

Państwowa Wyższa Szkoła Zawodowa w Legnicy
Wydział Nauk Technicznych i Ekonomicznych
Kierunek Informatyka, Rok II, n2PAM1, studia niestacjonarne

PROJEKT NA ZALICZENIA WYKŁADU I
LABORATORIUM Z PRZEDMIOTU
„PROJEKTOWANIE SYSTEMÓW BAZ DANYCH”
WIZYTY

Autor:
JAKUB HAWRO, PAWEŁ ŻELAZEK

Czerwiec 2020

Github:
<https://github.com/Lioheart/Weterynarz>

Spis treści

1	Koncepcja	4
	Cel bazy danych	4
	Opis dziedziny przedmiotowej	4
	Założenia wstępne	4
	Technologie wykorzystane do tworzenia aplikacji	5
2	Specyfikacja wymagań systemu	6
	Użytkownicy systemu	6
	Przypadki użycia systemu	7
	Model konceptualny danych	10
	Użyteczność	10
	Utrzymywalność	11
3	Model logiczny systemu	12
	Model danych – diagram klas UML	12
	Model funkcjonalny	18
4	Interfejs użytkownika	19
	Projekt graficzny oraz opis działania aplikacji	19
	Zapytania SQL stosowane w aplikacji	27

Spis tabel

2.1	Zmiana hasła	7
2.2	Zalogowanie	7
2.3	Przegląd umiejętności pracowników	8
2.4	Przegląd klientów	8
2.5	Przegląd usług	8
2.6	Przegląd pracowników	9
2.7	Przegląd rezerwacji wizyt	9

Spis rysunków

2.1	Use Case Diagram	7
2.2	Diagram ERD	10
3.1	Main UML	12
3.2	Klienci UML	13
3.3	Pracownicy UML	14
3.4	Pracownicy-Uslugi UML	14
3.5	Program (logowanie) UML	15
3.6	Zmiana hasła UML	16
3.7	Rezerwacje UML	17
3.8	Uslugi UML	17
4.1	Logowanie	19
4.2	Zmiana hasła	20
4.3	Pracownicy	21
4.4	Uzupełnienie pól w zakładce Pracownicy	22
4.5	Uslugi	23
4.6	Pracownik – Uslugi	24
4.7	Klienci	25
4.8	Rezerwacje	26

Rozdział 1

Koncepcja

Cel bazy danych

Celem bazy danych jest obsługa wizyt w gabinetach, np. dentystycznym czy u weterynarza. Inaczej, jest to system rezerwacji wizyt w różnych branżach. My postanowiliśmy użyć tego do obsługi gabinetu weterynaryjnego.

Opis dziedziny przedmiotowej

Aplikacja pozwoli na:

- Dodawanie, modyfikowanie oraz usuwanie pracowników,
- Ustalanie, czy dany pracownik obsługuje klientów, czy nie,
- Dodanie godzin pracy tym pracownikom, którzy obsługują klientów,
- Dodanie, modyfikacja oraz usuwanie usług, wykonywanych przez pracowników,
- Ustalanie które usługi są wykonywane przez których pracowników,
- Dodawanie, modyfikacja oraz usuwanie danych klientów,
- Ustawianie wizyt na określony termin, zmiana statusu na wykonane lub rezygnację z danego terminu,
- Wyświetlanie listy wizyt dla danego pracownika,
- Możliwość zmiany hasła.

Założenia wstępne

Program ten ma za zadanie rezerwować usługi weterynaryjne dla poszczególnych klientów. Aplikacja umożliwia dodawanie pracowników i klientów, edytowanie ich danych, ustawianie im odpowiednich godzin pracy (dla pracowników) oraz ich usuwanie (wszystkie typowe operacje na bazach danych). Po-
przez zakładkę usługi mamy możliwość stworzenia odpowiednich usług wykonywanych w przychodni

weterynaryjnej, które można połączyć z wybranymi pracownikami. Program umożliwia pracownikom dodawać klientów i rezerwować im usługi wybierając odpowiedni dzień oraz godzinę. Użytkownik ma także możliwość zmiany hasła.

Technologie wykorzystane do tworzenia aplikacji

- Instrukcje, tablice, pętle, funkcje, moduły, wyjątki, obsługa plików, obsługa Python 3.7.5 +, obsługa MySQL,

Zewnętrzne biblioteki:

- Biblioteka mysql-connector-python,
- Biblioteki PyQt5 (QPushButton, QVBoxLayout, QHBoxLayout, QSqlTable, QLabel, QWidget, QLineEdit, QSortFilterProxyModel, QCalendarWidget, QDateTimeEdit, QVBoxLayout, QCheckBox, QGroupBox, QHBoxLayout, QSqlTableModel, QSqlRelationalTableModel, QTextEdit, QVBoxLayout, QDialogButtonBox, QIcon, QSqlDatabase, QApplication, QMainWindow), sys, datetime.

Rozdział 2

Specyfikacja wymagań systemu

Użytkownicy systemu

Użytkownikami systemu jest administrator (konto, którego nie można usunąć) oraz pracownicy.

Tabela 2.1: Zmiana hasłaTabela 2.2: Zalogowanie

Nazwa przypadku użycia	Przegląd umiejętności pracowników
Aktor	Pracownik
Scenariusz główny	Umożliwia wyświetlenie listy wszystkich umiejętności, listy wszystkich pracowników obsługujących klientów oraz listy umiejętności wyuczonych przez pracowników
Scenariusz poboczny 1 Lista pracowników	Wyświetla listę wszystkich pracowników
Scenariusz poboczny 2 Lista usług	Wyświetla listę wszystkich wprowadzonych do systemu usług
Scenariusz poboczny 3 Lista usług wykonywanych przez pracowników	Wyświetla usługi danego pracownika. Jest możliwość dodania nowej usługi lub usunięcia usługi danemu pracownikowi

Tabela 2.3: Przegląd umiejętności pracowników

Nazwa przypadku użycia	Przegląd klientów
Aktor	Pracownik
Scenariusz główny	Umożliwia przeglądanie danych wszystkich klientów
Scenariusz poboczny 1	Dodanie nowego klienta
Scenariusz poboczny 2	Modyfikacja danych klienta
Scenariusz poboczny 3	Usunięcie klienta

Tabela 2.4: Przegląd klientów

Nazwa przypadku użycia	Przegląd usług
Aktor	Pracownik
Scenariusz główny	Umożliwia przeglądanie listy wszystkich usług wykonywanych w gabinecie weterynarii
Scenariusz poboczny 1	Dodanie nowej usługi
Scenariusz poboczny 2	Modyfikacja usługi
Scenariusz poboczny 3	Usunięcie usługi

Tabela 2.5: Przegląd usług

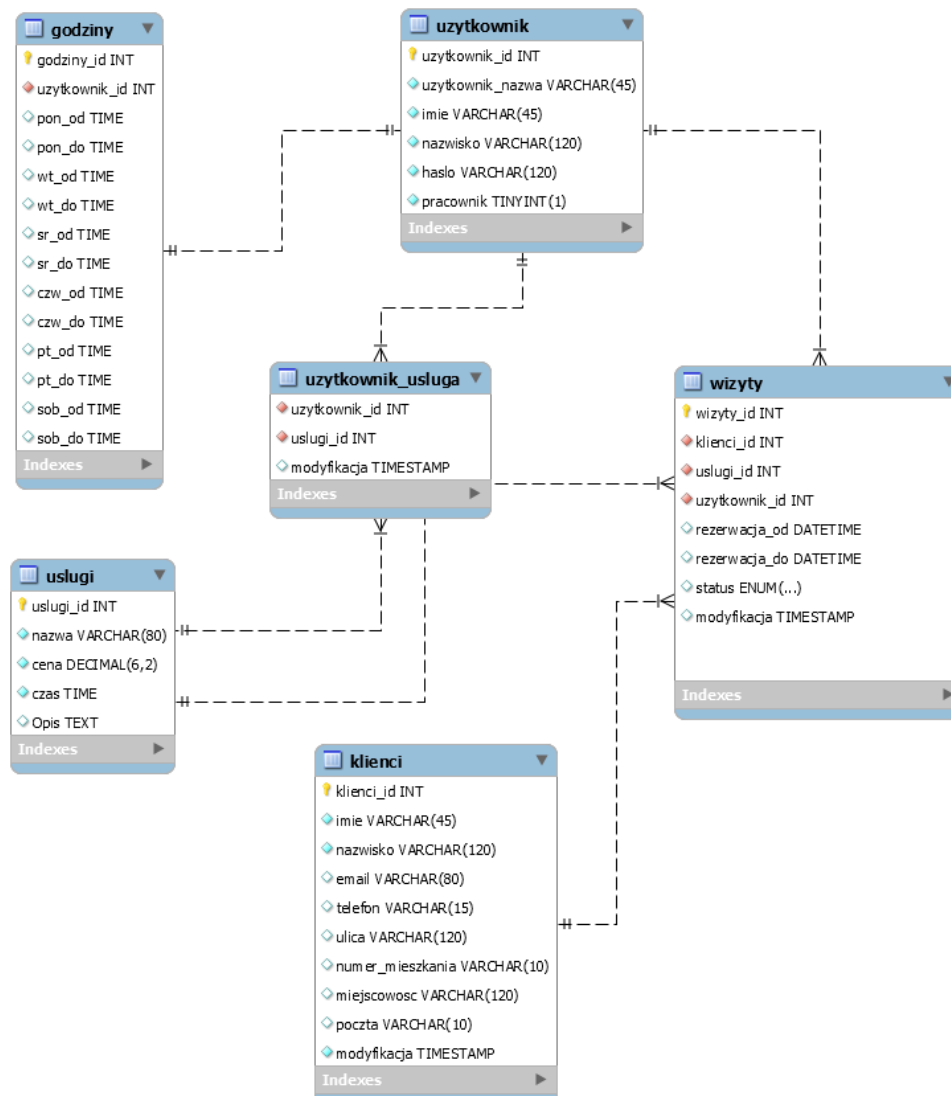
Nazwa przypadku użycia	Przegląd pracowników
Aktor	Pracownik
Scenariusz główny	Umożliwia przeglądanie listy wszystkich zatrudnionych pracowników, także tych, którzy nie wykonują usług związanych z klientami (np. księgową)
Scenariusz poboczny 1	Dodanie nowego pracownika
Scenariusz poboczny 2	Modyfikacja danych pracownika
Scenariusz poboczny 3	Usunięcie pracownika

Tabela 2.6: Przegląd pracowników

Nazwa przypadku użycia	Przegląd rezerwacji wizyt
Aktor	Pracownik
Scenariusz główny	Umożliwia przeglądanie informacji dotyczących przyszłych wizyt dla danego pracownika, dodanie nowej rezerwacji wizyty oraz modyfikacja statusu już istniejącej wizyty
Scenariusz poboczny 1 Wybór klienta	Umożliwia wybór klienta, dla którego dana wizyta ma zostać zarejestrowana
Scenariusz poboczny 2 Usunięcie rezerwacji	Usuwa już istniejącą rezerwację dla danego pracownika
Scenariusz poboczny 3 Wybór pracownika	Pozwala zobaczyć rezerwacje wizyt dla wybranego pracownika, możliwa jest także rezerwacja nowej wizyty dla wybranego pracownika
Scenariusz poboczny 4 Wybór daty i godziny pracy pracownika	Umożliwia wybór daty oraz godziny dla nowej rezerwacji wizyty
Scenariusz poboczny 5 Zmiana statusu rezerwacji	Umożliwia zmianę statusu na wykonane oraz anulowane. Domyślnie jest oczekujące
Scenariusz poboczny 6 Wybór usług oferowanych przez pracownika	Umożliwia wybranie usług, jakie oferuje dany pracownik. Wyświetla się po wybraniu pracownika

Tabela 2.7: Przegląd rezerwacji wizyt

Model konceptualny danych



Rysunek 2.2: Diagram ERD

Użyteczność:

1. Pomoc kontekstowa – w newralgicznych miejscach pomoc kontekstowa występuje, ale nie implementowaliśmy jej w każdym miejscu.
2. Dokumentacja użytkownika – nie tworzyliśmy dokumentacji, uznaliśmy, że stopień trudności obsługi jest minimalny i wystarczy raz pokazać w jaki sposób obsługiwać tą aplikację.

3. Estetyka i ergonomia interfejsu użytkownika – interfejs wizualnie jest ciekawy, nie przejawiający cech surowych. Staraliśmy się, aby opcje były widoczne, czytelne dla wszystkich. Jednakże w jednym aspekcie można było postarać się o lepszą ergonomię użytkownika (w zakładce rezerwacji wizyt).
4. Spójność interfejsu użytkownika, zgodność ze standardami – cały interfejs jest spójny wizualnie. Żadna z jego części nie odbiega od pozostałych.

Utrzymywalność:

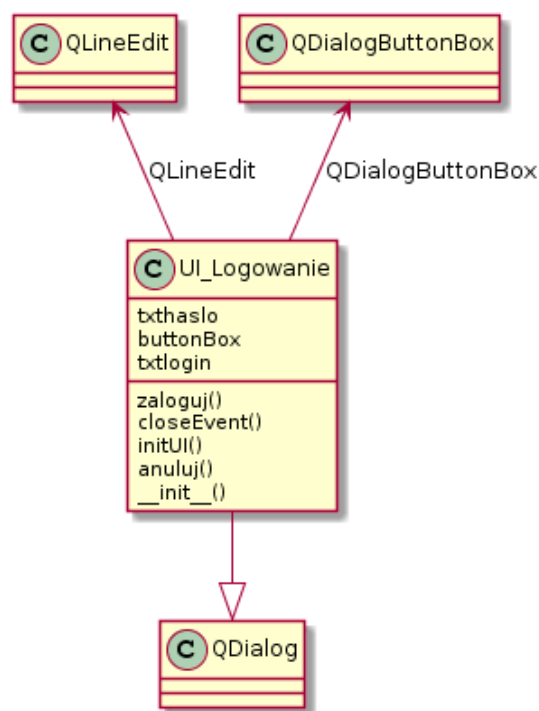
1. Rozszerzalność – aplikacja służy rezerwacji wizyt, zatem można ją zaimplementować wszędzie tam, gdzie rezerwuje się wizyty klientów – fryzjer, weterynarz, dentysta, lekarz.
2. Testowalność – nie ma możliwości przeprowadzenia testów sprawnościowych i takich też nie używaliśmy.
3. Konfigurowalność – można zmieniać tylko dane zawarte w bazie.
4. Adaptowalność – nie ma możliwości zmiany wyglądu interfejsu.
5. Administrowanie – konta utworzone w programie nie zawierają żadnych praw dostępu; są jednocześnie pracownikami. Wyszliśmy z założenia, że z programu będą korzystać jedynie pracownicy.

Rozdział 3

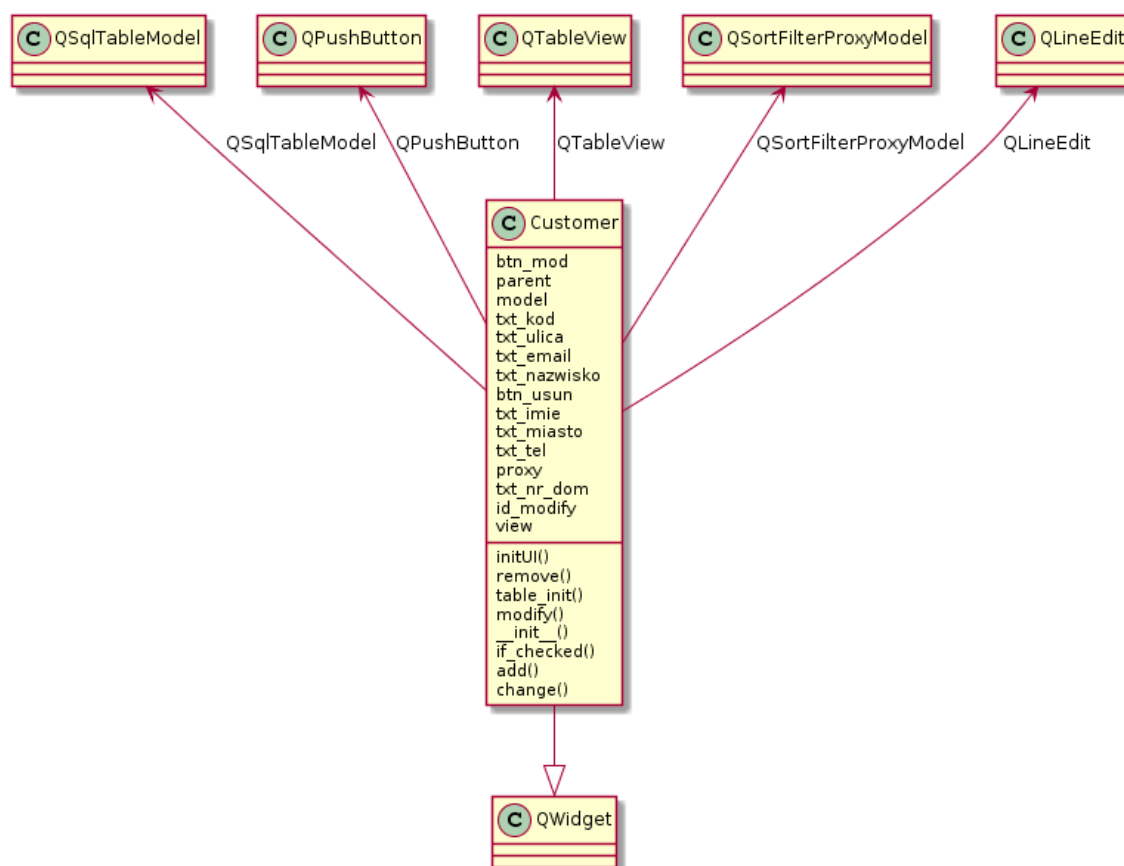
Model logiczny systemu

Model danych – diagram klas UML

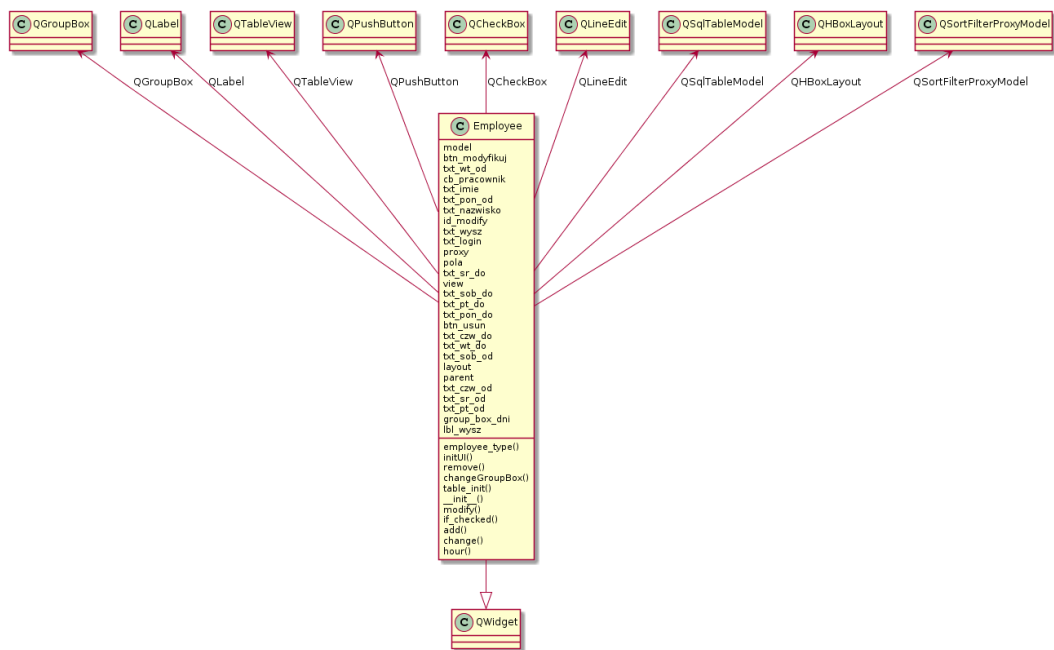
Ze względu na stopień skomplikowania tego diagramu klas, każda klasa będzie miała oddzielny diagram. W linku prowadzącym do projektu na *Githubie* będzie znajdować się także całościowy diagram klas. Ze względów funkcjonalnych, każda klasa, metoda oraz moduł są opisane w kodzie programu.



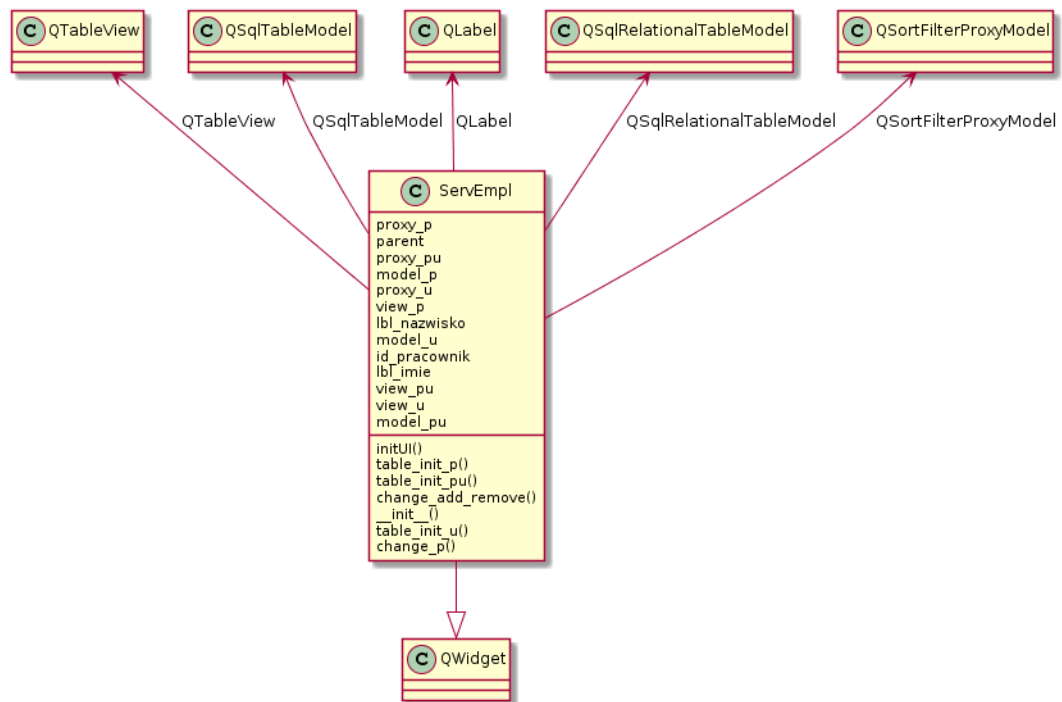
Rysunek 3.1: Main UML



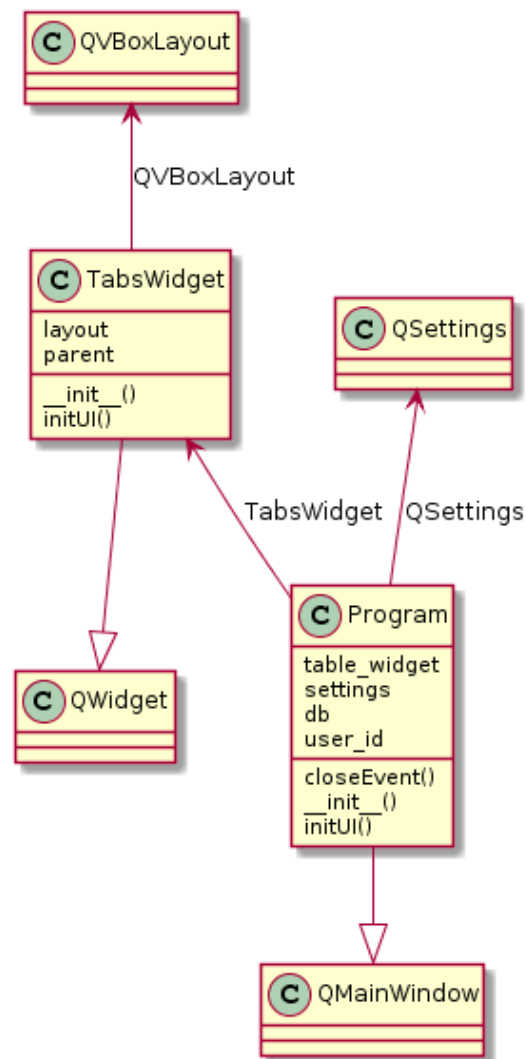
Rysunek 3.2: Klienci UML



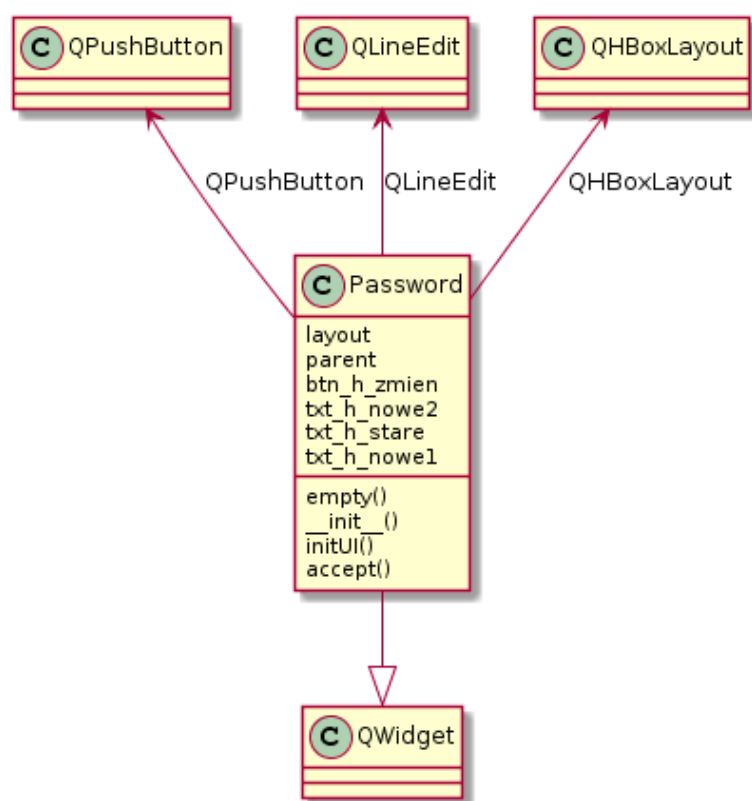
Rysunek 3.3: Pracownicy UML



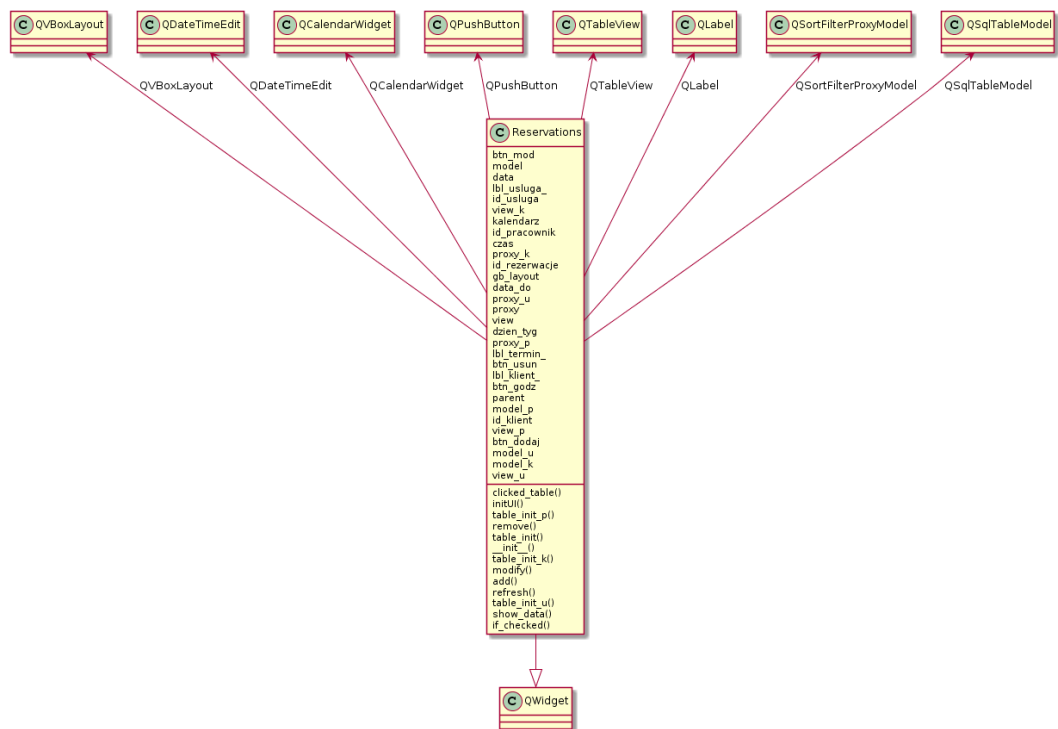
Rysunek 3.4: Pracownicy-Uslugi UML



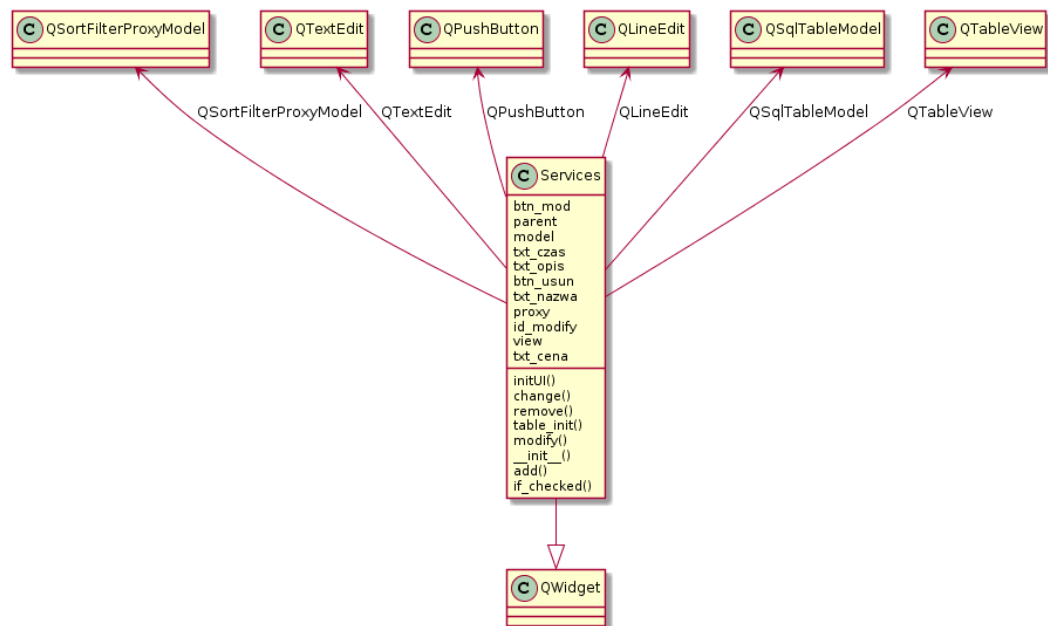
Rysunek 3.5: Program (logowanie) UML



Rysunek 3.6: Zmiana hasła UML



Rysunek 3.7: Rezerwacje UML



Rysunek 3.8: Usługi UML

Model funkcjonalny

Ze względu na powtarzalność opiszę jedynie dwa modele funkcjonalności.

Pierwszy polega na tym, że każdy przycisk w programie ma jakieś zadanie. Do każdego przycisku jest przypisana odpowiednia metoda. W większości ta metoda służy do tego, żeby nawiązać połączenie z bazą danych, wykonać polecenie query oraz odświeżyć dane wizualne za pomocą danych, zwróconych z metody.

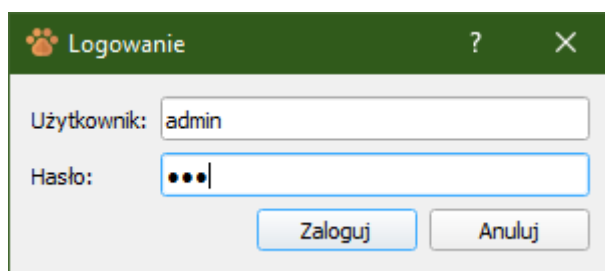
Druga polega na wyszukiwaniu informacji. Ze względów wydajnościowych wyszukiwanie odbywa się lokalnie (wczytywane są wszystkie dane i jedynie filtrowane za pomocą pola wyszukiwania). W tym przypadku połączenie z bazą danych jest zbędne.

Rozdział 4

Interfejs użytkownika

Projekt graficzny oraz opis działania aplikacji

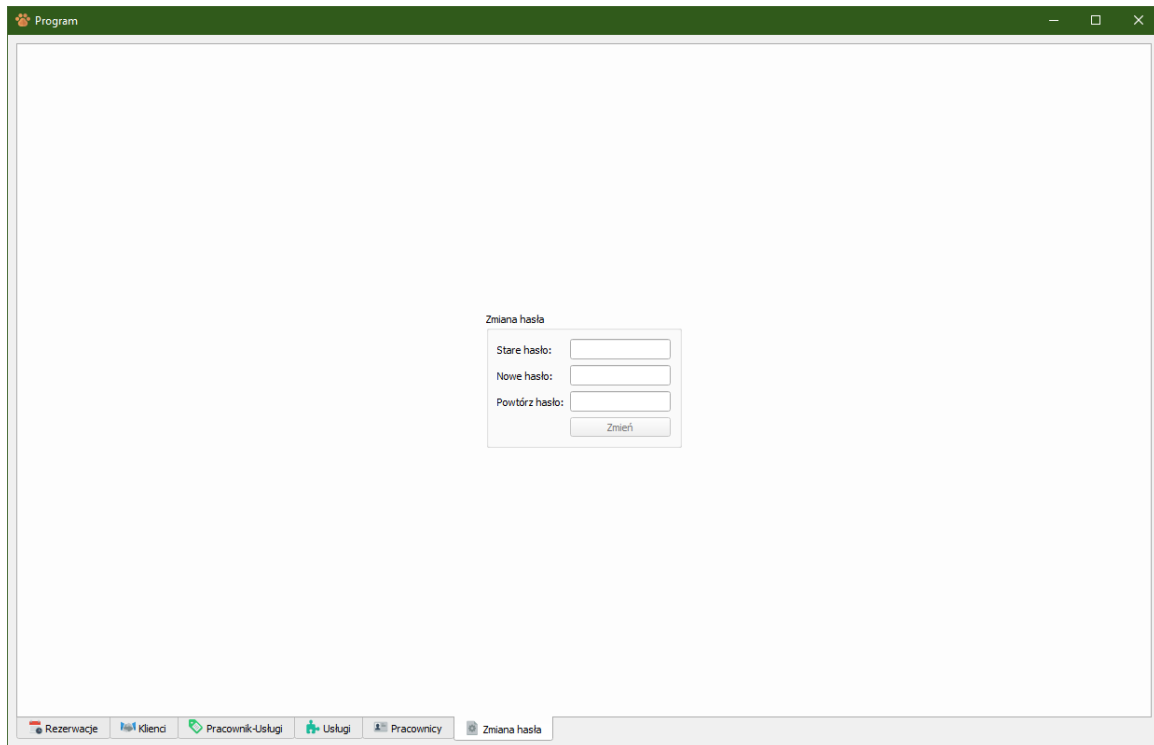
Instrukcja uruchomienia programu znajduje się w pliku README.md znajdującym się na *Githubie*. Program można uruchomić na dwa sposoby: Instalując poprzez plik setup.exe lub kompilując kod.



Rysunek 4.1: Logowanie

To pierwsza formatka, jaka pojawi się dla użytkownika. Zawiera dwa pola: Użytkownik oraz Hasło. Domyślnym użytkownikiem jest tutaj użytkownik „admin” o hasle „123”. Jednakże każdy pracownik, który zostanie później dodany może też się zalogować na swoje konto. Domyślnym hasłem tuż po utworzeniu jest tu login (czyli jeśli utworzymy użytkownika ala, hasło domyślne także będzie ala).

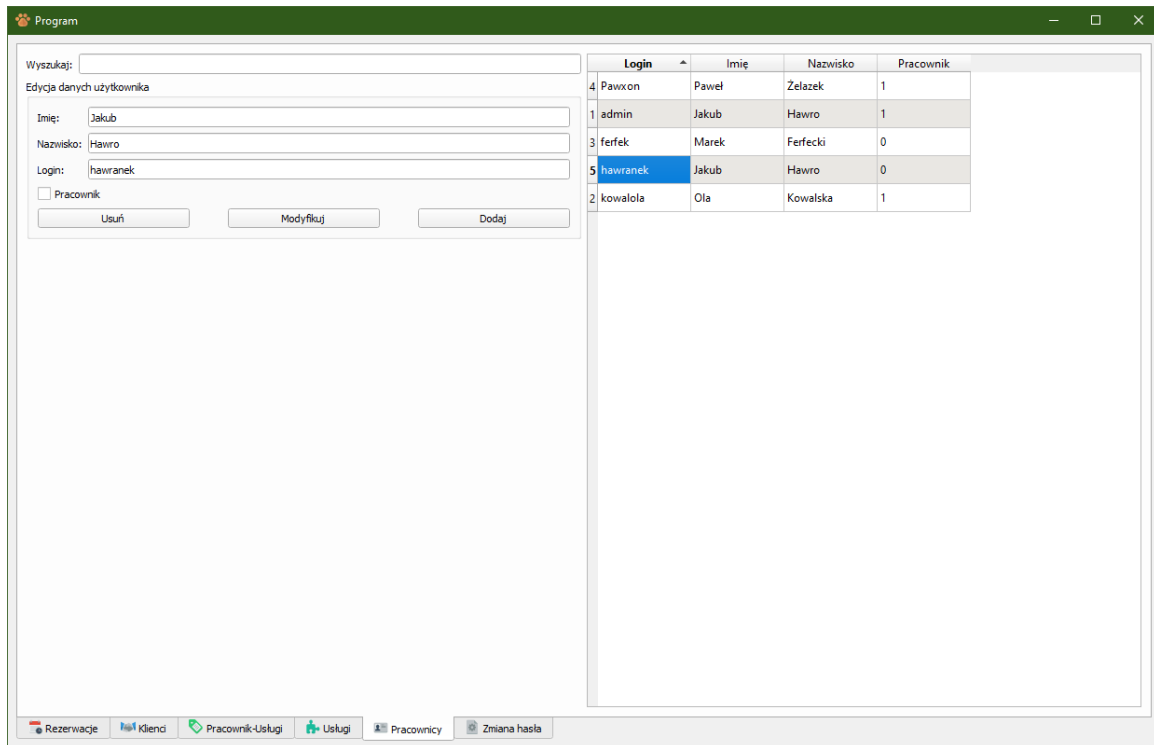
Pole Hasło jest zasłonięte. Nie ma możliwości aby podejrzeć hasło, jakie zostaje wprowadzane.



Rysunek 4.2: Zmiana hasła

Tuż po zalogowaniu pokaże nam się pierwsza zakładka Rezerwacje, jednak instrukcję użytkownika programu zacznę od ostatniej zakładki.

Zmiana hasła pozwala zmienić hasło obecnie zalogowanego użytkownika (niestety nie wyświetla się kto jest obecnie zalogowany). Aby upewnić się, czy to właściwy użytkownik chce zmienić hasło na inne, wymagane jest aby wpisał stare hasło. To takie proste zabezpieczenie, żeby czasem sprzątaczką nie zmieniła hasła sekretarce. Przycisk zmień wysyła zapytanie query do bazy danych, zmieniając jednocześnie hasło aktualnie zalogowanego użytkownika.



Rysunek 4.3: Pracownicy

Zakładka Pracownicy pozwala przejrzeć listę wszystkich pracowników danego gabinetu czy przychodni. Zakładka ta podzielona jest na dwa wizualne pola. Po prawej znajduje się tabela z wylistowanymi pracownikami. Widoczny jest login, imię oraz nazwisko oraz kolumna, czy dany pracownik jest pracownikiem (czyli czy może świadczyć usługi) czy nie (jak chociażby sekretarka, która nie będzie borowała zębów, lub szczepiła psów). Po lewej na samej górze znajduje się pole Wyszukaj. Wpisując tutaj frazę (najczęściej nazwisko pracownika), tabela po prawej będzie starała się pokazać jedynie te wiersze, które zgadzają się z frazą wpisaną w to pole. Działa to automatycznie i nie wymaga wpisywania klawisza Enter.

Poniżej znajduje się formularz dotyczący edycji danych użytkownika. Domyślnie jest on pusty, nie są aktywne przyciski Usuń oraz Modyfikuj. Aby dodać użytkownika, należy uzupełnić dane o imię, nazwisko oraz login (hasło nowego pracownika jest takie same jak login). Po uzupełnieniu tych pól i kliknięciu w przycisk Dodaj, zostanie wysłane zapytanie query do bazy, a następnie użytkownik zostanie dodany jako nowy pracownik. W momencie kliknięcia checkpointu Pracownik mamy możliwość zdefiniowania godzin pracy naszego pracownika (a także taki pracownik może świadczyć usługi klientom).

Program

Wyszukaj:

Edycja danych użytkownika

Imię:

Nazwisko:

Login:

☒ Pracownik

Godziny pracy

Poniedziałek:

Wtorek:

Środa:

Czwartek:

Piątek:

Sobota:

Szablony

Wycinek okna

	Login	Imię	Nazwisko	Pracownik
4	Pawxon	Paweł	Zelazek	1
1	admin	Jakub	Hawro	1
3	ferfek	Marek	Ferfecki	0
5	hawranek	Jakub	Hawro	0
2	kowalola	Ola	Kowalska	1

Rezerwacje Klienci Pracownik-Usługi Usługi Pracownicy Zmiana hasła

Rysunek 4.4: Uzupełnienie pól w zakładce Pracownicy

W momencie kliknięcia w szablon po prawej stronie, godziny pracy zostaną automatycznie uzupełnione. Można je także ręcznie edytować (zaleca się co najmniej zachowanie formatu HH:MM).

W przypadku chęci edytowania danych już istniejących lub usunięcia danego pracownika, należy kliknąć w wyświetlany wiersz po prawej stronie, a dane te zostaną przekazane do formularza edycji danych użytkownika. W tym momencie możemy bez problemu edytować dane, lub także usunąć te dane z bazy. Przyciski staną się aktywne. Po każdym użyciu przycisków pojawia się okienko informujące o powodzeniu (bądź nie) danej akcji.

Program

Wyszukaj zabieg:

Zabiegi

Nazwa:

Cena:

Czas:

Opis:

Dodaj

Modyfikuj

Usuń

	Nazwa	Cena	Czas	Opis
6	Obcinanie ...	10	00:15:00	
2	Odrobaczanie - ...	2,99	01:30:00	
3	Sterylizacja	199	00:30:00	Sterylizacja ...
1	Strzyżenie psa	14,49	01:00:00	Strzyżenie ...
5	Szczepienie kota	50	00:15:00	Szczepienie ...
4	Wizyta	20	00:15:00	Wizyta okresowa

Rezerwacje

Klienci

Pracownik-Usługi

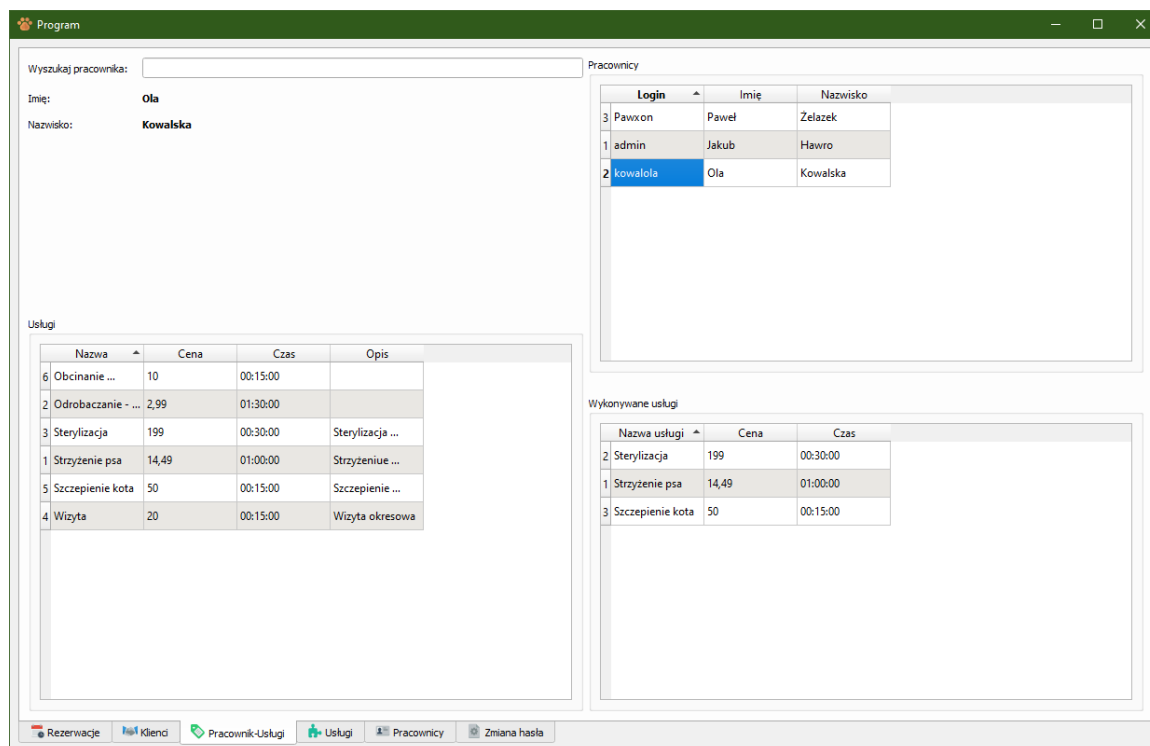
Usługi

Pracownicy

Zmiana hasła

Rysunek 4.5: Usługi

Następna zakładka reprezentuje usługi oferowane w ramach funkcjonowania gabinetu, przychodni czy zakładu. Tabela po prawej przedstawia nazwę usług, ich cenę, czas wykonania danej usługi oraz opis. U góry po lewej jest znana wyszukiwarka, której działanie poznaliśmy zakładkę wcześniej (każde pole wyszukiwania działa tak samo). Poniżej zabiegi, jakie można wykonać. Przycisk Modyfikuj i Usuń będą aktywne, jeśli wybierzemy już istniejącą usługę (wystarczy kliknąć w dany wiersz w tabeli usług). Aby dodać nową usługę, wystarczy wypełnić pola (pole Opis nie jest obowiązkowe).



Rysunek 4.6: Pracownik – Usługi

W tej zakładce widzimy aż trzy tabelę. Po prawej stronie u góry widzimy listę tylko tych pracowników, dla których przypisaliśmy godziny pracy (czyli ci, co mogą oferować swoje usługi klientom). Po prawej na dole mamy tabelę, której aktualnie nie widać. Aby ją odkryć, należy zaznaczyć danego pracownika powyżej. Ta tabela pokazuje jakie usługi może świadczyć dany (zaznaczony) pracownik. Tabela po lewej na dole reprezentuje listę wszystkich usług. U góry po lewej znajduje się pole z wyszukiwaniem pracownika oraz puste etykiety imienia i nazwiska. Zostaną one automatycznie uzupełnione w momencie, gdy wybierzemy dowolnego pracownika.

W tej zakładce możemy przypisać danemu użytkownikowi (którego zaznaczymy w tabelce Pracownicy – jego dane wyświetlą się pod polem wyszukiwania) listę usług, jakie może oferować klientom. Aby to zrobić, należy wybrać pracownika w tabeli Pracownicy, następnie dwukrotnie kliknąć na wybraną usługę. Zostanie to potwierdzone odpowiednimi komunikatami. Aby usunąć usługę z listy pracownika, należy ją kliknąć dwukrotnie w tabeli Wykonywane usługi.

Program

Wyszukaj klienta:

	Imię	Nazwisko	Email	Telefon	Ulica	Numer mieszkania	Miejscowosc	Kod pocztowy
1	Jacek	Wieczorek	jacek@przypad...	123123123	Chonowska	16/9	Wiosna	45-574
2	Paweł	Zelazek	zelazek_pawel...	579816900	Słowackiego	8	Jawor	59-400
3	Zawada	Marcin	ziewa91@wp.pl	0700880	Burdel	2	Bolesławiec	07-00

Edycja danych klienta

Imię: Ulica:

Nazwisko: Numer domu:

Email: Kod pocztowy:

Telefon: Miejscowość:

Rysunek 4.7: Klienci

W zakładce Klienci możemy natomiast dodawać, modyfikować oraz usuwać klientów naszego zakładu. Aby dodać klienta, wystarczy uzupełnić formularz Edycja danych klienta i kliknąć przycisk Dodaj. Aby zmodyfikować lub usunąć, należy kliknąć w już istniejącego użytkownika (jego dane pojawią się w polach formularza), a następnie kliknąć odpowiednie przyciski. Pole Wyszukaj klienta na samej górze szuka po wszystkich kolumnach (także możemy znaleźć wszystkich klientów z określonego miasta lub o tym samym nazwisku). Wszystkie modyfikacje będą potwierdzone za pomocą odpowiednich komunikatów.

Program

kwiecień 2020

Godziny:

Wyszukaj pracownika: ola

Imię Nazwisko

2 Ola Kowalska

Wyszukaj usługę:

Nazwa	Cena	Czas
1 Sterylizacja	199	00:30
2 Strzyżenie psa	14,49	01:00
3 Szczepienie kota	50	00:15

Wyszukaj klienta:

Imię	Nazwisko	Email	Telefon	Ulica	Numer mieszkania	Miejscowość
1 Jacek	Wieczorek	jacek@przypad...	123123123	Chonowska	16/9	Wiosna
2 Paweł	Żelazek	zelazek_pawel...	579816900	Słowackiego	8	Jawor
3 Zawada	Marcin	ziewa91@wp.pl	0700880	Burdel	2	Bolesławiec

Usługa:

Termin: 22.04.2020 00:00

Klient:

Klient	Usługa	Rezerwacja od	Rezerwacja do
1 Zawada Marcin	Szczepienie kota	28.04.2020 14:30	28.04.2020 14:30

Usunięty Zmiana statusu Dodaj

Rezerwacje Klienci Pracownik-Usługi Usługi Pracownicy Zmiana hasła

Rysunek 4.8: Rezerwacje

Ostatnia opisywana zakładka, a zarazem pierwsza, jaka jest widoczna dla użytkownika po zalogowaniu, to zakładka Rezerwacje.

W tej zakładce możemy zarezerwować wizytę u danego pracownika na określoną datę. Aby tego dokonać, musimy przede wszystkim mieć listę użytkowników, którzy mogą świadczyć usługi klientom, mieć przypisane usługi do naszych pracowników a także w bazie muszą znajdować się dane klientów.

Po prawej u góry znajduje się tabelka z pracownikami, którzy mogą wykonywać usługi. U góry każdej z tabelki znajduje się pole wyszukiwania. Aby wybrać danego pracownika, wystarczy kliknąć w odpowiedni wiersz. Po wybraniu pracownika pojawi nam się lista usług poniżej, jakie może oferować klientom (bądź nie, jeśli nie uzupełnimy tej listy). Aby wybrać, należy kliknąć na daną usługę. Na samym dole znajduje się tabelka z klientami, wraz z polem wyszukiwania. Aby wybrać, należy kliknąć w wybrany wiersz. Po lewej na dole znajduje się lista rezerwacji wizyt dla danego pracownika. Rezerwacje te możemy usuwać, zmienić ich status bądź dodać (o tym za chwilę). Powyżej znajdują się etykiety, które zostaną automatycznie uzupełnione o dane pracownika, terminu oraz klienta po wybraniu odpowiednich pól. U góry znajduje się kalendarz dla szybszego wybierania daty rezerwacji wizyty. Na samym środku znajduje się formularz z przyciskami odpowiadającym danym godzinom. Godziny te są automatycznie wybierane dla danego pracownika. Ustaliliśmy, że rezerwacje wizyt można robić co pół godziny.

Aby zarezerwować wizytę, należy podążać za instrukcją. Najpierw wybieramy pracownika, do które-

go chcemy mieć zarejestrowaną wizytę. Następnie usługę, jaką ma wykonać. Następnym krokiem jest wybranie klienta. Ostatnim krokiem jest wybranie daty (można wybrać z pola kalendarza lub wpisać w etykietę Termin) oraz wybranie godziny, na którą ma być wykonana rezerwacja. Gdy uzupełnimy te wszystkie dane, możemy kliknąć przycisk Dodaj. Rezerwacja dla danego pracownika pojawi się w tabeli po lewej stronie. Taką rezerwację można usunąć bądź zmienić jej status z Oczekującej na Wykonaną bądź Rezygnację. UWAGA! Nie pojawiają się tu rezerwacje wsteczne (na dni, które już się odbyły).

Zapytania SQL stosowane w aplikacji

```
1 query = "SELECT * FROM uzytkownik WHERE uzytkownik_nazwa = '{}' AND haslo = sha('{}')  
2 ;".format(login, haslo)
```

Za pomocą tego zapytania wysyłana jest informacja, czy dany użytkownik istnieje w bazie wraz z odpowiednim hasłem. Jeśli tak, następuje zalogowanie się do aplikacji.

```
1 query = 'UPDATE uzytkownik SET haslo = sha(%s) WHERE uzytkownik_id = %s AND haslo =  
2 sha(%s);'
```

Za pomocą tego zapytania następuje zmiana hasła danego użytkownika.

```
1 query = QSqlQuery('SELECT uzytkownik_id, uzytkownik_nazwa, imie, nazwisko, pracownik  
2 FROM uzytkownik;')
```

Za pomocą tego zapytania następuje pobranie informacji z bazy danych. Należy zauważyć, że zapytanie te, w przeciwieństwie do dwóch poprzednich, objęte jest w funkcję QSqlQuery, a to ze względu na to, że te zapytanie jest obsługiwane bezpośrednio poprzez PyQt5.

```
1 query1 = 'INSERT INTO uzytkownik (imie, nazwisko, uzytkownik_nazwa, haslo, pracownik)  
2 VALUES (%s, %s, %s, ' \  
3 'sha(%s), %s)'
```

Za pomocą tego zapytania następuje wprowadzenie nowego użytkownika do bazy danych. Należy zauważyć, że wartości VALUES nie są podane jawnie a poprzez %s, co ma zapobiec SQL Injection. Wartości te są dodawane jako osobny argument do funkcji obsługującej zapytania do bazy danych.

```

1 query2 = 'INSERT INTO godziny (uzytkownik_id, pon_od, pon_do, wt_od, wt_do, sr_od,
    sr_do, czw_od, czw_do, pt_od, pt_do, sob_od, sob_do) VALUES (LAST_INSERT_ID(),"'
    + self.txt_pon_od.text() + "','" + self.txt_pon_do.text() + "','" + self.
    txt_wt_od.text() + "','" + self.txt_wt_do.text() + "','" + self.txt_sr_od.text()
    + "','" + self.txt_sr_do.text() + "','" + self.txt_czw_od.text() + "','" + self.
    txt_czw_do.text() + "','" + self.txt_pt_od.text() + "','" + self.txt_pt_do.text()
    + "','" + self.txt_sob_od.text() + "','" + self.txt_sob_do.text() + '");'
2

```

W tym długim zapytaniu wstawiamy godziny pracy pracownika. Należy zauważyć funkcję wbudowaną w MySQL o nazwie LAST_INSERT_ID(). Dzięki niej można pobrać ostatnie ID użyte podczas zapytań do bazy danych. Z tego też powodu musiałem bezpośrednio przekazać wartości.

```

1 query1 = 'DELETE FROM uzytkownik WHERE uzytkownik_id = %s'
2 query2 = 'DELETE FROM godziny WHERE uzytkownik_id = %s'
3 query3 = 'DELETE FROM uzytkownik_usluga WHERE uzytkownik_id = %s'
4 query4 = 'DELETE FROM wizyty WHERE uzytkownik_id = %s'
5

```

Aby usunąć danego użytkownika, musimy najpierw usunąć użytkownika z tabel referencyjnych. W przeciwnym wypadku wystąpi błąd. Dlatego w tym wypadku wykonywane są aż cztery zapytania, po których następuje wykonanie (commit). Dzięki temu możemy wykonać cztery zapytania, wysyłając tylko raz informację o wykonywanych instrukcjach do serwera baz danych.

```

1 query = QSqlQuery(
2     'SELECT wizyty.wizyty_id, CONCAT(klienci.imie, " ", klienci.nazwisko) AS klient,
    uslugi.nazwa, '
3     'wizyty.rezerwacja_od, wizyty.rezerwacja_do, wizyty.status FROM klienci, uslugi
    NATURAL JOIN wizyty WHERE wizyty.rezerwacja_od > CURRENT_TIMESTAMP AND wizyty.
    uzytkownik_id = ' + str(
4         self.id_pracownik) + ';'
5

```

Za pomocą tego zapytania wyszukuję już istniejące wizyty dla danego pracownika. Jak widać, znajduje się tutaj warunek, aby pokazać jedynie te wizyty, które mają znacznik czasu większy od obecnego: wizyty.rezerwacja_od > CURRENT_TIMESTAMP