



**AGH**

# **Harmon-Presentation**

**Inżynieria Oprogramowania 2012**

Mikołaj Jakubowski, Krzysztof Romanowski

## 1. Wstęp

Celem projektu było wyprodukowanie aplikacji wspomagającej osoby układające podział godzin na uczelni. Podział godzin nie miał obejmować przydzielania studentów do konkretnych grup, a jedynie rozmieszczać prowadzących wraz z przedmiotami po salach w budynkach. Każdej takiej encji miała być przypisana grupa, reprezentująca maksymalną liczbę studentów mogących uczęszczać na dane zajęcia. Głównym założeniem projektu była możliwość przeciągania zajęć między okienkami przeglądarki.

## 2. Podsumowanie prac

Aplikacja działa w środowisku webowym na przeglądarce Chrome 19. Większość przewidzianych funkcjonalności została zaimplementowana, pomimo tego że projekt był jedynie prezentacją możliwości środowiska w którym aplikacja była tworzona. Brakujące funkcjonalności:

- nie da się przeciągać prowadzących/przedmiotu na rząd w kontrolerze ticket, tworząc na w pół przygotowane zajęcia
- nie widać w jasny sposób konfliktów na podziale godzin
- nie da się w łatwy sposób tworzyć grup dni pomocnych do edycji
- model danych jest luźny pod względem constraintów
- nie ma strony startowej
- nie ma ustawionego połączenia z persystentnej bazy danych
- zajęcia powinno się dać tworzyć uwzględniając 2 pola lub więcej, niekoniecznie wszystkie
- dodać zmienną długość zajęć, w tej chwili każde ma wpisane 90 min
- w widoku kalendarza trudno rozpoznać godziny
- w kalendarzu trudno sobie trafić w dobrą godzinę, przydałby się jakiś snapping
- dodanie kosza, który po przeciągnięciu na niego terminu, usuwa go

## 3. Informacje dla przyszłego programisty

Narzędzia wykorzystane to tworzenia aplikacji:

- SpringToolSuite 3
- Grails 1.3.7
- Plugin grails do STS
- Maven
- Groovy
- Tomcat
- Html5/JS

Przydatne linki:

- <http://www.html5rocks.com/en/tutorials/dnd/basics/>
- <http://grails.org/>
- <http://groovy.codehaus.org/>
- <http://www.infoq.com/minibooks/grails-getting-started>
- <http://grails.org/Maven+Integration>

Ponieważ maven zajmuje się wszystkimi zależnościami nie trzeba przejmować się instalowaniem bibliotek. Zostaną one pobrane przy pierwszym uruchomieniu.

Aplikacje odpala się przez **mvn grails:run-app**.

Całość kodu powstała jako przykład możliwości wykorzystania html5 i przed dalszym rozwojem należy przeprowadzić refactoring(w całym kodzie wymieszana jest nazwa ticket i term, nie ma między nimi znaczącej granicy) lub zacząć pisać projekt od nowa używając jedynie jego fragmentów. Należy też zwrócić się na stronę <http://grails.org/AJAX-Driven+SELECTs+in+GSP> i użyć jej wiedzy.

Klasy domenowe(najlepiej przyglądnąć się kodowi, w tej kwestii jest bardzo jasny):

- Person - posiada imię i nazwisko
- Lecturer - prowadzi wiele zajęć
- Student - jest w wielu grupach
- Subject - jest prowadzony na wielu zajęciach, ma nazwę i min ludzi
- Building - ma wiele pokoi, ma nazwę
- Room - jest w budynku i jest w nim prowadzone wiele zajęć, ma nazwę i max ludzi
- Group - lista studentów, posiada nazwę, bierze udział w wielu zajęciach
- Ticket - jest to czwórka {lecturer, group, room, subject} inaczej zajęcia
- Term - termin - konkretne zajęcia o konkretnej godzinie
- DateGroup - jest to pomocnicza klasa do generowania kalendarza, pozwala personalizować jego widoki

#### 4. Informacje dla przyszłego użytkownika

Aplikacja służy do układanie podziałów godzin. Została przetestowana na przeglądarce Chrome 19. Po otwarciu strony głównej zobaczymy listę podstron na które możemy przejść. Poniżej zostanie opisany standardowy proces personalizacji aplikacji, dodawania zajęć oraz ich modyfikacji.

- a. personalizacja kalendarza (wybieranie widoków)
  - i. otwórz stronę główną
  - ii. przejdź do strony 'Personalizacja kalendarza'
  - iii. lista reprezentuje grupy dat które później da się otworzyć w kalendarzu
  - iv. kliknij na górze 'New DateGroup'
  - v. podaj nazwę danej grupy
  - vi. daty wpisuj w formacie XX.YY gdzie XX to miesiąc a YY to dzień miesiąca
  - vii. kolejne komórki kalendarza oddzielaj ';' (średnikiem)
  - viii. kolejne dni w komórce oddzielaj ',' (przecinkiem)
  - ix. stosując tą metodę dodaj potrzebne widoki jak np. Poniedziałki, Tydzień 1, Cały Semestr, Weekend 1, itp.
- b. w podobny sposób dodaj budynki
- c. następnie pokoje(sale)
- d. prowadzących
- e. przedmioty
- f. grupy

- g. przejdź do widoku 'Zajęcia'
  - i. dodaj kilka przykładowych zajęć, nie przejmuj się jeżeli niektóre podane będą błędne lub jeszcze niewiadome, w takim przypadku wybierz losową opcję
  - ii. w czasie dalszego przebiegu aplikacja pozwoli ci łatwo zmieniać źle wprowadzone dane
  - iii. na tym etapie nie podaje się jeszcze godzin zajęć
- h. otwórz co najmniej 2 okna przeglądarki, w jednym widok Zajęcia, w drugim Kalendarz
  - i. ustaw tak okna aby się jak najmniej zasłaniały
  - ii. widok kalendarza pokazuje grupy dni, których date są opisane w górnej kratce
  - iii. każda komórka to jedna godzina, jest ich 24, zaczynając od 0:00
  - iv. klikając lewym przyciskiem myszy na rząd reprezentujący potrzebne zajęcia, przenieś go, nie puszczając lewego przycisku myszy, na odpowiednią godzinę odpowiedniego dnia
  - v. w kalendarzu pojawi się prostokąt z odpowiednim opisem reprezentujący dany termin
- i. posługiwanie się kalendarzem
  - i. po lewej stronie kalendarza można wybrać dokładne miejsce, prowadzącego i czas w jakim chcemy umieścić zajęcia
  - ii. trzeba na to uważać. Przeciągając zajęcia które na liście są w sali 123, do kalendarza dla sali 456, tak naprawdę stworzymy nowe zajęcia w sali 456 z terminem o wybranej godzinie!
  - iii. to samo dotyczy się prowadzących
  - iv. zmieniając atrybuty po lewej stronie (sala itd) tak naprawdę otwieramy zupełnie różne kalendarze
  - v. można to rozwiązać wybierając kalendarz z pustym atrybutem, wtedy pokażą się terminy dla np. dowolnej sali w wybranym budynku lub dla danej sali i wszystkich prowadzących
- j. modyfikowanie terminów
  - i. każdy termin da się zmodyfikować, można to zrobić na kilka sposobów
    1. wejdź do widoku Zjęcia i zmień atrybuty zajęć odpowiadających konkretnemu terminowi. Pamiętaj że zmienisz wtedy opis wszystkich terminów dla danego zajęcia (użyj gdy chcesz przepisać wszystkie zajęcia jednego prowadzącego do drugiego lub gdy zmienia się układ sal i trzeba przenieść wszystkie zajęcia)
    2. otwórz dwa kalendarze w osobnych oknach przeglądarki i przeciągnij jakiś termin między nimi lub przeciągnij termin między dniami w danym kalendarzu, uważaj żeby nie trafić na inne zajęcia
    3. otwórz kalendarz i jeden z widoków {przedmiot, prowadzący, grupa, sala} i przeciągnij rząd z listy na termin zmieniając jego atrybuty. Nie przejmuj się gdy termin zniknął. Znaczy to

że termin znalazł się w innym kalendarzu.

- ii. terminy usuwa się w widoku Terminy, po kliknięciu na jego numer, wybierz Delete

k. uwagi

- i. kiedy dana kolumna w kalendarzu posiada więcej niż jeden dzień, oznacza to że od każdym zajęciem pokazanym na niej kryje się jedno lub więcej, ale pokazany jest tylko ostatni znaleziony. Oznacza to, że tak kolumna w praktyce pozwala tylko na dodawanie zajęć, żeby je usunąć należy każdy termin wyciągnąć razy ilość dni w danej kolumnie.
- ii. Nie należy przeciągać zajęć na siebie, powoduje to wyżej nieokreślone zachowanie. W najlepszym przypadku, zajęcie które jest celem zostanie usunięte i zastąpione tym przeciąganym.

5. Informacje dla przyszłego administratora

- a. na chwilę obecną aplikacja nie używa żadnej bazy danych. Należy zapewnić jej np. bazę MySQL i wypełnić plik DataSource odpowiednimi danymi w sekcji production. Po deploy aplikacja będzie z niej korzystać.
- b. deploy aplikacji robi się przez **mvn grails:package**
- c. aplikacja wymaga serwera tomcat w wersji ~6
- d. aplikacja wymaga dużej ilości pamięci, należy jej zwiększyć xmax i xperm, do co najmniej 1024m, inaczej serwer będzie rzucał wyjątek PermGen
- e. aplikacja nie jest jeszcze w żaden sposób chroniona, więc należy jej adresu nie upubliczniać