|  |  |
| --- | --- |
| **Nazwa projektu:** | Bank Krwi |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Autorzy specyfikacji:** | | | | |
|  | | | | |
|  | *imię i nazwisko* | *nr indeksu* | *grupa* | *adres email* |
|  | Mikołaj Wawrzyniak | 106062 | 1 | g.m.wawrzyniak@gmail.com |
|  | Adrian Karalus | 106068 | 1 | adrian.karalus@gmail.com |

Opis projektu

Celem projektu jest zaprojektowanie i implementacja serwisu dla banku krwiodawstwa. Ma on pozwalać na zbieranie danych z poszczególnych stacji krwiodawstwa na temat pobranych zasobów krwi, ponadto ma również pozwalać monitorować ogólne zapasy krwi. System będzie aplikacją webową poprzez, którą możliwe będzie wprowadzanie i modyfikacja niezbędnych danych. Autorz chcą go wykonać jako serwis typu REST, będący aplikacją monolityczną.

Funkcjonalność projektu

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

|  |  |  |  |
| --- | --- | --- | --- |
| *Lp* | *Opis funkcjonalności* | *Typ* | *Status* |
| 1 | Dodawanie kont użytkowników | obowiązkowa |  |
| 2 | Autoryzacja użytkowników | obowiązkowa |  |
| 3 | Zarządzanie kontami użytkowników i zakresem uprawnień | obowiązkowa |  |
| 4 | Dodawanie i modyfikacja danych na temat zebranej krwi | obowiązkowa |  |
| 5 | Prezentacja danych o stanie zapasów krwii | obowiązkowa |  |
| 6 | Walidacja wprowadzanych danych | obowiązkowa |  |
| 7 | Obsługa zapytań o migracje zapasów krwii | obowiązkowa |  |
| 8 | Powiadomienia w ramach serwisu o zmianie w zapasach krwi | obowiązkowa |  |
| 9 | Prezentacja danych o honorowych krwiodawcach | obowiązkowa |  |
| 10 | Prezentacja danych o procentowym udziale poszczególnych grup krwi dla każdej placówki | obowiązkowa |  |
| 11 | Powiadomienia mailowe o zmianie stanu zapasów krwi | opcjonalna |  |
| 12 | Wymiana wiadomości między stacjami krwiodastwa | opcjonalna |  |
| 13 | Mechanizm odzyskiwania haseł | opcjonalna |  |
| 14 | Pobranie danych o stanie krwii do pdf | opcjonalna |  |
| 15 | Pobranie danych o stanie krwi w formacie .xls | opcjonalna |  |

Koncepcja realizacji

Jak wspomniano we wstępie projekt będzie monolityczną aplikacją internetową, zrealizowaną w architekturze REST. Z całości systemu można wyodrębnić trzy elementy

1. Baza danych
2. Aplikacja serwerowa
3. Interfejs użytkownika

**DB**

JSON

Interfejs użytkownika będzie stroną typu single page application, zrealizowaną z wykorzystaniem *Mean.js*, który jest zbiorem technologii dostarczający kompleksowy zestaw narzędzi do tworzenia tego typu aplikacji internetowych.

Zastosowane technologie

*Mean.js* składa się z następujących technologii:

1. Angular.js
2. MongoDB
3. Node.js
4. Twitter Bootstrap

Na wybór tych technologii wpłynęły dwa czynniki, po pierwsze, podejście single page application, znacząco poprawia komfort użytkownika, poprzez zwiększenie responsywności aplikacji klienckiej w porównaniu do typowy aplikacji internetowych, które wymagają częstego ładowania całych widoków klienckich. Drugim powodem wybrania właśnie *Mean.js,* była chęć poznania nowej technologii przez autorów, do tej pory pracujących głównie z Ruby on Rails i .Net.

Wnioski z realizacji projektu

* Mean.js zwłaszcza po stronie serwera, jest mało intuicyjny, jeśli idzie o proces debugowania, co utrudnia usuwanie ewentualnych błędów.
* Jest to narzędzie dostarczające kompleksowe rozwiązanie, więc proces tworzenia aplikacji, o ile całość narzędzi jest dobrze znana programiście, może być bardzo szybki.

Dokumentacja techniczna

System jest aplikacją modułową składającą się z następujących części:

1. Core – jest odpowiedzialny za layout całości, po nad to stanowi trzon aplikacji, do której dodawane są kolejne moduły.
2. User – odpowiada za autoryzację i rejestrację użytkowników. Dba również o pilnowanie poziomu uprawnień poszczególnych użytkowników.
3. Blooddonation – moduł odpowiada za poszczególne donacje krwi, pozwala na wszelkie operacje typu CRUD na tymże zasobie.
4. Donnor – CRUD’owy moduł zarządzania krwiodawcami, obsługuje również funkcjonalność listy honorowych krwiodawców
5. Station – moduł zajmujący się obsługą stacji krwiodawstwa, zarządza zasobami krwi należącymi do danej stacji i pracownikami do niej przypisanymi.
6. Transfers – moduł komunikujący stacje między sobą w zakresie zapytań o transfer krwi po między nimi.

Po stronie klienckiej każdy moduł składa się z:

* Kontrolera, wywołującego metody serwisów w celu pobrania danych i dostarcza je do widoków. Po za tym dostarcza metod pozwalających na edycję i tworzenie nowych zasobów.
* Serwisów, odpowiedników modeli z wzorca MVC, ich odpowiedzialnością jest dostarczanie danych.
* Widoków, będących dokumentami html, z osadzonymi dyrektywami angular.js.
* Arkuszów styli css
* Dyrektyw, stanowiących samodzielne komponenty do osadzenia w widokach.
* Plików konfiguracyjnych
  + Deklarujących elementy menu w pasku nawigacyjnym strony.
  + Definiujące stany i odpowiadające im url, a także kontrolery i widoki.

Materiały źródłowe

* <http://meanjs.org/>
* https://www.youtube.com/watch?v=XHpMH\_5n2fQ

Materiały źródłowe

* <https://github.com/angular/angular-seed>
* <http://www.krew.gda.pl/>
* https://angularjs.org/