

**PAŃSTWOWA WYŻSZA SZKOŁA ZAWODOWA**

**W NOWYM SĄCZU**

**INSTYTUT TECHNICZNY**

**PRACA DYPLOMOWA**

**APLIKACJA WEBOWA UOŻLIWIAJĄCA WYBÓR  
ODPOWIEDNIEGO LEKARZA SPECJALISTY**

**Autor: Michał Pawlikowski**

**Kierunek: Informatyka Stosowana**

**Numer albumu: 29653**

**Promotor: dr inż. Stanisława Plichta**

**Akceptacja promotora: .....**

**data i podpis**

**NOWY SĄCZ 2023**



## Spis treści

Wstęp.....	5
1. Cel i zakres pracy .....	6
2. Analiza wymagań projektu.....	7
2.1. Wymagania funkcjonalne .....	7
2.2. Wymagania niefunkcjonalne .....	7
2.3. Diagram przypadków użycia .....	8
2.4. Diagram ERD .....	8
3. Zastosowane technologie .....	9
3.1. HTML.....	9
3.2. CSS .....	9
3.3. MySQL .....	9
3.4. PHP.....	9
3.5. JavaScript.....	9
3.6. Laravel.....	10
Literatura .....	11



## Wstęp

W dzisiejszych czasach znalezienie odpowiedniego lekarza specjalisty, który trafnie zdiagnozuje nasze problemy i pomoże nam wyjść z choroby to nie wszystko. Dla każdej grupy wiekowej oraz specjalizacji jest ważne co innego. Załóżmy, że jesteśmy po ciężkim urazie kolana i musimy udać się do fizjoterapeuty. Tutaj nie przyda się tylko pokazanie ćwiczeń, które musimy wykonywać. Cenną rzeczą w tym przypadku jest wytłumaczenie jak działają nasze stawy, jak musimy zachowywać się w danych sytuacjach, żeby jeszcze bardziej nie pogłębić swojej kontuzji. Z kolei rozpatrując przykład ze względu na grupy wiekowe, weźmy pod uwagę najmłodszych. Kluczową cechą gdy sugerujemy się wyborem lekarza dla naszych pociech jest to, czy będzie miał on odpowiednie podejście do dzieci.

W tej pracy skupino się właśnie na problemie, który dotyczy nas wszystkich, przy wyborze odpowiedniego specjalisty. Oceny i opinie danego lekarza pomogą nam zdecydować, czy to właśnie jemu chcemy powierzyć nasze życie.

## 1. Cel i zakres pracy

Celem niniejszej pracy jest utworzenie aplikacji webowej umożliwiającej znalezienie odpowiedniego lekarza specjalisty w danym obszarze. Wykorzystując zebraną wiedzę, zostanie zaprojektowany prosty interfejs, który umożliwi łatwe wybranie interesującej nas lokalizacji, oraz danej specjalizacji. Powstanie również specjalna zakładka dla lekarzy, która pozwoli stworzyć mu swój własny profil. Aby stało się to możliwe, stworzona aplikacja zostanie połączona z bazą przechowującą dane. Aplikacja podzielona zostanie na trzy rodzaje:

- Panel klienta – podstawowy interfejs, ułatwiający znalezienie odpowiedniego specjalisty
- Panel lekarza – panel ukazujący swój własny edytowalny profil, który będzie zawierał: specjalizację, adres, kontakt, godziny pracy, oraz cennik
- Panel administratora – administrator będzie mógł zarządzać użytkownikami, opiniami, weryfikacja lekarzy

Zakres pracy:

- Utworzenie przyjemnego i łatwego w obsłudze interfejsu aplikacji,
- Utworzenie połączenia aplikacji z bazą danych,
- Zaimplementowanie uwierzytelniania użytkowników,
- Możliwość stworzenia i edycji profilu lekarza
- Możliwość wystawienia opinii
- Przetestowanie aplikacji przy pomocy grupy użytkowników,
- Dołączenie instrukcji pomagającej nowym użytkownikom zapoznać się z działaniem aplikacji,
- Wnioski podsumowujące pracę nad aplikacją z wykorzystanymi technologiami.

## **2. Analiza wymagań projektu**

### **2.1. Wymagania funkcjonalne**

Wymagania funkcjonalne są zbiorem wszystkich funkcjonalności, które były wymieniane na wstępnym etapie analizy. Jest to opis wszystkich funkcji, które będzie oferowała aplikacja.

#### **Wymagania zwykłego użytkownika**

- Jako użytkownik mogę znaleźć lekarza w wyznaczonym obszarze
- Jako użytkownik mogę udzielić opinii
- Jako użytkownik mogę wyeksportować dane i adres lekarza do pliku PDF

#### **Wymagania lekarza**

- Jako lekarz mogę zalogować i wylogować się z własnego konta
- Jako lekarz mogę stworzyć i edytować swój własny profil
- Jako lekarz mogę skasować własne konto

#### **Wymagania administratora**

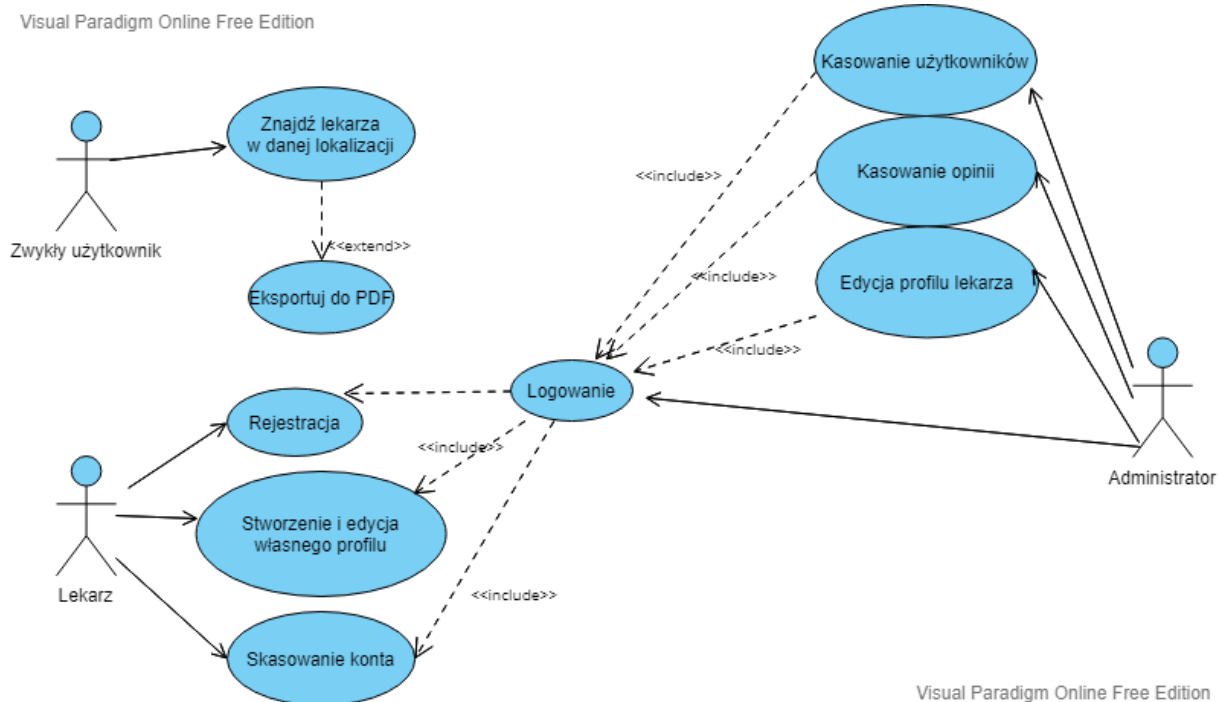
- Jako administrator mogę zalogować i wylogować się z własnego konta
- Jako administrator mogę skasować opinie
- Jako administrator mogę skasować użytkownika
- Jako administrator mogę edytować profil danego lekarza

### **2.2. Wymagania niefunkcjonalne**

Wymagania niefunkcjonalne są zbiorem wszystkich wymagań jakie mają być odnośnie systemu, bezpieczeństwa, wydajności itp.

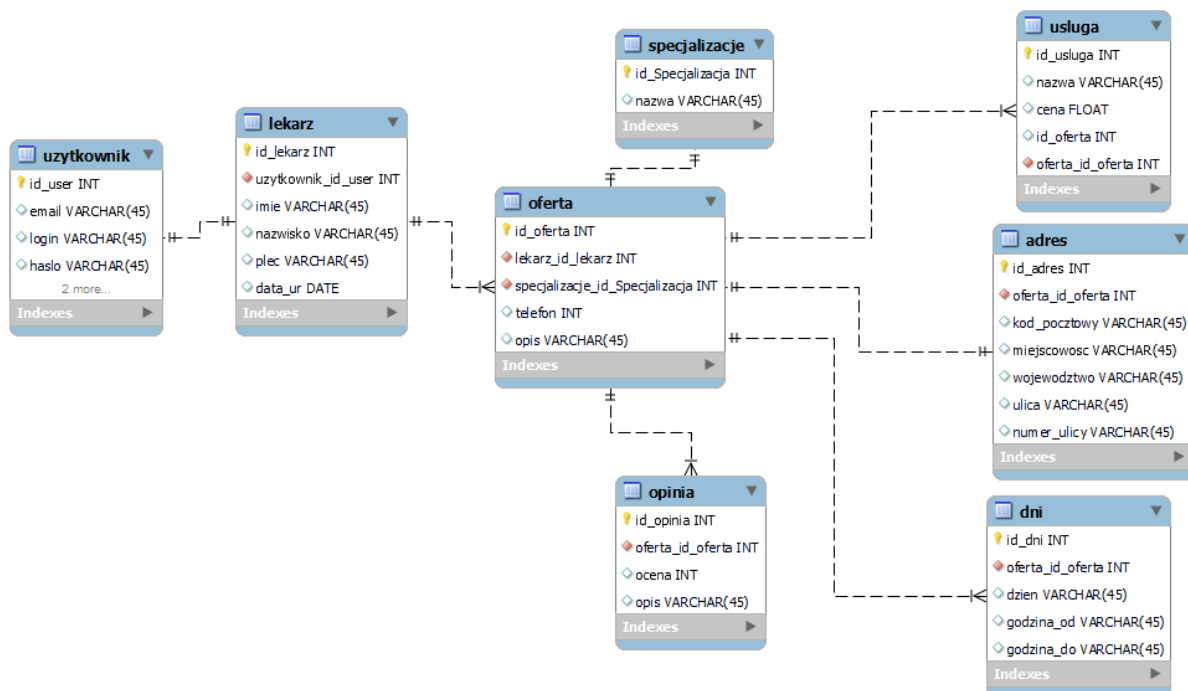
- Niezawodność
- Poufność i stopień zabezpieczenia danych
- Przejrzysty interfejs

## 2.3. Diagram przypadków użycia



Rysunek 1 - Diagram przypadków użycia

## 2.4. Diagram ERD



Rysunek 2 - Diagram ERD



### **3. Zastosowane technologie**

#### **3.1. HTML**

HTML (HyperText Markup Language) jest językiem hipertekstowych znaczników stosowanych do tworzenia hipertekstowych dokumentów, najczęściej będącymi podstawą stron internetowych. Składa się z kilku kluczowych komponentów: znaczników, typów danych, referencji znakowych, odwołań w postaci encji i deklaracji typu dokumentu.

#### **3.2. CSS**

Kaskadowe arkusze stylów (Cascading Style Sheets) język służący do opisu formy prezentacji (wyświetlania) stron WWW. Arkusz stylów CSS to lista dyrektyw (tzw. reguł) ustalających w jaki sposób ma zostać wyświetlana przez przeglądarkę internetową zawartość wybranego elementu (lub elementów) (X)HTML lub XML. Można w ten sposób opisać wszystkie pojęcia odpowiedzialne za prezentację elementów dokumentów internetowych, takie jak rodzina czcionek, kolor tekstu, marginesy, odstęp międzywierszowy lub nawet pozycja danego elementu względem innych elementów bądź okna przeglądarki.

#### **3.3. MySQL**

Jest to wolnodostępny, otwartoźródłowy system zarządzania relacyjnymi bazami danych. Był pisany raczej z myślą o szybkości niż kompatybilności ze standardem SQL – przez dłuższy czas MySQL nie obsługiwał nawet transakcji, co było zresztą głównym argumentem przeciwników tego projektu. Wprowadza również swoje rozszerzenia i nowe elementy języka. Serwer MySQL dostępny jest dla wszystkich popularnych platform systemowych i różnorakich architektur procesorów.

#### **3.4. PHP**

Interpretowany, skryptowy język programowania zaprojektowany do generowania stron internetowych i budowania aplikacji webowych w czasie rzeczywistym. PHP jest najczęściej stosowany do tworzenia skryptów po stronie serwera WWW, ale może być on również używany do przetwarzania danych z poziomu wiersza poleceń, a nawet do pisania programów pracujących w trybie graficznym

#### **3.5. JavaScript**

Skryptowy oraz wieloparadygmatowy język programowania, stworzony przez firmę Netscape. Najczęściej spotykanym zastosowaniem języka JavaScript są strony internetowe. Skrypty te służą najczęściej do zapewnienia interakcji poprzez reagowanie na zdarzenia, walidacji danych wprowadzanych w formularzach lub tworzenia złożonych efektów

wizualnych. Skrypty JavaScriptu uruchamiane przez strony internetowe mają znacznie ograniczony dostęp do komputera użytkownika.

### **3.6. Laravel**

Framework do aplikacji internetowych napisany w języku PHP bazujący na wzorcu architektonicznym Model-View-Controller. Laravel udostępnia m.in. modułowy system budowania aplikacji z dedykowanym menedżerem zależności, różne sposoby dostępu do relacyjnych baz danych, narzędzia pomagające we wdrażaniu i utrzymaniu aplikacji oraz jego ukierunkowaniem na cukier syntaktyczny.

## Literatura

[html] <https://pl.wikipedia.org/wiki/HTML>

[css] [https://pl.wikipedia.org/wiki/Kaskadowe\\_arkusze\\_styl%C3%B3w](https://pl.wikipedia.org/wiki/Kaskadowe_arkusze_styl%C3%B3w)

[Mysql] <https://pl.wikipedia.org/wiki/MySQL>

[Php] <https://pl.wikipedia.org/wiki/PHP>

[JavaScript] <https://pl.wikipedia.org/wiki/JavaScript>

[laravel] <https://pl.wikipedia.org/wiki/Laravel>