

Projekt Zaliczeniowy - PAI

Stworzyłem prostą aplikację zarządzającą todo (rzeczami do zrobienia)

Można za jej pomocą:

- zalogować się na istniejące konto lub utworzyć nowe
- dodawać (lub usuwać) nowe todos - każde zawiera opis i planowany termin wykonania
- edytować istniejące todos

Aplikacja jest prosta, ale działa bez zarzutu, ponadto spełnia wszystkie postawione przed nią założenia, a mianowicie:

- ☒ ochronę treści strony przed niezalogowanymi użytkownikami
- ☒ utworzenie tabeli users, w której będą przechowywane dane użytkowników
- ☒ możliwość przeprowadzania operacji na bazie danych (edytowanie todo)
- ☒ odporność na ataki typu SQL injection itd.
- ☒ Pełną walidację danych - zarówno użytkownika jak i todo

Obsługa aplikacji

Aby móc uruchomić aplikację należy najpierw skonfigurować połączenie z bazą danych. Robi się to poprzez plik `application.properties` (`src/main/resources/application.properties`)

Należy skonfigurować następujące parametry:

- `spring.datasource.url`
- `spring.datasource.username`
- `spring.datasource.password`

W przypadku korzystania z xamppa należy:

1. Uruchomić xamppa - server apache i mysql
2. utworzyć nową bazę danych (nazwa według uznania - w moim przypadku "project")
3. ustawić parametr `spring.datasource.url` - wystarczy zmienić ostatni człon linku na nazwę utworzonej przez siebie bazy danych
4. ustawić parametr `spring.datasource.username` na nazwę użytkownika, który będzie komunikował się z bazą danych (domyślny użytkownik to root i to sugeruje zostawić)

5. ustawić parametr `spring.datasource.password` na hasło użytkownika podanego punkt wyżej (jeśli użytkownikiem jest root - hasło jest niepotrzebne - parametr można usunąć)

Po wykonaniu powyższych kroków aplikacja jest gotowa do uruchomienia

Tabele zostaną utworzone automatycznie, ponadto automatycznie zostaną też utworzeni 2 użytkownicy o danych:

| username | email | password | |-----|-----|-----| | student | student@pollub.edu.pl | 1234Pa\$word |
| prowadzocy | prowadzocy@pollub.pl | 1234Pa\$word | Oczywiście możliwe jest dodawanie nowych użytkowników

Numer portu & linki

Aplikacja zostaje uruchomiona na serwerze **tomcat**, na porcie **8080** - to od niego należy rozpoczynać wpisywanie jakichkolwiek adresów

Linki w aplikacji:

AuthController

Endpoint	Metoda	Opis
/index	GET	Obsługuje żądanie strony głównej
/	GET	Przekierowuje do strony głównej
/register	GET	Wyświetla formularz rejestracji użytkownika
/register/save	POST	Obsługuje żądanie rejestracji użytkownika
/login	GET	Wyświetla formularz logowania
/login	POST	obsługuje logowanie użytkownika

ToDoController

Endpoint	Metoda	Opis
/list-todos	GET	Pobiera listę wszystkich zadań
/add-todo	GET	Wyświetla stronę do dodawania nowego zadania
/add-todo	POST	Dodaje nowe zadanie
/delete-todo	GET	Usuwa zadanie o określonym identyfikatorze
/update-todo	GET	Wyświetla stronę aktualizacji zadania
/update-todo	POST	Aktualizuje zadanie