

Akademia Górniczo-Hutnicza im. Stanisława Staszica
w Krakowie wydział Fizyki i Informatyki Stosowanej

Baza Danych: Przychodnia lekarska

Joanna Marcinkowska

Kraków, 2020

Spis treści

1. Projekt koncepcji, założenia.....	3
1.1. Temat projektu	3
1.2 Analiza wymagań użytkownika	3
1.3 Funkcje	3
2. Projekt diagramów	4
2.1 Budowa i analiza diagramu przepływu danych (DFD)	4
2.2 Encje i atrybuty	4
2.3 Relacje między encjami.....	6
3. Projekt logiczny.....	7
3.1 Projektowanie tabel, kluczy i indeksów	7
3.2 Słowniki danych	7
3.3 Analiza zależności funkcyjnych i normalizacja tabel.....	Błąd! Nie zdefiniowano zakładki.
3.4 Operacje na danych	9
4. Projekt funkcjonalny.....	9
4.1 Interfejsy do prezentacji, edycji i obsługi danych	9

1. Projekt koncepcji, założenia

1.1. Temat projektu

Celem projektu jest usprawnienie działania przychodni lekarskiej w której pacjenci i lekarze mają możliwość logowania się na swoje profile. Użytkownicy mogą wyświetlać informacji o sobie i o innych użytkownikach oraz umawianie bądź usuwanie wizyt. Pacjent ma dostęp do wszystkich swoich informacji o nadchodzących i przeszłych wizytach. Administrator ma możliwość dodawania i zwalniania lekarzy, oraz modyfikowania informacji o chorobach i leczeniu.

1.2 Analiza wymagań użytkownika

W bazie są trzy rodzaje użytkowników: pacjent, lekarz i administrator.

Pacjenta posiada możliwość:

- logowania i rejestrowania się;
- przeglądania swojego profilu;
- umawiania się na wizyty.

Lekarz posiada możliwość:

- logowania się;
- przeglądania swojego profilu;
- wyszukiwania pacjentów;
- dostępu do informacji o wizytach;
- dostępu do informacji o dyżurach.

Administrator posiada możliwość:

- dodawania i usuwania użytkowników;
- dodawania i usuwania chorób i lekarstw.
- dodawania dyżurów lekarzy.

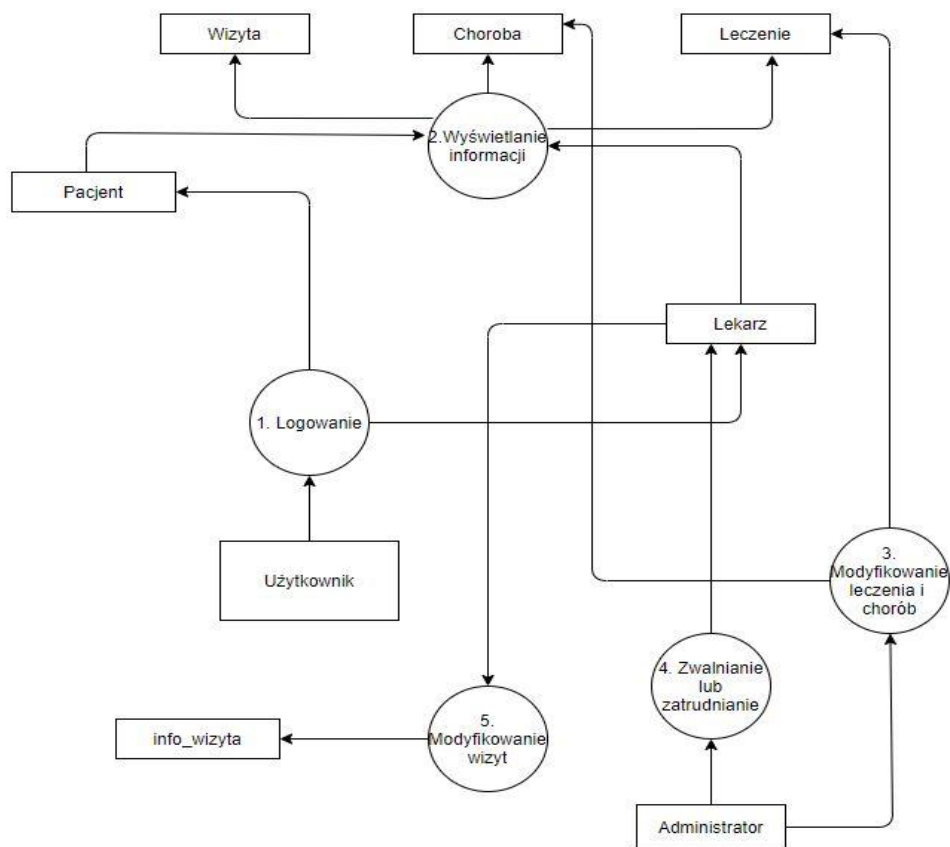
1.3 Funkcje

Podstawowe funkcje które będą realizowane to:

- logowanie i rejestrowanie użytkowników;
- umawianie i usuwanie wizyt;
- dodawanie informacji o wizytach;
- wyświetlanie informacji o użytkownikach i wizytach;
- dodawanie nowych użytkowników, chorób i lekarstw;
- wyszukiwanie pacjentów.

2. Projekt diagramów

2.1 Budowa i analiza diagramu przepływu danych (DFD)



2.2 Encje i atrybuty

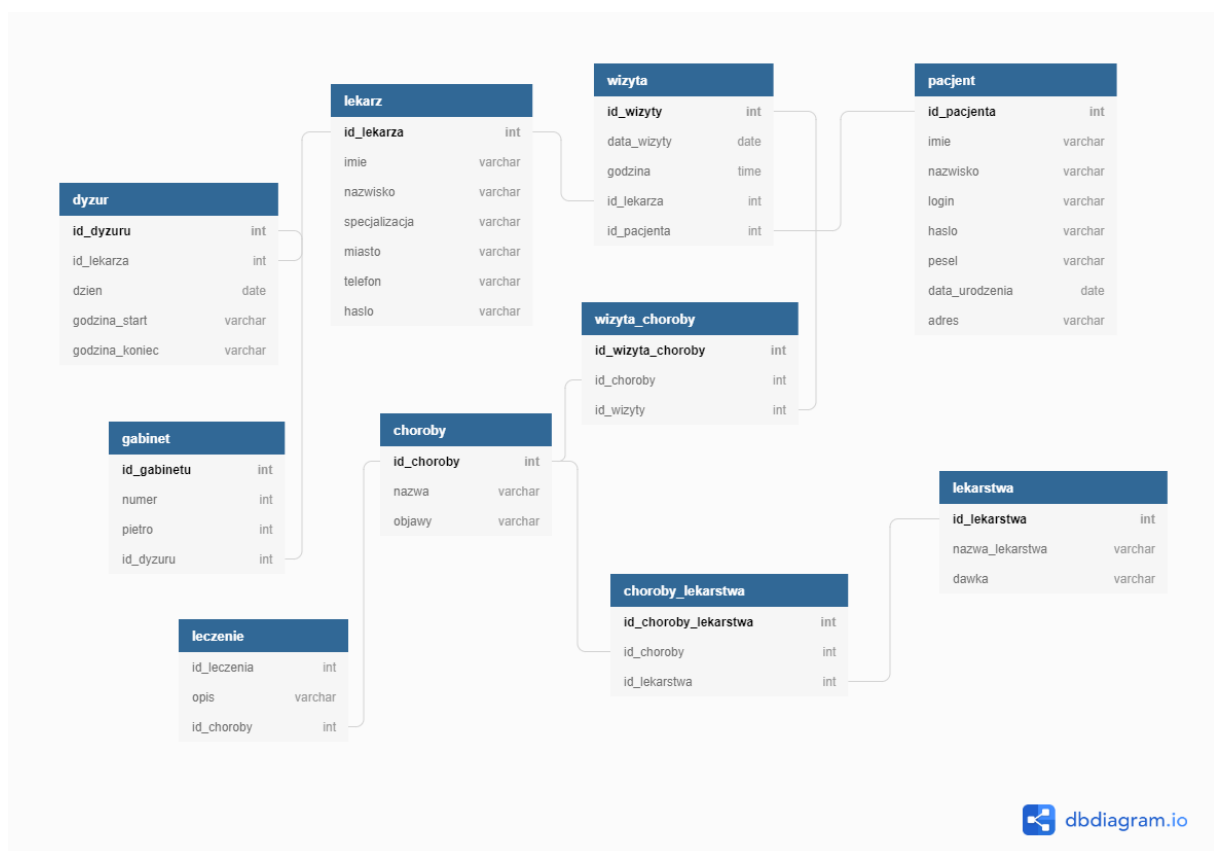
Encje pojawiające się w bazie danych to:

- pacjent – zawiera informacje o pacjencie tj.:
 - id_pacjenta – identyfikator pacjenta;
 - imię – imię pacjenta;
 - nazwisko – nazwisko pacjenta;
 - PESEL – numer pesel pacjenta;
 - data urodzenia – datę urodzenia pacjenta;
 - adres – adres zamieszkania;
 - login;
 - hasło;
- lekarz – zawiera informacje o lekarzu tj.:
 - id_lekarza – numer wykonywania zawodu;
 - imię – imię lekarza;
 - nazwisko – nazwisko lekarza;
 - specjalizacja – specjalizację lekarza;
 - miasto – miejsce urodzenia lekarza;
 - telefon – numer telefonu lekarza;
 - hasło – hasło do konta;
- wizyta – zawiera informacje na temat wizyt pacjentów
 - id_wizyty – identyfikator wizyty;

- data_wizyty – data wizyty;
 - godzina_wizyty – godzina wizyty;
 - id_pacjenta – identyfikator pacjenta;
 - id_lekarza - identyfikator lekarza.
- lekarstwa – zawiera informacje na temat leków, które są stosowane w chorobie o podanym identyfikatorze
 - id_lekarstwa – identyfikator lekaŕstwa;
 - nazwa_lekarstwa – nazwa lekaŕstwa
 - dawka – dawka lekaŕstwa;
- choroby – zawiera informacje na temat choroby
 - id_choroby – identyfikator choroby
 - nazwa_choroby – nazwa choroby;
 - objawy – objawy choroby;
- leczenie – zawiera informacje na temat leczenia choroby o podanym identyfikatorze
 - id_leczenia - identyfikator leczenia;
 - opis - opis_leczenia;
 - id_choroby – identyfikator choroby;
- dyŕur – zawiera informacje o dyŕurze konkretnego lekarza
 - id_dyŕuru – identyfikator dyŕuru;
 - id_lekarza - identyfikator lekarza będaŕcego na dyŕurze;
 - dzieŕ – dzieŕ dyŕuru;
 - godzina_start - godzina rozpoczęcia;
 - godzina_koniec - godzina zakoŕczenia;
- gabinet – zawiera informacje o gabinecie w którym lekarz odbywa dyŕur
 - id_gabinetu – identyfikator gabinetu;
 - numer – numer gabinetu;
 - piętro – piętro na którym znajduje się gabinet;
- wizyta_choroby – zawiera informacje na temat chorób i wizyt
 - id_wizyta_choroby – identyfikator;
 - id_choroby – identyfikator choroby;
 - id_wizyty – identyfikator wizyty;
- choroby_lekarstwa – zawiera informacje o chorobach i przepisanych na nie lekaŕstwach
 - id_choroby_lekarstwa - identyfikator;
 - id_choroby – identyfikator choroby;
 - id_lekarstwa – identyfikator lekaŕstwa.

2.3 Relacje między encjami

Schemat relacyjnej bazy danych



Typy związków:

- Lekarz - dyżur

Związek 1:M. Lekarz może mieć wiele dyżurów i na jednym dyżurze może być wiele lekarzy.

- Pacjent - info_wizyta

Związek 1 : M. Pacjent może być umówiony na kilka wizyt, ale na jedną wizytę może być umówiony tylko jeden pacjent.

- Lekarz – info_wizyta

Związek binarny 1:M. Lekarz może mieć umówionych wiele wizyt, ale na jednej umówionej wizycie może być tylko jeden lekarz.

- Lekarz – wizyta

Związek 1:M. Lekarz może przyjmować na wielu wizytach, ale na jednej wizycie jest tylko i wyłącznie jeden lekarz.

- Pacjent – wizyta

Związek 1:M. Pacjent może być na wielu wizytach, ale na jednej wizycie może być tylko jeden pacjent. Pacjent może przyjść na wizytę ale nie musi, ale wizyta odbywa się tylko wtedy gdy jest obecny lekarz.

3. Projekt logiczny

3.1 Projektowanie tabel, kluczy i indeksów

Tabele zostały zaprojektowane w taki sposób aby nazwy atrybutów o takim samym znaczeniu w różnych tabelach nosiły tą samą nazwę. Każda tabela zawiera w sobie klucz główny który jest poprzedzony przedrostkiem id, a następnie nazwą tabeli z której pochodzi. Klucze obce występujące w tabelach mają taką samą nazwę jak klucze główne tabel z których pochodzą. Klucze główne są typu int i większość jest numerowana automatycznie.

3.2 Słowniki danych

Tabela pacjent

Nazwa atrybutu	Typ danych	Klucz
id_pacjenta	integer	PK
imie	varchar	
nazwisko	varchar	
login	varchar	
haslo	varchar	
pesel	varchar	
adres	varchar	
data_urodzenia	date	

Tabela lekarz

Nazwa atrybutu	Typ danych	Klucz
id_lekarza	int	PK
imie	varchar	
nazwisko	varchar	
specjalizacja	varchar	
miasto	varchar	
telefon	varchar	
haslo	varchar	

Tabela dyżur

Nazwa atrybutu	Typ danych	Klucz
id_dyżuru	int	PK
dzień	date	
godzina_start	varchar	
godzina_koniec	varchar	
id_lekarza	int	PFK

Tabela wizyta

Nazwa atrybutu	Typ danych	Klucz
id_wizyty	int	PK
data_wizyty	date	
godzina	varchar	
id_pacjenta	int	PFK
id_lekarza	int	PFK

Tabela lekarstwa

Nazwa atrybutu	Typ danych	Klucz
id_lekarstwa	int	PK
nazwa_lekarstwa	varchar	
dawka	varchar	

Tabela choroby

Nazwa atrybutu	Typ danych	Klucz
id_choroby	int	PK
nazwa	varchar	
objawy	varchar	

Tabela wizyta_choroby

Nazwa atrybutu	Typ danych	Klucz
id_wizyta_choroby	int	PK
id_choroby	int	PFK
id_lekarstwa	int	PFK

Tabela choroby_lekarstwa

Nazwa atrybutu	Typ danych	Klucz
id_choroby_lekarstwa	int	PK
id_choroby	int	PFK
id_lekarstwa	int	PFK

Tabela leczenie

Nazwa atrybutu	Typ danych	Klucz
id_leczenia	int	PK
id_choroby	int	PFK
opis	varchar	

Tabela gabinet

Nazwa atrybutu	Typ danych	Klucz
id_gabinetu	int	PK
numer	int	
piętro	int	
id_dyżuru	int	PFK

3.3 Operacje na danych

Funkcje przedstawione w punkcie 1.3 będą w większości realizowane za pomocą widoków m.in.:

- lekarz_wizyty – wyświetla informacje o pacjencie o zadanym peselu i informacje o wcześniejszych wizytach i zdiagnozowanych chorobach, oraz przepisanych lekach;
- pacjent_informacje – wyświetla informacje o pacjencie, wcześniejszych wizytach u lekarzy, przebytych chorobach, przepisanych lekach oraz przebiegu leczenia;
- pacjent_wizyty – wyświetla informacje o wizytach z przeszłości danego pacjenta;

Oraz funkcji:

- zwolnij_lekarza – zwalnia lekarza o podanym identyfikatorze;
- wyświetl_lekarzy – wyświetla dane o lekarzach o podanej specjalizacji;
- pacjent_rejestracja – wstawia informacje o pacjentach do tabeli;
- pacjent_check – sprawdzenie danych logowania pacjenta;
- lekarz_login – sprawdzanie danych logowania lekarza;
- pacjent_i – wyświetlanie informacji o przeszłych wizytach pacjentów;
- wizyta_lek – wyświetlanie informacji o przyszłych wizytach lekarzy;
- dodaj_lekarstwo – dodawanie informacji o lekarstwie;

Logowanie i rejestracja będzie sprawdzane przy pomocy funkcji:

- logowanie – sprawdza czy użytkownik o podanym loginie i hasle istnieje w bazie;
- rejestracja – rejestruje podanego użytkownika w bazie.

4. Projekt funkcjonalny

4.1 Interfejsy do prezentacji, edycji i obsługi danych

W panelu pacjenta dostępne są:

- formularz logowania,
- formularz rejestracji,
- formularz umawiania wizyty.

W panelu lekarza dostępne są:

- formularz logowania;
- formularz do wypełniania danych o wizytach.

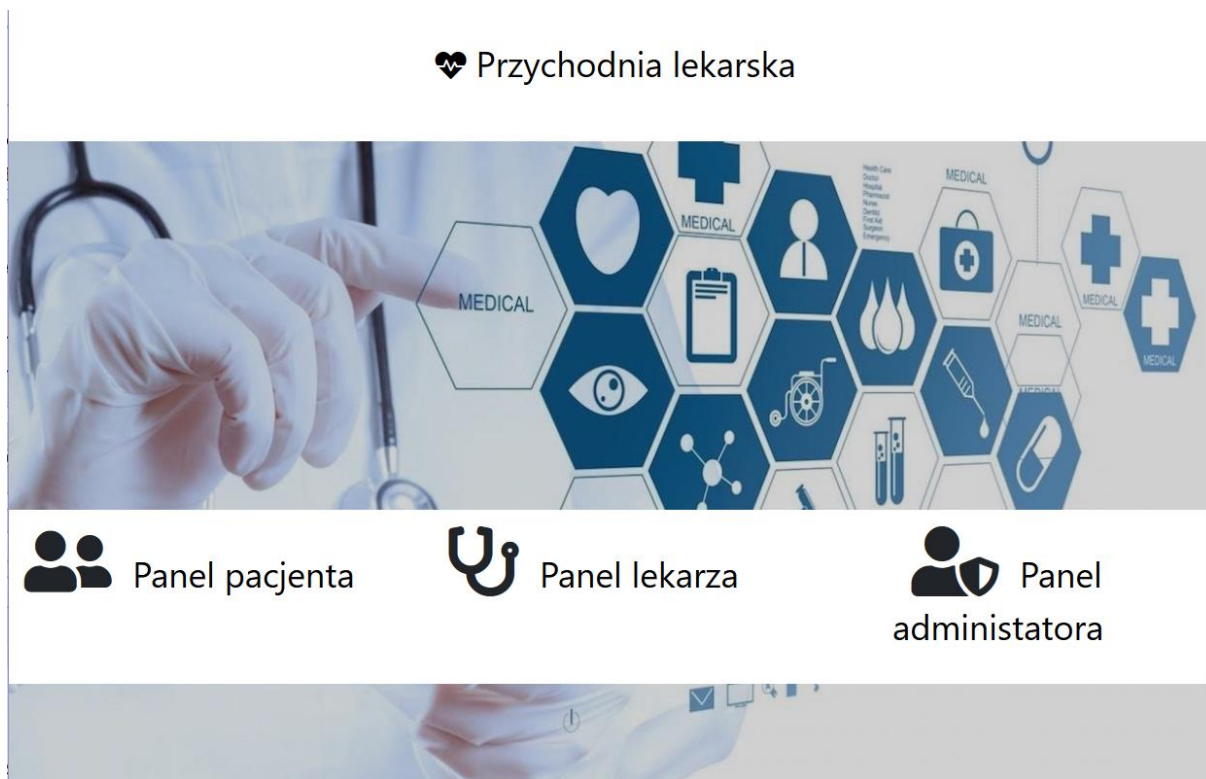
Formularz logowania i rejestracji wymaga wpisania odpowiednich danych do konkretnych pól. Formularz umawiania wizyty wymaga wpisania daty i godziny umawianej wizyty oraz wybrania lekarza, natomiast formularz do wypełniania danych o wizytach wymaga wpisania przebiegu leczenia, dawki leków, objawów choroby oraz numeru PESEL pacjenta.

4.2 Raporty

Projekt stworzony jest z wykorzystaniem języka PHP, JavaScript oraz HTML. Backend zajmuje się przekazywaniem danych do bazy. Większość danych jest reprezentowana w postaci tabel.

4.3 Panel sterowania aplikacji

Panelem startowym jest niżej pokazany panel, dzięki któremu użytkownik ma dostęp do swojego konta. Aby uzyskać dostęp do panelu pacjenta oraz panelu lekarza należy najpierw się zalogować, albo w przypadku nowych pacjentów istnieje możliwość rejestracji.



5. Dokumentacja

5.1 Wprowadzanie danych

Dane są wprowadzane ręcznie z poziomu aplikacji.

5.2 Dokumentacja użytkownika

Aplikację można znaleźć pod adresem: <http://pascal.fis.agh.edu.pl/~7marcinkowska/Bazy/index.php>

Po wejściu na stronę istnieje możliwość wyboru jednego z trzech paneli: panelu pacjenta, panelu lekarza oraz panelu administratora. W panelu pacjenta istnieje możliwość zalogowania się lub zarejestrowania, natomiast do panelu lekarza i panelu administratora można się tylko zalogować. Administratorem jest jeden ustalony lekarz i tylko on ma dostęp do panelu. W panelu pacjenta można przeglądać i edytować dane, przeglądać przyszłe i nadchodzące wizyty oraz umawiać się na wizyty. W panelu lekarza istnieje możliwość edytowania danych, przeglądania dyżurów oraz nadchodzących i przeszłych wizyt, dodawania informacji do wizyt, znajdowania informacji o pacjentach oraz dodawania nowych lekarstw. Administrator ma możliwość dodawania oraz usuwania lekarzy, dodawania nowych leków i chorób oraz przeglądania wszystkich lekarzy i dodawania dyżurów.

5.3 Wykaz literatury

- <https://www.tutorialspoint.com/postgresql>
- https://www.w3schools.com/php/php_mysql_intro.asp