|  |  |
| --- | --- |
| **Nazwa projektu:** | System obsługi gabinetu stomatologicznego |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Osoby realizujące projekt:** | | | | |
|  | | | | |
|  | *imię i nazwisko* | *nr indeksu* | *Grupa* | *adres email* |
|  | Tomasz Kamiński | Inf84817 | Sr1510 | [kaminski.tomasz.a@gmail.com](mailto:kaminski.tomasz.a@gmail.com) |
|  | Wojciech Kuć | Inf84834 | Sr1510 | [wojciech.kuc@10g.pl](mailto:wojciech.kuc@10g.pl) |

# Opis projektu

Celem projektu jest stworzenie aplikacji – systemu do obsługi gabinetu stomatologicznego. Główną ideą aplikacji i jej podstawowym założeniem jest stworzenie elektronicznej wersji kartoteki pacjentów, która obecnie w większości gabinetów stomatologicznych przechowywana jest w tradycyjnej, papierowej formie. Projektowi przyświeca zatem cel zinformatyzowania gabinetu stomatologicznego i ułatwienia dentyście lub jego asystentce pracy.

Projektowany system wg założeń autorów ma skupiać się przede wszystkim na modelowaniu kartoteki pacjentów, ewidencji stanu ich uzębienia w czasie, w tym przebytych chorób związanych z poszczególnymi zębami i podjętych przez stomatologa działań zmierzających do wyleczenia pacjenta. Mniejszy nacisk zostanie położony na prowadzenie terminarza spotkań i pomoc przy planowaniu wizyt. Funkcjonalność projektu

Spis planowanych funkcjonalności z podziałem na obowiązkowe (min. 10) i opcjonalne (min. 5).

|  |  |  |  |
| --- | --- | --- | --- |
| *Lp* | *Opis funkcjonalności* | *Typ* | *Status* |
| 1 | Dodawanie nowych pacjentów, usuwanie istniejących pacjentów | obowiązkowa | W trakcie |
| 2 | Edycja danych personalnych pacjentów (imię, nazwisko, PESEL, etc.) | obowiązkowa | W trakcie |
| 3 | Dodawanie danych o odbytych wizytach danego pacjenta | obowiązkowa | W trakcie |
| 4 | Przeglądanie historii wizyt danego pacjenta | obowiązkowa | W trakcie |
| 5 | Nanoszenie informacji o stanie poszczególnych zębów na graficznej mapie uzębienia | obowiązkowa | W trakcie |
| 6 | Przeglądanie historii stanu poszczególnych zębów danego pacjenta | obowiązkowa |  |
| 7 | Edycja listy dostępnych operacji związanych z uzębieniem, w tym możliwość ustalenia ceny poszczególnych operacji | obowiązkowa |  |
| 8 | Wycena wizyty na podstawie operacji, które zostały podczas niej wykonane | obowiązkowa |  |
| 9 | Załączanie do ewidencji wizyt dodatkowych dokumentów, np. zdjęć rentgenowskich | obowiązkowa |  |
| 10 | Możliwość wydruku poszczególnych dokumentów wchodzących w skład kartoteki (dane pacjenta, historia stanu uzębienia) | obowiązkowa |  |
| 11 | Prowadzenie kalendarza wizyt: dodawanie przez lekarza nowej wizyty opisanej datą, godziną oraz umówionym pacjentem | opcjonalna |  |
| 12 | Sprawdzanie terminu wizyt przez pacjenta, w tym możliwość umawiania w danym terminie | opcjonalna |  |
| 13 | Przechowywanie zewnętrznych plików w kartotece pacjenta | opcjonalna |  |
| 14 | Zarządzanie użytkownikami systemu | opcjonalna |  |
| 15 | Obsługa wielu lekarzy, w tym możliwość przypisania pacjenta do jednego lub wielu z nich | opcjonalna |  |

# Koncepcja realizacji

Projekt w zamierzeniu ma stanowić aplikację webową. Docelowo klient będzie korzystał z przeglądarki internetowej.

Aplikacja zostanie zrealizowana w technologii Java/J2EE z wykorzystaniem frameworków Spring oraz Hibernate w warstwie modelu i dostępu do bazy danych. Do realizacji warstwy widoku zostanie użyta biblioteka Spring MVC. Po stronie serwera znajdować się będzie baza danych, przechowująca dane wprowadzane przez lekarza; wybrano MySQL z silnikiem InnoDB, rozwijaną przez firmę Oracle.

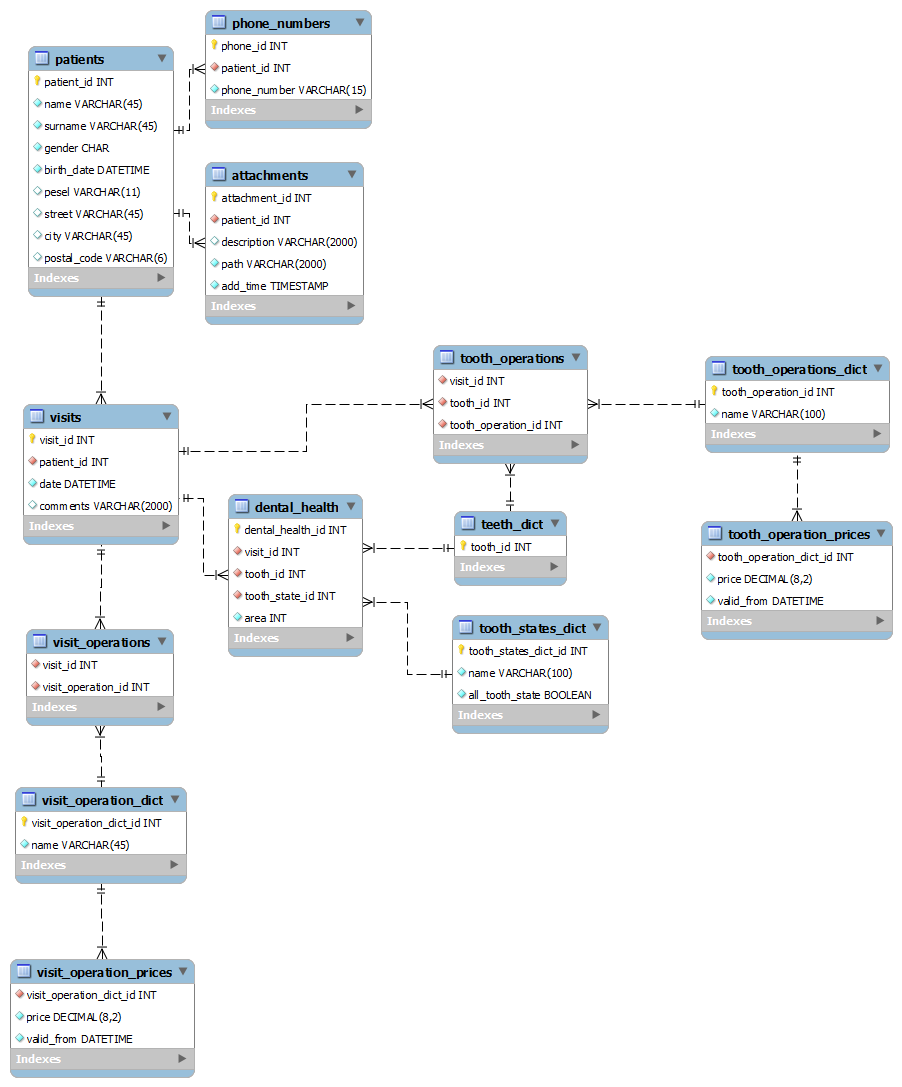
Po stronie klienta użyte zostaną technologie HTML/CSS, jQuery. Dane będą wprowadzane przez lekarza za pomocą formularzy HTML. Ich walidacja będzie zrealizowana po stronie zarówno serwera (ograniczenia integralnościowe, zgodność z modelem) jak i przeglądarki (sprawdzanie błędów syntaktycznych, np. ilość cyfr w PESELu).

Do implementacji mapy uzębienia zostanie użyty element mapy HTML. Układ uzębienia będzie zapisany w notacji FDI World Dental Federation.

# Zastosowane technologie

* MySQL, Spring, Spring MVC, Hibernater, HTML, CSS, jQuery
* Uzasadnić wybór, jakie są zalety i ograniczenia wynikające z takiego a nie innego wyboru

# Dokumentacja techniczna

* Wstępny schemat bazy danych
* Opis zastosowanych rozwiązań
* Specyfikacja interfejsów

# Materiały źródłowe

* Adresy WWW
* Artykuły
* Książki