Obraz zawierający tekst, Czcionka, Grafika, zrzut ekranu

Opis wygenerowany automatycznie

**Kolegium Nauk Przyrodniczych**

**Kierunek:**Informatyka

**Rok akademicki:**2023/2024 (4 semestr)

**Przedmiot:**Aplikacje internetowe 1

**Prowadzący:**dr inż. Piotr Grochowalski

**Temat projektu:**System zarządzania placówką medyczną

**Wykonanie:**Krzysztof Motas  
125145

Patryk Jarosiewicz  
123066

1. **Opis projektu**

Projekt bazy danych został stworzony z myślą o zarządzaniu informacjami medycznymi w placówce medycznej, zapewniając wsparcie w prowadzeniu dokumentacji dotyczącej personelu, lekarstw, metod leczenia oraz pacjentów.

Dzięki naszej aplikacji personel medyczny może zarządzać danymi pacjentów, śledzić historię chorób i leczenia, oraz szybko dostępować do informacji o lekarzach, lekach i gabinetach.

1. **Funkcjonalności projektu**

Baza danych umożliwia dokładne śledzenie historii wizyt każdego pacjenta u lekarza, w tym datę wizyty, powód wizyty oraz diagnozę lekarza. Ta funkcjonalność pozwala na skuteczne monitorowanie stanu zdrowia pacjentów oraz świadczenie im właściwej opieki medycznej.

Ważnym elementem projektu jest również moduł dotyczący leków i recept. Baza danych przechowuje informacje o dostępnych lekach, wraz z ich ceną. Ponadto, system umożliwia przypisywanie leków do konkretnych recept przepisanych pacjentom podczas wizyt, wraz z informacjami dotyczącymi dawkowania oraz odpłatności leku.

Niektóre z dostępnych funkcji:

* rejestrowanie pacjentów,
* przyjmowanie wizyt,
* tworzenie dokumentacji medycznej,
* przepisywanie leków oraz zarządzanie dostępnymi lekarzami/gabinetami,
* analiza danych, np. popularność przepisywanych leków, lekarze z największą liczbą wizyt, średni czas trwania wizyty, statystyki dotyczące wieku pacjentów, popularność specjalizacji lekarskich, częstotliwość wizyt w zależności od dnia tygodnia/czasu, trendy w ilości wizyt w czasie, średnia liczba wizyt w zależności od wieku pacjentów, rozkład płci pacjentów w różnych grupach wiekowych itp.,
* historia pacjentów.

Dzięki temu systemowi personel medyczny może śledzić pacjentów i zapewnić im odpowiednią opiekę medyczną.

1. **Zaprojektowane tabele oraz powiązania**

Na podstawie wyżej wymienionych założeń i opisu funkcjonalności projektu, stworzono następujące tabele:

* Tabela patients (Pacjenci):
* Identyfikator: id (integer, klucz główny)
* Imię: name (varchar)
* Nazwisko: last\_name (varchar)
* Płeć: gender (varchar)
* Adres: address (varchar)
* Numer telefonu: phone\_number (varchar)
* Tabela doctors (Lekarze):
* Identyfikator: id (integer, klucz główny)
* Imię: name (varchar)
* Nazwisko: last\_name (varchar)
* Specjalizacja: specialization (varchar)
* Numer telefonu: phone\_number (varchar)
* Tabela rooms (Sale):
* Identyfikator: id (integer, klucz główny)
* Nazwa: name (varchar)
* Skrzydło: wing (varchar)
* Piętro: floor (varchar)
* Dział: department (varchar)
* Identyfikator nadzorującego lekarza: supervising\_doctor\_id (integer, klucz obcy od doctors.id)
* Tabela visits (Wizyty):
* Identyfikator: id (integer, klucz główny)
* Identyfikator pacjenta: patient\_id (integer, klucz obcy od patients.id)
* Identyfikator lekarza: doctor\_id (integer, klucz obcy od doctors.id)
* Powód: reason (varchar)
* Data rozpoczęcia: start\_date (datetime)
* Data zakończenia: end\_date (datetime)
* Tabela documentations (Dokumentacje):
* Identyfikator: id (integer, klucz główny)
* Identyfikator wizyty: visit\_id (integer, klucz obcy od visits.id)
* Data: date (datetime)
* Diagnoza: diagnosis (varchar)
* Metoda leczenia: treatment\_method (varchar)
* Tabela prescriptions (Recepty):
* Identyfikator: id (integer, klucz główny)
* Identyfikator wizyty: visit\_id (integer, klucz obcy od visits.id)
* Data ważności: expiration\_date (datetime)
* Tabela medicines (Leki):
* Identyfikator: id (integer, klucz główny)
* Nazwa: name (varchar)
* Cena: price (integer)
* Tabela prescription\_medicines (Leki na receptach):
* Identyfikator recepty: prescription\_id (integer, klucz obcy od prescriptions.id) Identyfikator leku: medicine\_id (integer, klucz obcy od medicines.id) Dawkowanie: dosage (varchar)
* Opłata: payment (integer)

Pomiędzy tabelami zostały stworzone następujące relacje:

* doctors.id (Tabela doctors) - rooms.supervising\_doctor\_id (Tabela rooms)

Jest to relacja jeden-do-wielu (>). Jeden lekarz może nadzorować wiele pomieszczeń.

* patients.id (Tabela patients) < visits.patient\_id (Tabela visits)

Jest to relacja jeden-do-wielu (<). Jeden pacjent może mieć wiele wizyt.

* doctors.id (Tabela doctors) < visits.doctor\_id (Tabela visits)

Jest to relacja jeden-do-wielu (<). Jeden lekarz może przeprowadzić wiele wizyt.

* visits.id (Tabela visits) - documentations.visit\_id (Tabela documentations)

Jest to relacja jeden-do-jednego (-). Jedna wizyta może mieć tylko jedną dokumentację medyczną.

* prescriptions.visit\_id (Tabela prescriptions) > visits.id (Tabela visits)

Jest to relacja wiele-do-jednego (>). Wiele recept może być przypisanych do jednej wizyty.

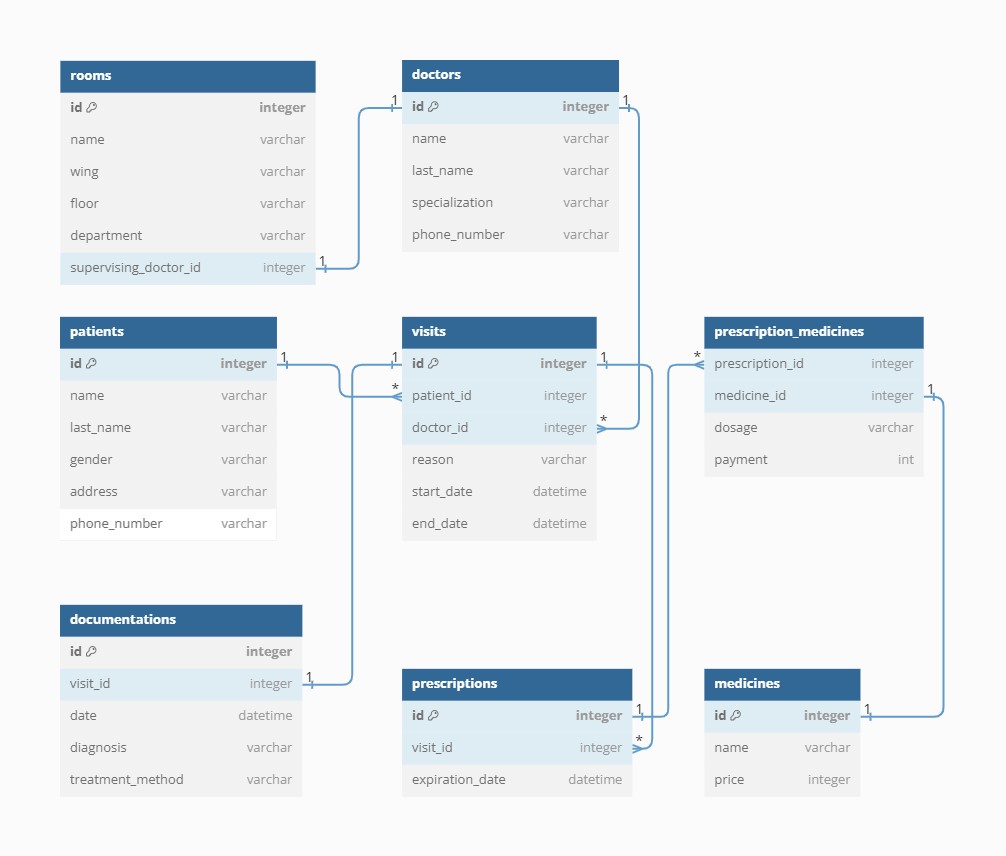
* prescription\_medicines.prescription\_id (Tabela prescription\_medicines) > prescriptions.id (Tabela prescriptions)

Relacja wiele-do-jednego (>). Wiele leków może być przypisanych do jednej recepty.

* prescription\_medicines.medicine\_id (Tabela prescription\_medicines) - medicines.id (Tabela medicines)

Relacja jeden-do-jednego (-). Oznacza to, że każde przypisanie leku na recepcie może być powiązane tylko z jednym lekiem, który jest zdefiniowany w tabeli leków.

Do wizualnego przedstawienia tabel oraz powiązań pomiędzy nimi, stworzono poniższy diagram ERD.



Rysunek 1 Diagram ERD