use case'ów 3](#_Toc1753)

[4.3 Diagram use case'ów UML 4](#_Toc29056)

[4.4 Specyfikacja use case’ów 4](#_Toc13252)

[4.5 Diagramy aktywności UML dla use case’ów 4](#_Toc25345)

[5. Analiza wymagań niefunkcjonalnych 4](#_Toc15644)

[5.1 Interfejsy użytkownika 4](#_Toc22916)

[5.2 Interfejsy programowe 4](#_Toc7574)

[6. Analiza wymagań dotyczących jakości modelowanego systemu 4](#_Toc31027)

[7. Analiza warunków serwisowania 5](#_Toc21455)

[8. Analiza ograniczeń architektury systemu 5](#_Toc3283)

[9. Model bazy danych 5](#_Toc5039)

[9.1 Specyfikacja atrybutów bazy danych 5](#_Toc1046)

[9.2 Specyfikacja operacji na danych 7](#_Toc5088)

[9.3 Specyfikacja reguł poprawności i zgodności typów danych 7](#_Toc12726)

# 1. Wstęp

W podanym dokumencie przedstawiono następujące elementy: analizę wymagań funkcjonalnych, analizę wymagań niefunkcjonalnych, analizę wymagań dotyczących jakości modelowanego systemu, analizę warunków serwisowania oraz analizę ograniczeń architektury systemu. Dodatkowo w ostatnim rozdziale przedstawiono model bazy danych, który został podzielony na specyfikację atrybutów bazy danych, specyfikację operacji na bazie danych oraz s[pecyfikację reguł poprawności i zgodności typów danych](#_heading=h.1y810tw). W rozdziale dotyczącym analizie wymagań funkcjonalnych wyszczególniono, specyfikację aktorów, listę use case’ów, diagram use case UML oraz diagram aktywności UML. Następnie rozdział dotyczący analizy metod niefunkcjonalnych został podzielony na poszczególne interfejsy: użytkownika, sprzętowy, komunikacyjny oraz programowy.

# 2. Cele analizy specyfikacji

Celami analizy specyfikacji wymagań jest stworzenie use case’ów, określenie wymagań niefunkcjonalnych: interfejsów użytkownika, sprzętowych, komunikacyjnych i programowych. Przeanalizowanie wymagań jakości, warunków serwisowania i ograniczeń architektury.

# 3. Zakres specyfikacji

Oprogramowanie obsługuje system bazo-danowy w celu ułatwienia prowadzenia dokumentacji oraz obsługi hotelu. Oprogramowanie działa poprawnie na systemach Windows 10/11

Następujące funkcje oprogramowania:

* Logowanie do oprogramowania,
* Dodawanie, edycja oraz usuwanie psów,
* Dodawanie, edycja oraz usuwanie właścicieli psów,
* Dodawania, edycja oraz usuwanie wizyt w hotelu.

Użytkownikiem systemu będzie osoba zatrudniona przez hotel dla psów, również właściciel danego przedsiębiorstwa będzie użytkownikiem tego oprogramowania. Użytkownicy mają prawo do edycji, usuwania oraz dodawania krotek do bazy danych.

Tabele takie jak wizyty, właściciele oraz aktywności są połączone w sposób naturalny z tabelą zawierającą informacje o psach.

Wymagania niefunkcjonalne oprogramowania:

* Zapewnienie bezpieczeństwa danych w celu zachowania poufności oraz przepisów RODO,
* Dokładność informacji, które są niezbędne do prawidłowego oraz bezpiecznego dla wszystkich funkcjonowania hotelu,
* Szybki czas reakcji na zapytania użytkownika
* Łatwość korzystania z aplikacji
* Mała liczba błędów lub ich brak w trakcie korzystania z aplikacji

# 4. Analiza wymagań funkcjonalnych - model use case'ów

## 4.1 Specyfikacja aktorów

We wszystkich use casach występuje jeden aktor - Pracownik;

## 4.2 Lista use case'ów

* Logowanie
* PlanowanieWizyty
* PrzyjazdPieska
* ProwadzenieDziennikaAktywności
* WyjazdPieska

## 4.3 Diagram use case'ów UML

* DiagramUCLogowanie.jpg
* DiagramUCPlanowanieWizyty.jpg
* DiagramUCPrzyjazdPieska.jpg
* DiagramUCProwadzenieDziennikaAktywności.jpg
* DiagramUCWyjazdPieska.jpg

## 4.4 Specyfikacja use case’ów

* Logowanie.docx
* PlanowanieWizyty.docx
* PrzyjazdPieska.docx
* ProwadzenieDziennikaAktywności.docx
* WyjazdPieska.docx

## 4.5 Diagramy aktywności UML dla use case’ów

* DiagramAktywnościLogowanie.jpg
* DiagramAktywnościPlanowanieWizyty.jpg
* DiagramAktywnościPrzyjazdPieska.jpg
* DiagramAktywnosciDziennik.jpg
* DiagramAktywnościWyjazdPieska.jpg

# 5. Analiza wymagań niefunkcjonalnych

## 5.1 Interfejsy użytkownika

Komunikacja urządzenie - użytkownik odbywa się przy pomocy urządzenia wskazującego, jest to interfejs graficzny, który ma na celu umożliwienie użytkownikowi łatwego oraz szybkiego odczytywania danych.

Wymienić formatki - powklejać okna

## 5.2 Interfejsy programowe

interfejsy programistyczne - win wpf

# 6. Analiza wymagań dotyczących jakości modelowanego systemu

Głównym wymaganiem klienta było wymaganie dotyczące użytkowania systemu, to znaczy łatwość wdrożenia systemu w hotelu dla psów oraz łatwość późniejszej obsługi oraz wykonywania zadań na danym oprogramowaniu.

Dodatkowym wymaganiem było wymaganie odporności na awarie danego oprogramowania oraz odporrność systemu informatycznego na błędy oraz awarie.

Na samym końcu wzięto pod uwagę efektywność systemu w odniesieniu do zachowywania w czasie, pobierania danych z bazy danych oraz wykorzystania zasobów komputera.

# 7. Analiza warunków serwisowania

Warunki serwisowania zawarte z hotelem są zawarte na podstawie umowy serwisowej oprogramowania. Oprogramowanie będzie regularnie serwisowane w celu eliminacji wszelkich występujących w trakcie użytkowania oprogramowania błędów. W umowie serwisowej zawarto również możliwość dalszego rozwoju oprogrogramowania w zależności od zapotrzebowania klienta.

# 8. Analiza ograniczeń architektury systemu

Do ograniczeń systemu należą:

* brak metod analitycznych, które jednoznacznie rozstrzygają, czy system spełnia oczekiwania
* problemy komunikacyjne między interesariuszami
* brak dedykowanego architekta oprogramowania.

# 9. Model bazy danych

## 9.1 Specyfikacja atrybutów bazy danych

W bazie danych możemy wyróżnić następujące tabele, które zawierają następujące atrybuty:

dbo.dailyActive

idActivity, Primary Key, Integer, not null

idDog, Integer, null

dateActivity, Date, null

hourActivity, time(7), null

activityDescription, varchar(max), null

dbo.dogOwner

idOwner, Primary Key, Integer, not null

login, varchar(255), not null

password, varchar(255), not null

name, varchar(255), null

surname, varchar(255), null

address, varchar(255), null

phoneNumber, varchar(255), null

email, varchar(255), null

idNumber, varchar(255), null

dbo.dogs

idDog, Primary Key, Integer, not null

Name, varchar(255), null

sterilization, Integer, null

lastEstrus, Date, null

breed, varcahar(255), null

color, varchar(255), null

age, float, null

weight, float, null

food, varchar(255), null

feedingFrequency, Integer, null

feedingHour, varchar(255), null

feedingNotes, varchar(255), null

hoursOfWalks, varchar(255), null

lengthOfWalks, Integer, null

healthStatus, varchar(255), null

veterinarianIndication, varchar(255), null

Vaccination, Date, null

ticksProtection, Date, null

vetInfo, varchar(255), null

characterDescription, varchar(255), null

catsReaction, varchar(255), null

favToy, varchar(255), null

knownCommands, varchar(255), null

beautyTreatments, varchar(255), null

hotelStays, varchar(255), null

dbo.visists

idVisits, Primary Key, Integer, not null

dogsName, varchar(255), null

beginDate, Date, null

endDate, Date, null

idDog, Foreign Key, Integer, null

## 9.2 Specyfikacja operacji na danych

Przy pomocy oprogramowania możemy przeprowadzić następujące operacje na bazach danych:

addDailyInfoDate - dodajemy do bazy danych informacje na temat dziennej aktywności psów w hotelu

addDateDog - dodajemy informacje na temat psów przebywających w hotelu

addVisit - dodajemy wizytę danego psa w hotelu

deleteDailyInfoDate - usuwamy informację na temat dziennej aktywności

deleteDog - usuwamy dane na temat psa

deleteVisit - usuwamy informacje na temat wizyty

editDailyInfoDate - edycja informacji na temat dziennej aktywności

editDateDog - edycja inforamcji na temat psa

editVisit - edycja informacji na temat wizyty psa

getDailyInfoDate - pobieranie danych z tabeli DayliInfoDate

getDogInfo - pobiera danych z tabeli na temat dodanych psów

getDogName - pobiera informacje na tema imienia danego psa

getVisists - pobiera informacje na temat wizyt w hotelu danego psa

## 9.3 Specyfikacja reguł poprawności i zgodności typów danych

Nie dotyczy