

TIN – Opis projektu

Projekt realizuje funkcjonalność dodawania nowych książek do oferty sklepu internetowego i składa się z 3 modułów:

- baza danych MySQL
- aplikacja server-side
- aplikacja client-side

Aby uruchomić projekt poprawnie, należy:

1. Stworzyć działającą bazę danych na serwerze MySQL wykorzystując skrypt *tworzenie_bazy.sql*, który znajduje się w katalogu głównym projektu. Powinna zostać założona i wypełniona danymi baza o nazwie *tin_projekt*.
2. Skonfigurować połączenie z bazą danych w aplikacji server-side. W tym celu należy wpisać poprawne opcje konfiguracyjne w pliku *[KATALOG PROJEKTU]\server_side_app\config\database.js*. Zalogowany do bazy użytkownik musi mieć pełne uprawnienia dostępu do bazy.
3. Uruchomić aplikację server-side. W tym celu należy uruchomić wiersz poleceń, zmienić aktywny katalog na *[KATALOG PROJEKTU]\server_side_app* i uruchomić polecenie *nodemon*. Aby sprawdzić czy aplikacja server-side pracuje poprawnie, można w przeglądarce wejść na adres <http://localhost:4000/>. Powinna wyświetlić się wiadomość „hello from the backend server”.
4. Uruchomić aplikację client-side. W tym celu należy uruchomić nowe okno wiersza poleceń, zmienić aktywny katalog na *[KATALOG PROJEKTU]\client_side_app* i uruchomić polecenie *npm start*. Po kilkunastu sekundach powinno automatycznie wyświetlić się nowe okno przeglądarki z witryną <http://localhost:3000/>.

Opis aplikacji:

Aplikacja serwerowa wykorzystuje ORM Sequelize, pobiera i dodaje dane z/do bazy MySQL. Do walidacji danych wykorzystałem odpowiednio przygotowane modele danych (podkatalog *models*). Głównym zadaniem aplikacji serwerowej jest odpowiadanie na żądania GET i POST wysyłane przez aplikację kliencką i odpowiednio pobieranie lub dodawanie danych do bazy.

Aplikacja kliencka realizuje funkcjonalności podglądu dostępnych i dodawania nowych książek do oferty sklepu internetowego (wysyła żądania GET i POST do aplikacji serwerowej i przetwarza odpowiednio odpowiedzi). Użytkownik ma wgląd do książek znajdujących się w

bazie (*Lista książek w sklepie*). Wyświetlany jest tytuł oraz imię i nazwisko autora książki – dane są pobierane z tabel bazy danych.

W nagłówku strony znajduje się pole logowania użytkownika (to przykładowa funkcjonalność – wpisujemy tylko nazwę użytkownika). Jeżeli pole jest puste – przycisk *zaloguj* jest nieaktywny. Po wypełnieniu pola logowania i kliknięciu *zaloguj* nazwa użytkownika przechowywana jest w *localStorage*. Zalogowany użytkownik ma możliwość dodawania nowych książek do bazy.

Pod listą dostępnych książek znajduje się formularz służący do dodawania nowych książek do bazy danych. Aby możliwe było dodanie książki do bazy, konieczne jest spełnienie warunków:

- pole tytuł nie może być puste
- wybrana data wydania (walidacja w postaci zastosowania odpowiedniego typu pola – *date*)
- wybrany autor (walidacja przez zastosowanie listy dostępnych w bazie autorów – zastosowałem *react-select*. Dzięki temu nie jest możliwe podanie nieistniejącego w bazie autora).
- użytkownik musi być zalogowany w aplikacji

Po spełnieniu wymienionych powyżej warunków przycisk *Dodaj* jest aktywny i jest możliwe dodanie nowej książki do bazy danych.

Po poprawnym dodaniu książki do bazy wyświetlana jest podstrona z informacją.