

# Web\_Excel\_filter\_for\_schedule

## Dokumentacja końcowa

### Techniczna

Piotr Karaś, Adam Buczek

#### Plan testów

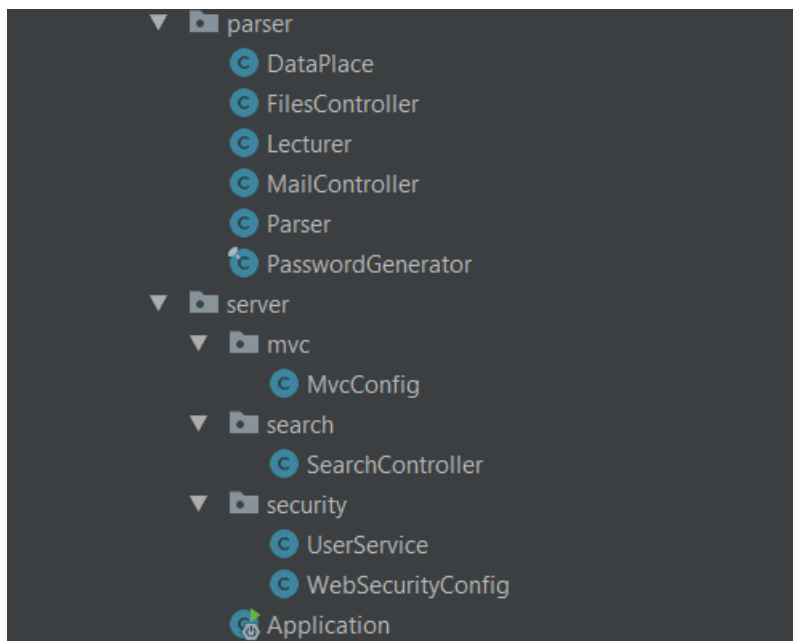
##### Co zostało przetestowane:

- \* wyszukiwanie wykładowcy po nazwisku
- \* wyszukiwanie wykładowcy po nazwisku i imieniu
- \* wyszukiwanie kolumny, w której znajduje się prowadzący
- \* sumowanie komórek znajdujących się w jednej kolumnie (jeden prowadzący)

##### Co może jeszcze zostać przetestowane:

- \* logowanie
- \* resetowanie hasła – wysyłanie e-maila
- \* resetowanie hasła – zmiana hasła za pomocą linku aktywacyjnego
- \* poprawne zapamiętywanie komórek, gdzie są informacje
- \* wczytywanie zestawu zaznaczonych komórek przy wybieraniu interesujących nas informacji
- \* zapisywanie zestawu zaznaczonych komórek przy wybieraniu interesujących nas informacji

#### Rozmieszczenie wykorzystanych klas (src)



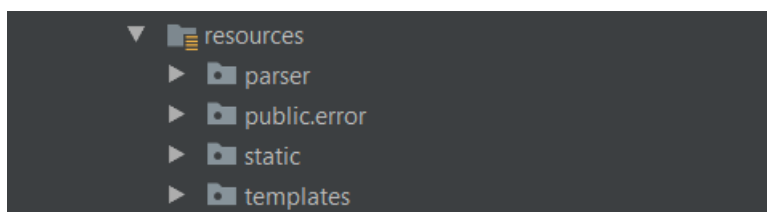
## **Pakiet parser:**

- DataPlace – klasa odpowiedzialna za opis struktury pliku przechowywanego na serwerze. W nim znajdują się pola określające przykładowo: w którym wierszu znajdują się nazwiska prowadzących, gdzie znajdują się przedmioty itd.
- FilesController – zapisywanie i odczytywanie danych z plików. Zawiera zdefiniowane ścieżki np. plik z rozkładem zajęć, z hasłami, plik określający strukturę potrzebną do zdefiniowania klasy DataPlace
- Lecturer – plik określający prowadzącego: imię, nazwisko itd., potrzebny do odbierania i wysyłania JSONa
- MailController – klasa określająca maila na który wysyłany jest link aktywacyjny przy resetowaniu hasła i z jakiego maila on wychodzi. Samo wysyłanie maila robione jest w sposób współbieżny. Tylko ostatni link aktywacyjny jest aktywny i można z niego skorzystać przez godzinę.
- Parser – klasa odpowiedzialna za parsowanie pliku .xlsx, korzysta z klasy DataPlace i biblioteki Apache POI. Tutaj wyszukujemy prowadzących, ich zajęcia, godziny itd.

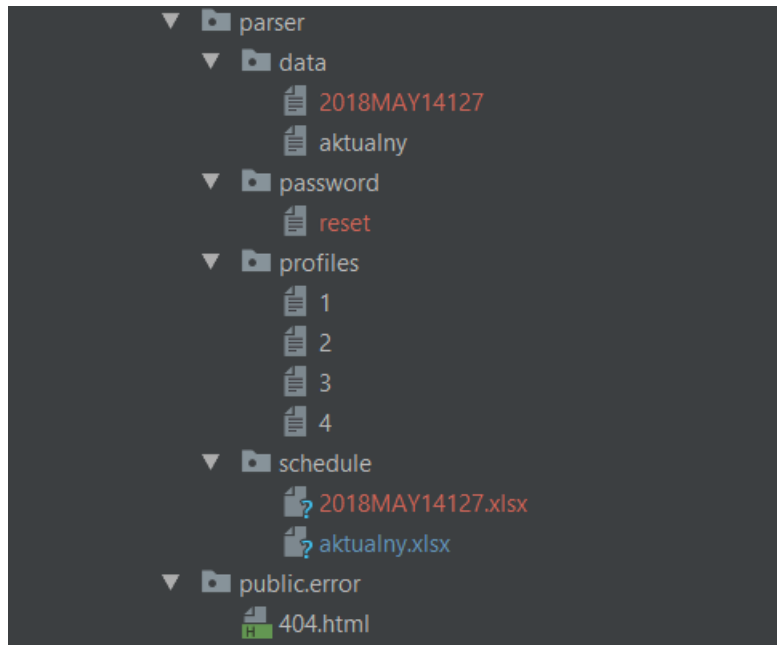
## **Pakiet server:**

- Application – klasa odpowiedzialna za start serwera
- MvcConfig – określenie mapowania i dodania widoków do rejestru.
- SearchController – klasa odpowiedzialna za logikę RESTową aplikacji. Odbiera zapytania i wysyła odpowiedzi. Określa sposób przekazywania informacji. Komunikacja zachodzi za pomocą struktury JSON. Korzysta z klas pakietu 'parser'.
- UserService – metody ładujące użytkownika / admina do serwisu. Odpowiada za zaktualizowanie hasła na serwerze.
- WebSecurityConfig – określenie widoków dostępnych publicznie oraz tych wymagających autoryzacji i konfiguracja danych do logowania

## **Resources:**

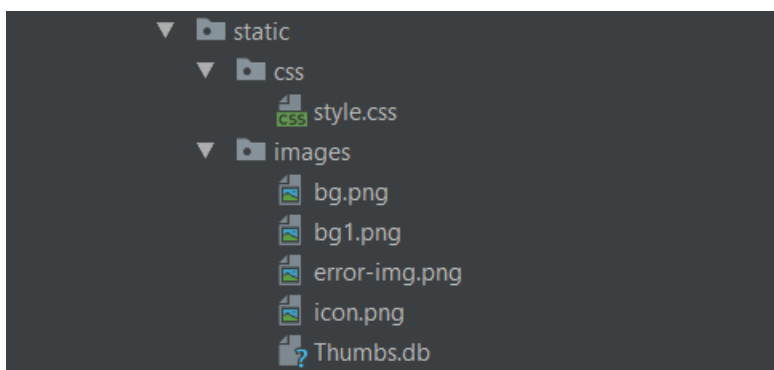


## Parser i public.error



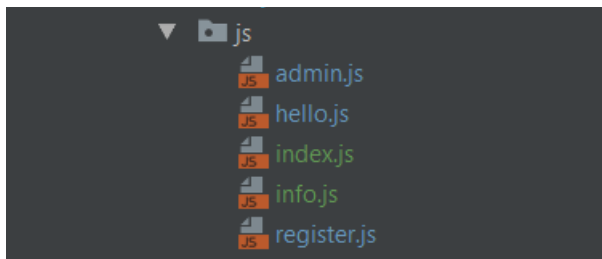
- data – plik z zapisaną strukturą planu, z którego korzysta DataPlace
- password – folder przechowujący pliki z zaszyfrowanym hasłem i linkiem aktywacyjnym
- profiles – pakiet zawierający pliki z zapamiętanymi profilami
- schedule – pliki z rozkładem zajęć oraz plik z zapisaną strukturą planu, z którego korzysta DataPlace
- public.error – przy próbie odwołania się do ścieżki, która nie jest zdefiniowana pokazuje się domyślna strona błędu

## Static



- css – plik zawierający style dla html
- images – obrazki dla stron html

## Static/js

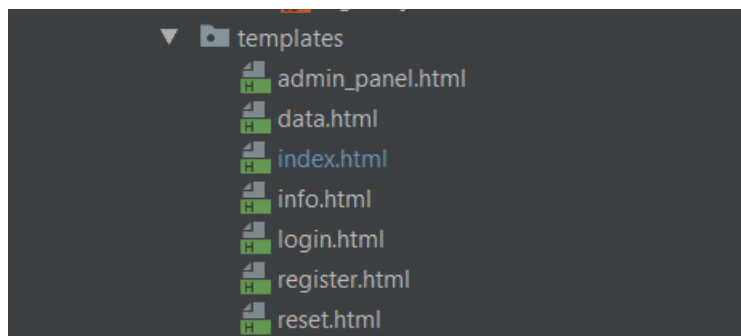


Pliki odpowiedzialne za interakcje interfejsu użytkownika:

- info.js – dla pliku info.html
- index.js – dla pliku index.html
- register.js – dla pliku register.html
- admin.js – dla pliku admin\_panel.html
- hello.js – dla reszty

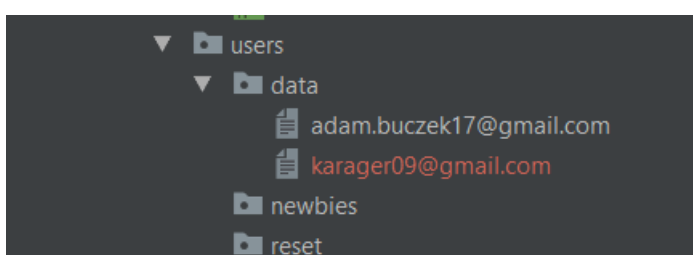
## Templates

Pliki html ze stronami z interfejsem użytkownika



- admin\_panel – definiowanie danych określających strukturę pliku z rozkładem zajęć
- data – wyświetlanie informacji o ilości godzin
- index – strona startowa
- info – formularz z profilami i wyborem interesujących nas informacji
- login – strona z logowaniem, aby uzyskać uprawnienia administratora
- register – strona gdzie podajemy e-mail, żeby się zarejestrować
- reset – strona z resetowaniem hasła, wchodzimy za pomocą linku aktywacyjnego

## Users.data



pliki z zakodowanymi hasłami  
użytkowników i wykorzystywane  
przy resetie