

Autorzy:  
Wiktoria Salamon  
Filip Rurak

# Projekt aplikacji bazodanowej

## “Przychodnia”

### 1. Cel aplikacji

Aplikacja ma ułatwić lekarzom oraz pacjentom przechowywanie wyników badań laboratoryjnych oraz umawianie się na wizyty.

### 2. Opis funkcjonalności

Pacjenci mogą wyszukiwać wolne terminy u wybranego lekarza i zapisywać się na wizytę w wybranym terminie. Mogą również odwoływać umówione wcześniej wizyty.

Po przeprowadzeniu badań laboratoryjnych lekarz może wprowadzić wyniki badań do systemu. Wprowadzone wyniki danego pacjenta, mogą być przez niego wyświetlone. Można również sprawdzić, które wyniki nie mieszczą się w normach oraz porównywać wartości z poprzednimi wynikami.

W systemie przewidziane jest również stanowisko sekretarki przychodni, która może rejestrować pacjentów, umawiać ich na wizyty oraz pokazywać im ich wyniki badań.

### 3. Opis bazy danych

#### a. Opis tabel

- Pracownicy - tabela przechowuje informacje na temat pracowników przychodni, którzy korzystają z aplikacji.
  - staff\_id (int) - unikalne id pracownika
  - imie (varchar(45)) - imię pracownika
  - nazwisko (varchar(45)) - nazwisko pracownika
  - typ (enum("lekarz", "sekretarka")) - stanowisko zajmowane w przychodni
  - specjalizacja(varchar(45)) - dotyczy tylko lekarzy, np. endokrynolog, dermatolog
- Pacjenci - tabela przechowująca informacje na temat pacjentów zarejestrowanych w przychodni
  - p\_id (int) - unikalne id pacjenta
  - imie (varchar(45)) - imię pacjenta
  - nazwisko (varchar(45)) - nazwisko pacjenta
  - pesel (char(11)) - pesel pacjenta
  - data\_urodzenia (date) data urodzenia pacjenta
- Wyniki badan - tabela przechowująca podstawowe informacje na temat wystawionych wyników laboratoryjnych z jednego pobrania krwi
  - wb\_id (int) - id wyników
  - data\_wystawienia (datetime) - dokładny czas wprowadzenia wyników do systemu
  - id\_pacjenta (int) - id pacjenta, którego dotyczą wyniki
  - id\_lekarza (int) - id lekarza, który zlecił badanie oraz wprowadza jego wyniki do systemu
- Wyniki\_zawartosc - tabela przechowująca wyniki konkretnych badań laboratoryjnych, np. Glukozy
  - id\_wynikow (int) - id ogólnego wyniku badań

- id\_badiania (int) - id konkretnego badania
- wartosc (float) - wartość liczbowa badanej cechy
- poprzednia\_wartosc (float) - wartość wyznaczona podczas poprzedniego badania
- uwagi (varchar(100)) - uwagi lekarza na temat konkretnego wyniku
- Badania - tabela wszystkich dostępnych badań w przychodni wraz z ich opisem
  - b\_id (int) - unikalne id badania
  - nazwa (varchar(45)) - nazwa badania
  - min\_wartosc (float) - min wartość u zdrowego człowieka
  - max\_wartosc (float) - max wartość u zdrowego człowieka
  - jednostka (varchar(10)) - jednostka, w której podawany jest wynik
  - uwagi (varchar(100)) - uwagi dotyczące badania
- Wizyty - tabela zawierająca umówione wizyty
  - w\_id(int) - id wizyty
  - id\_pacjenta(int) - id pacjenta zapisanego na wizytę
  - id\_lekarza(int) - id lekarza u którego pacjent jest umówiony
  - data\_wizyty(date) - data umówionej wizyty
  - godzina\_wizyty (int) - id godziny umówionej wizyty
- Godziny - tabela przechowująca możliwe godziny wizyt
  - g\_id (int) - id godziny
  - poczatek (time) - godzina, o której wizyta się zaczyna
  - koniec (time) - godzina, o której wizyta się kończy
- Uzytkownicy - tabela użytkowników aplikacji
  - id\_uzytkownika (int) - dla pracowników: 1-999, dla pacjentów: powyżej 1000
  - login (char(10)) - login użytkownika
  - haslo (varchar(15)) - hasło użytkownika

**b. Planowane triggerzy:**

- dodaj\_pracownika AFTER INSERT ON Pracownicy - dodaje nowego użytkownika do tabeli Uzytkownicy
- usun\_pracownika AFTER DELETE ON Pracownicy - usuwa użytkownika z tabeli Uzytkownicy
- sprawdz\_poprawnosc\_peselu BEFORE INSERT ON Pacjenci - sprawdza, czy pesel ma poprawny format
- usun\_dane\_pacjenta AFTER DELETE ON Pacjenci - usuwa wszystkie wyniki badań oraz wizyty danego pacjenta
- usun\_zawartosc\_wynikow AFTER DELETE ON Wyniki\_badan - usuwa wszystkie pozycje z tabeli Wyniki\_zawartosc związane z usuwanym wynikiem
- dodaj\_poprzednia\_wartosc BEFORE INSERT ON Wyniki\_zawartosc - wpisuje poprzednią wartość wyniku danego badania, wyszukaną w bazie danych
- usun\_badanie AFTER DELETE ON Badania - usuwa wszystkie wyniki danego badania
- sprawdz\_poprawnosc\_badiania BEFORE INSERT ON Badania - sprawdza, czy max > min

**c. Planowane procedury:**

- procedury do początkowego wypełnienia tabel
- Procedury wywoływane przez aplikację, np do dodawania nowego rekordu do tabeli

**d. Planowane transakcje:**

- dodaj\_wizyte - sprawdza, czy podczas umawiania wizyty nie została już zajęta przez innego pacjenta

**e. Backup** - możliwość stworzenia backup'u bazy przez admina oraz przywrócenie poprzedniego stanu bazy

**f. Uprawnienia użytkowników**

Tabela \ Użytkownik	Użytkownik			
	Sekretarka	Pacjent	Lekarz	Admin
Wizyty	Select, Insert, Update, Delete	Select, Insert, Delete	Select	Select, Insert, Update, Delete
Pacjenci	Select, Insert, Update	-	Select, Insert, Update	Select, Insert, Update, Delete
Pracownicy	Select	Select	-	Select, Insert, Update, Delete
Wyniki_badan	Select	Select	Select, Insert, Update	Select, Insert, Update, Delete
Wyniki_zawartosc	Select	Select	Select, Insert, Update	Select, Insert, Update, Delete
Badania	Select	Select	Select	Select, Insert, Update, Delete
Uzytkownicy	-	-	-	Select, Insert, Update, Delete
Godziny	Select	Select	Select	Select, Insert, Update, Delete

#### 4. Diagram bazy danych

