



Serwis przechowywania oraz czytania książek on-line
Dokumentacja

Autor: Vitaliy Stinskiy

Kraków, 2012.01.11

Spis treści

Opis	3
Wymagania funkcjonalne	3
Wymagania niefunkcjonalne	3
Implementacja	3
Baza danych	4
Funkcjonalność	5
Użyte narzędzia i technologie.....	8
Podsumowanie.....	8

Opis

Co raz częściej w poszukiwaniu nowej książki ludzie sięgają nie do najbliższej księgarni, czy sklepu „empik”, tylko do internetu. Za ostatnie kilka lat tradycyjne papierowe książki znacznie straciły swoją popularność. W większej mierze, ten fakt jest spowodowany tym, że znacznie zwiększyła się liczba posiadaczy smartfonów z dużymi wyświetlaczami oraz tak zwanych ebook`ów.

Idea serwisu, o którym jest mowa, polega na tym żeby dać możliwość użytkownikom komputerom, laptopów oraz smartfonów mieć możliwość przechowywac i czytać swoje książki on-line. Również dzielić się opinią o przeczytanych książkach z innymi użytkownikami serwisu. Oraz czytać książki które udostępniają inni użytkownicy. Jedynie co jest wymagane od użytkownika, posiadanie urządzenia które obsługuje współczesną przeglądarkę oraz dostęp do Internetu, dzięki rozwojowi technologii, dzisiaj te wymagania spełnia nawet najtańszy smartfon.

Wymagania funkcjonalne

- Główne wymaganie funkcjonalne, wynika z zasadniczej idei serwisu – **możliwość przechowywania książek**.
- Możliwość dodawania książek, tj. upload pliku tekstowego na serwer.
- Wygodne zarządzanie katalogiem dodanych książek.
- Dostęp do książek które udostępnili inni użytkownicy
- Możliwość czytania książek on-line, za pomocą przeglądarki.
- Możliwość zostawienia opinii o książce.

Wymagania нефunkcjonalne

Przechowywanie książek oznacza przechowywanie plików na serwerze, więc niezbędny jest dostęp do dysku z możliwością zapisu, jak i wystarczająco duża ilość wolnego miejsca. Baza danych powinna mieć możliwość przechowywania tekstu o dużym rozmiarze w jednym polu.

Wybrane przez ze mnie technologie dla implementacji serwisu, narzuca pewne ograniczenia:

- Web-serwer z możliwością obsługi skryptów PHP
- PHP wersji 5.2 lub nowszej

Implementacja

Serwis projektowałem na podstawie wzorca MVC. Więc przedstawienie diagramu klas wydaje mi się zbędny. Opiszę w skrócie najważniejsze klasy:

BookController – najważniejsza klasa, jest odpowiedzialna za główne funkcji serwisu. Zawiera metody obsługujące dodawanie, edycje i usuwania książek. Przegląd katalogu własnych książek. Przegląd książek udostępnionych przez innych użytkowników. Głosowanie na książki. Czytanie książki.

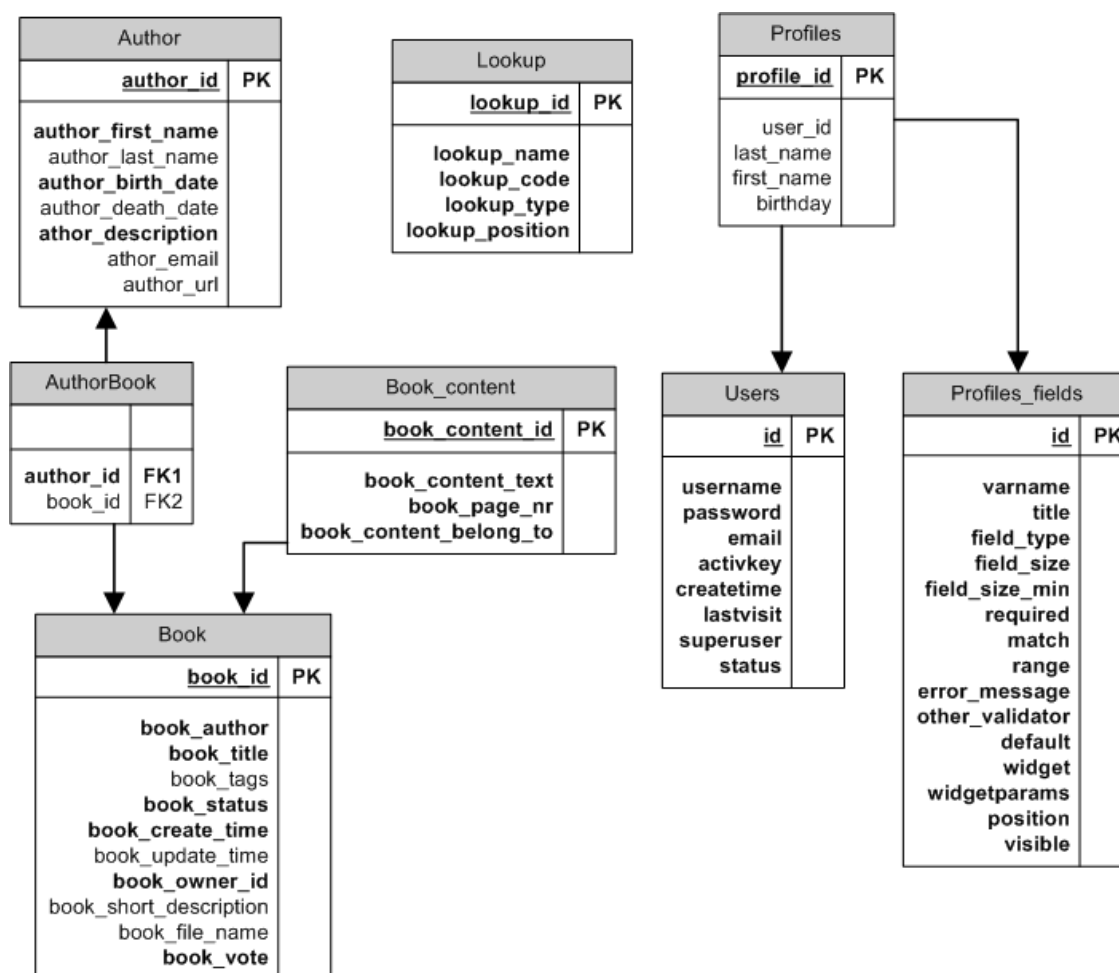
AuthorController – ta klasa odpowiada za czynności związane z dodaniem, edycją i usuwaniem autorów. Również jak i wyświetlanie profilu autora, z listą utworów. Filtrowanie autorów według pierwszej litery nazwiska.

AjaxController – zawiera metody obsługujące asynchroniczne zapytania. Np. głosowanie

UserModule – klasa zawiera całą logikę obsługi użytkowników. Rejestrację nowego użytkownika, logowanie, wylogowanie, przewrócenia hasła, edycje profilu, etc.

Baza danych

Na Rys.1. został przedstawiony schemat bazy danych. Baza danych zawiera 8 tabel. Tabela Book przechowuje dane o książce, tytuł, krótki opis, datę utworzenia, liczbę głosów, etc. Tabela Author zawiera informacje o autorach książek. Jest ona powiązana z tabelą Book relacją wiele do wielu, za pomocą tabeli łączącej AuthorBook. Zawartość książki jest przechowywana w tabeli Book_content. Całą książkę jest rozbijana na części o długości 850 znaków, i zapisywana do poszczególnych rekordów tabeli Book_content. Jest to zrobione z myślą o optymalizację czytania książki, żeby nie pobierać całej zawartości książki a tylko jej część.

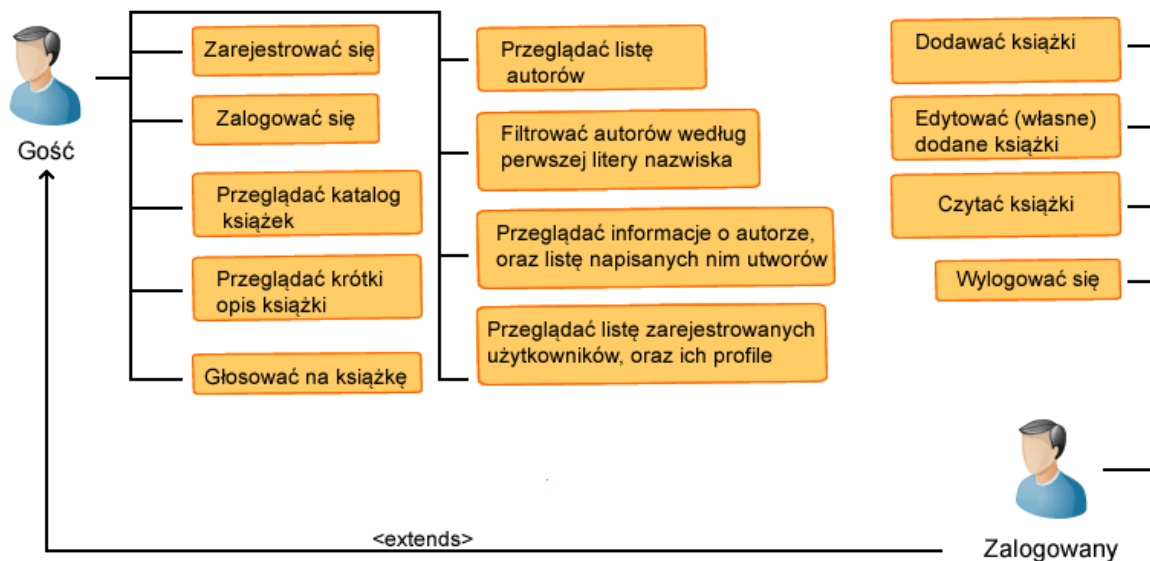


Rys.1. Schemat bazy danych

Tabela Lookup zawiera wpisy które będą używane do różnych celów, np. przechowywania ustawień. Tabela Profiles, Users oraz Profiles_fields zawierają informacje dotyczącą użytkowników serwisu, jak również i informacje niezbędne do uwierzytelniania i autoryzacji.

Funkcjonalność

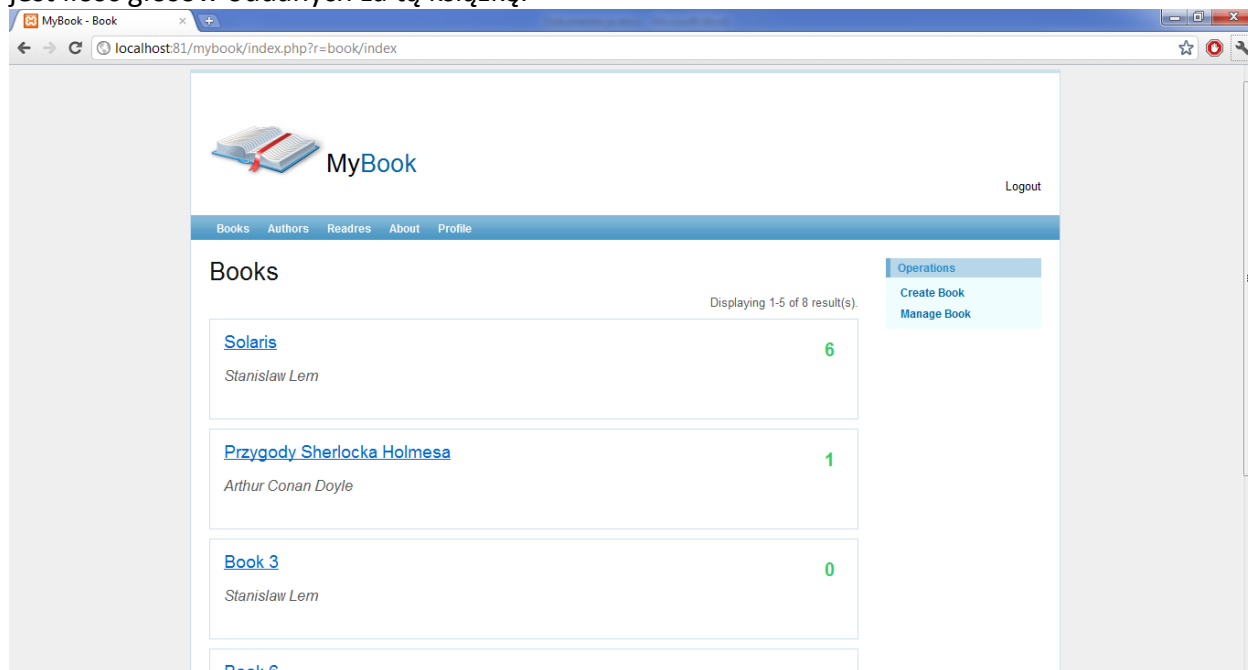
Na Rys.2. przedstawiłem diagram przypadków użycia który ilustruje funkcjonalność serwisu.



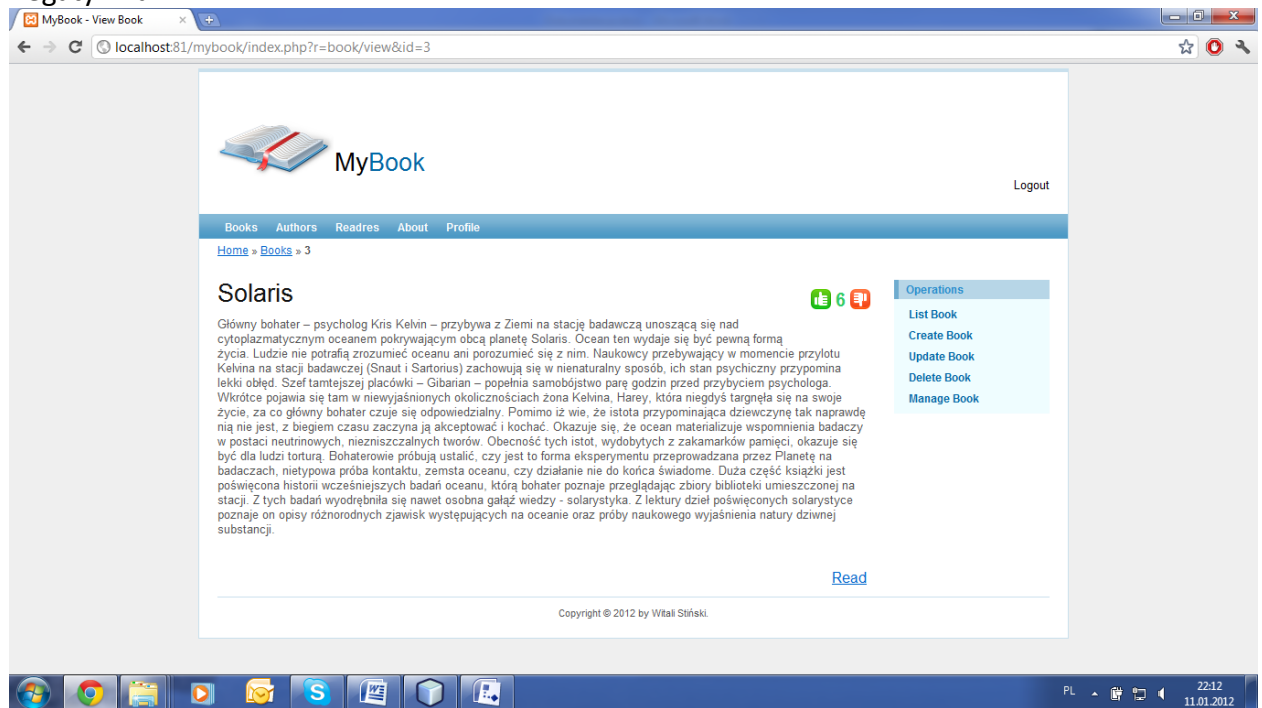
Rys.2 . Diagram przypadków użycia

Poniżej również umieściłem kilka ekranów serwisu z opisami.

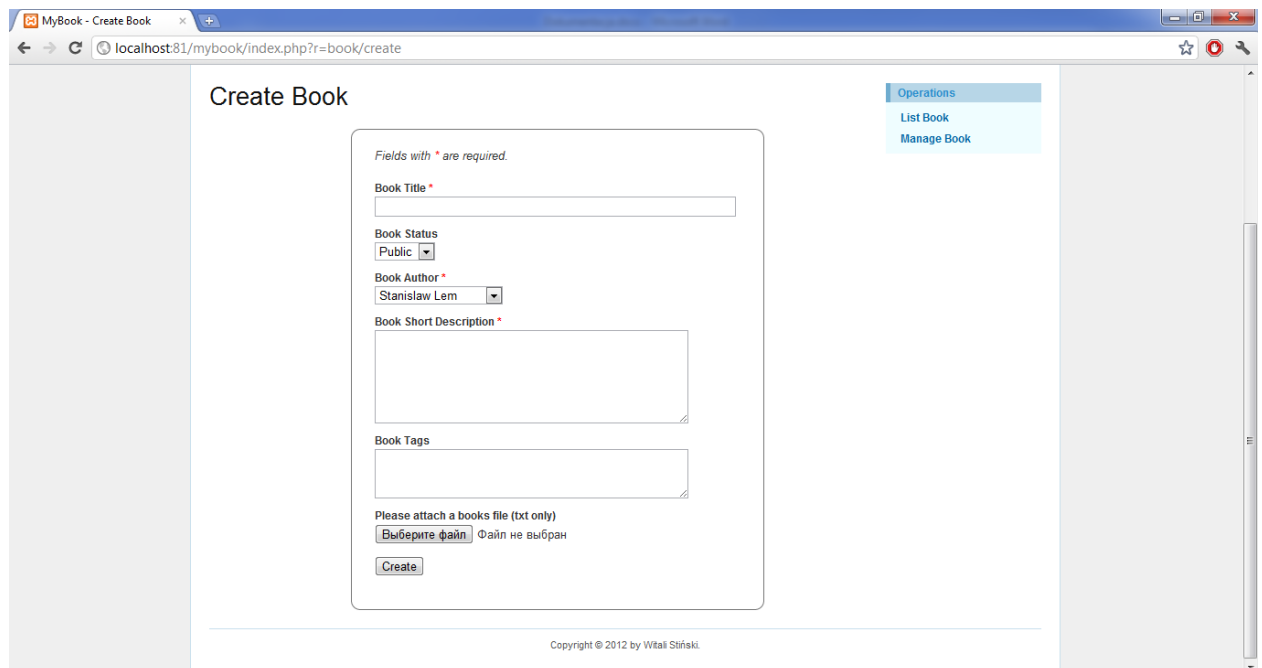
Strona główna, użytkownik zalogowany. Po lewej stronie znajduje lista książek posortowana według liczby głosów, po prawej menu zarządzania katalogiem książek. Cyfry w opisie strony to jest ilość głosów oddanych za tą książkę.



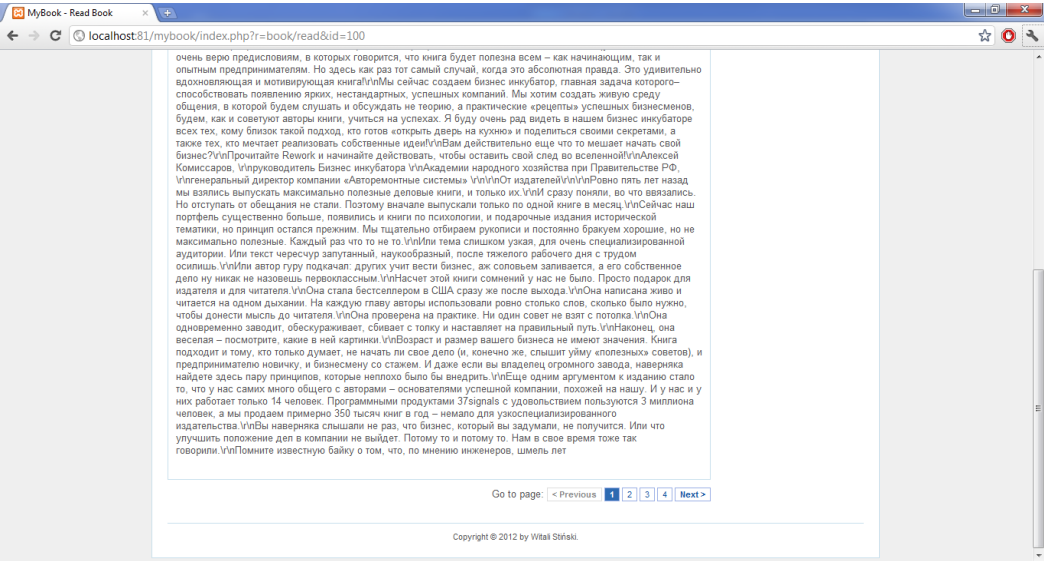
Krótki opis książki, oraz możliwość głosowania. Głosować można tylko jeden raz. Głos może być pozytywny lub negatywny. Liczba głosów jest zielona jeżeli jest pozytywna i czerwona jeżeli negatywna.



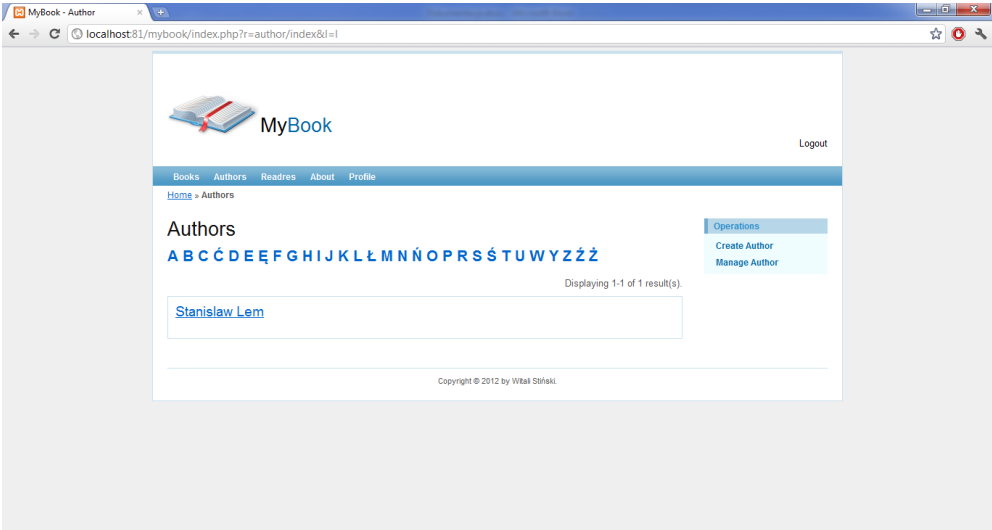
Formularz dodawania nowej książki. Plik może być tylko w formacie .txt. Po przesłaniu pliku na serwer jest on wczytywany do bazy danych częściami o długości 850 znaków:



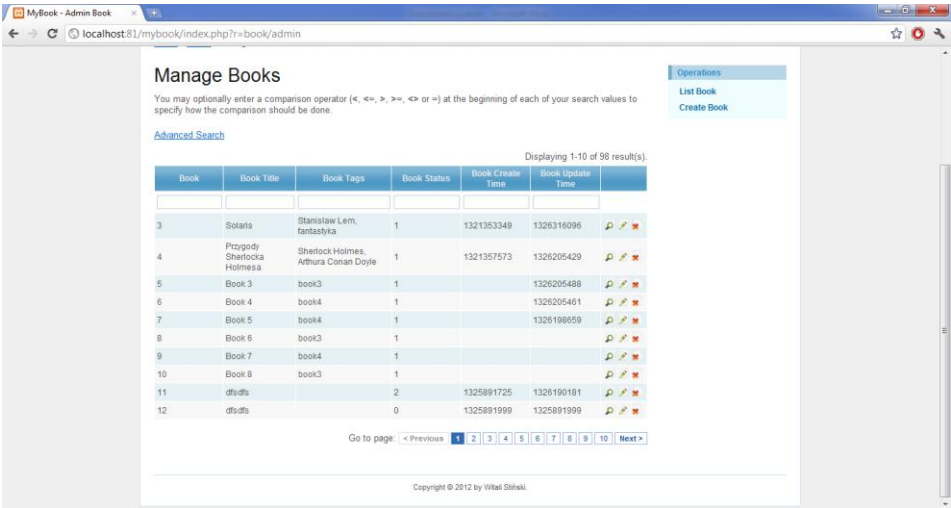
Czytanie książki:



Lista autorów według pierwszych liter nazwiska:



Zarządzanie katalogiem książek:



Użyte narzędzia i technologie

Dla implementacji serwisu użyłem frameworku YiiFramework. Projekt jest napisany w języku PHP wersji 5.3. Użyłem bazy danych MySQL z silnikiem InnoDB. W warstwie prezentacji używałem HTML, CSS oraz biblioteka JavaScript – jQuery. Również użyłem technologii asynchronicznych zapytań AJAX.

Dla napisania kodu używałem środowiska NetBeans 7.0, oraz systemu kontroli wersji Git. Projekt jest udostępniony w publicznym repozytorium GitHub: <https://github.com/vitsv/MyBook>

Podsumowanie

Podsumowując swoją pracę nad projektem, mogę powiedzieć, że nawet za taki krótki czas udało się zrealizować w pełni funkcjonalny serwis. W projekcie zaimplementowałem wszystkie funkcje które zostały przedstawione na początku, za wyjątkiem drobnych funkcjonalności które niosą bardziej kosmetyczny charakter niż funkcyjny. Między innymi to np. zawijanie słów.

Osiągnięcie takich wyników, za taki krótki czas jest możliwe głównie za pomocą współczesnych frameworków. Zdecydowałem się użyć w swoim projekcie nowy dla mnie Php framework Yii. Jest on stosunkowo „młody” lecz posiada olbrzymi możliwości, zawiera w sobie wiele innowacyjnych i nowoczesnych pomysłów.

Dzięki Java Script frameworkowi jQuery obsłużyć Ajax zapytania można w łatwy i elastyczny sposób, framework oferuje dla tego duże możliwości. Również jQuery udostępnia wiele narzędzi do stworzenia interakcyjnego interfejsu po stronie klienta.

W obecnym stanie, mogę nadać projektowi wersje Beta. Co oznacza że jeszcze dużo rzeczy można ulepszyć oraz dodać. Przede wszystkim to obsługa różnych formatów plików, które będą przysyłać na serwer użytkownicy (obecnie jest obsługiwany tylko .txt). Dodać więcej socjalnych elementów, żeby zwiększyć komunikację użytkowników serwisu między sobą. Również należy dopracować interfejs czytanie książek, dodać możliwość zawijania słów. Czytanie książek powinno być wygodnym na każdym urządzeniu, czy to jest laptop, czy smartfon.