

Krzysztof Dąbrowski

Projekt indywidualny — Raport końcowy

14 czerwca 2019

Spis treści

1. Opis projektu	1
1.1. Wzorzec MVC	1
2. Opis problemu	1
3. Projekt klas	3
4. Prezentacja działania	3
5. Publikacja w chmurze	4
5.1. Podjęte kroki	4
6. Szczegóły techniczne	5
6.1. Schemat działania	5
6.2. Walidacja danych	5
6.3. Zastosowane mechanizmy	5
7. Możliwości rozwoju	6

1. Opis projektu

Celem projektu jest napisanie aplikacji webowej pozwalającej na planowanie semestru studentom korzystającym z systemu *USOS*. W tym celu został wykonany program w języku *C#* w oparciu o framework *ASP.net MVC*. Dzięki temu możliwe było łatwe wydzielenie warstw widoku, modelu i kontrolerów.

Aplikacja została opublikowana w Internecie z wykorzystaniem chmury *Microsoft Azure* co jest dokładniej opisane w punkcie 5.

1.1. Wzorzec MVC

Aby wprowadzić spójność wśród klas aplikacji, wyznaczyć zakres funkcji realizowanych przez dany element oraz sprecyzować zasady komunikacji architektura projektu będzie bazować na wzorcu Model View Controller. Wyraźny podział elementów zaproponowany przez ten wzorzec będzie bezpośrednio realizowany przez podział klas na odpowiadające przestrzenie nazw.

Klasy z przestrzeni *Models* reprezentują automaty zajęcia uczelniane, plan zajęć oraz fragmenty przetwarzanego dokumentu.

Klasy z przestrzeni *Views* są odpowiedzialne za wygląd i wyświetlanie interfejsu programu. Składają się na nie pliki *.shtml*, które są przetwarzane dynamicznie do plików *.html* zwracanych do przeglądarki.

Klasy z przestrzeni *Controllers* wykonują walidacje po stronie serwera. Kontrolują również przejście do kolejnych części aplikacji poprzez przekierowania oraz zwracanie widów.

2. Opis problemu

Celem aplikacji jest pomoc w rozwiązaniu realnego problemu jakim jest wybór odpowiednich grup na zajęcia akademickie przez studentów korzystających z systemu *USOS*.

Na wielu uczelniach nie ma podziału na grupy dziekańskie, dla których jest ogólnie układany plan. Zamiast tego student jest odpowiedzialny za własnoręczne skonfigurowanie planu na przyszłe semestry.

Wygląd przykładowego planu dla kierunku Geografia prowadzonego na Uniwersytecie Warszawskim jest przedstawiony na rysunku 1.

Łatwo zauważyć, że tak przedstawiony plan może być przytłaczający i nieczytelny. Z tego powodu mój projekt pozwala na wczytanie planu po podaniu linku oraz wyświetla plan gdzie widoczne są jednocześnie jedynie zajęcia z wybranych grup.

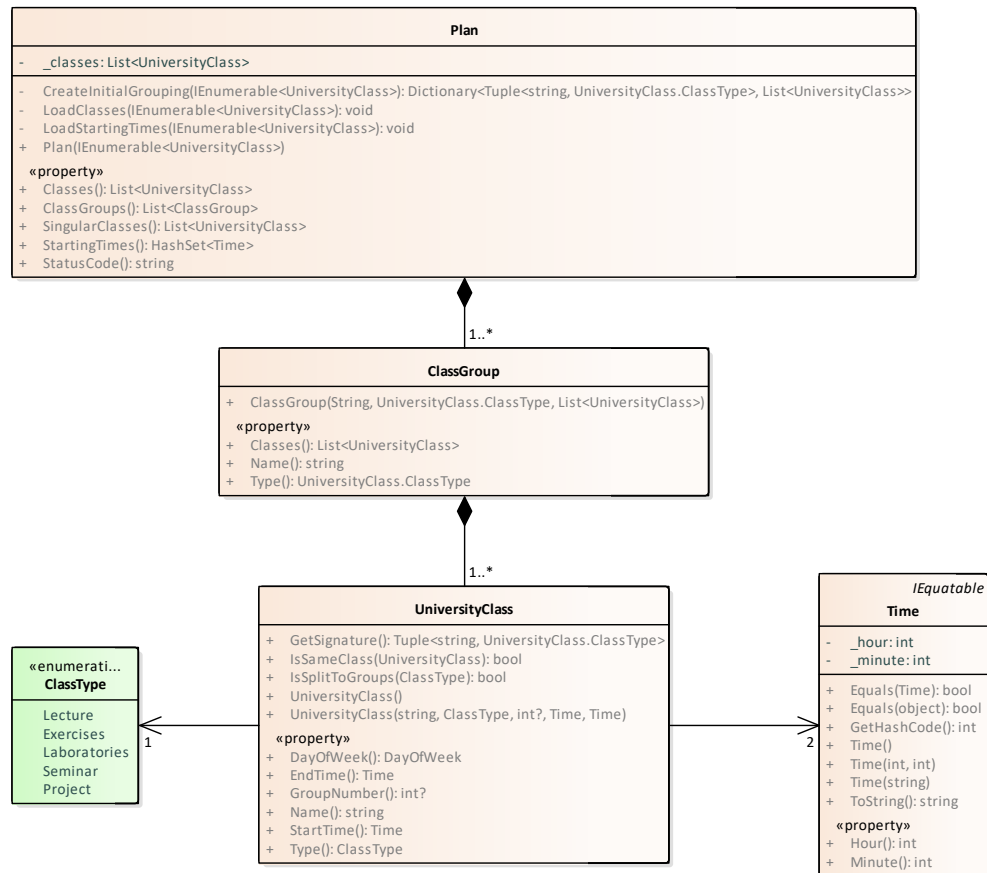
Dzięki temu student może przeprowadzić wizualizację różnych odpowiadających mu konfiguracji planu. Dodatkowo zajęcia różnych typów są zaznaczone innymi kolorami co daje dodatkowe informacje wizualne.

	Poniedziałek	Wtorek	Środa	Czwartek	Piątek
7:00					
8:00		8:00-9:30, CW gr.3 Podstawy geoinformatyki II (208-prac.komp.) - Wojciech Pokojski	8:00-9:30, WYK gr.1 Teledetekcja środowiska (102) - Bogdan Zagajewski		8:00-9:30, WYK gr.1 Współczesne procesy urbanizacji (102) - Sylwia Dudek-Mańkowska
9:00				9:00-11:15, CW gr.1 Zastosowania matematyki i statystyki w geografii (108) - Magdalena Piłkowska, Janusz...	
10:00	9:45-11:15, KON gr.1 Geografia rozwoju (102) - Mirosława Czerny	9:45-11:15, WYK gr.1 Geoeekologia (102) - Wojciech Lewandowski	9:45-11:15, CW gr.4 Podstawy geoinformatyki II (208-prac.komp.) - ... Teledetekcja środowiska (13-prac.komp.) - Bogdan Zagajewski	9:45-11:15, CW gr.3 Geoeekologia (221) - Wojciech Lewandowski	9:45-11:15, CW gr.3 Zastosowania matematyki i statystyki w geografii (108) - ... Biogeografia (301) - Tomasz Grabowski
11:00					9:45-11:15, CW gr.1 Współczesne procesy urbanizacji (115) - Sylwia...
12:00	11:30-13:00, CW gr.1 Podstawy geoinformatyki II (208-prac.komp.) - Wojciech Pokojski	11:30-13:00, CW gr.1 Teledetekcja środowiska (13-prac.komp.) - Bogdan Zagajewski	11:30-13:00, CW gr.5 Podstawy geoinformatyki II (208-prac.komp.) - ... Teledetekcja środowiska (13-prac.komp.) - Bogdan Zagajewski	11:30-13:00, CW gr.4 Zastosowania matematyki i statystyki w geografii (108) - ...	11:30-13:00, CW gr.2 Współczesne procesy urbanizacji (115) - Sylwia... Biogeografia (301) - Tomasz Grabowski
13:00	13:15-14:45, CW gr.2 Podstawy geoinformatyki II (208-prac.komp.) - ... Rozwój regionalny i polityka regionalna (107) - ...	13:15-14:45, CW gr.2 Teledetekcja środowiska (13-prac.komp.) - Bogdan Zagajewski	13:15-14:45, CW gr.6 Teledetekcja środowiska (13-prac.komp.) - Anita Sabat-Tomała	13:15-14:45, CW gr.6 Podstawy geoinformatyki II (208-prac.komp.) - Mariusz Porczek	13:15-14:45, CW gr.5 Współczesne procesy urbanizacji (115) - Mirosław Grochowski
14:00					13:15-14:45, CW gr.5 Zastosowania matematyki i statystyki w geografii (108) - ... Biogeografia (301) - Tomasz Grabowski
15:00	15:00-16:30, CW gr.7 Teledetekcja środowiska (13-prac.komp.) - Anita Sabat-Tomała	15:00-16:30, CW gr.4 Biogeografia (126) - Agnieszka Sosnowska	15:00-16:30, CW gr.5 Teledetekcja środowiska (13-prac.komp.) - Anita Sabat-Tomała	15:00-16:30, WYK gr.1 Biogeografia (102) - Jan Matuszkiewicz	15:00-16:30, CW gr.3 Podstawy geoinformatyki II (208-prac.komp.) - Mariusz Porczek
16:00		15:00-16:30, CW gr.1 Rozwój regionalny i polityka regionalna (107) - ...	15:00-16:30, CW gr.4 Rozwój regionalny i polityka regionalna (107) - ...	15:00-16:30, CW gr.7 Współczesne procesy urbanizacji (115) - Klaudia Peszat	15:00-16:30, WYK gr.1 Rozwój regionalny i polityka regionalna (102) - Katarzyna Podhorodecka
17:00	16:45-18:15, CW gr.8 Teledetekcja środowiska (13-prac.komp.) - Anita Sabat-Tomała	16:45-18:15, CW gr.2 Biogeografia (126) - Agnieszka Sosnowska	16:45-18:15, CW gr.5 Geoeekologia (221) - Tomasz Grabowski	16:45-18:15, CW gr.4 Współczesne procesy urbanizacji (115) - Klaudia Peszat	
18:00		16:45-18:15, CW gr.2 Rozwój regionalny i polityka regionalna (107) - ...	16:45-18:15, CW gr.5 Rozwój regionalny i polityka regionalna (107) - Dorota Rucińska		

Rysunek 1. Modele

3. Projekt klas

Projekt klas reprezentujący model planu jest przedstawia diagram klas zawarty na figurze 2.



Rysunek 2. Modele

4. Prezentacja działania

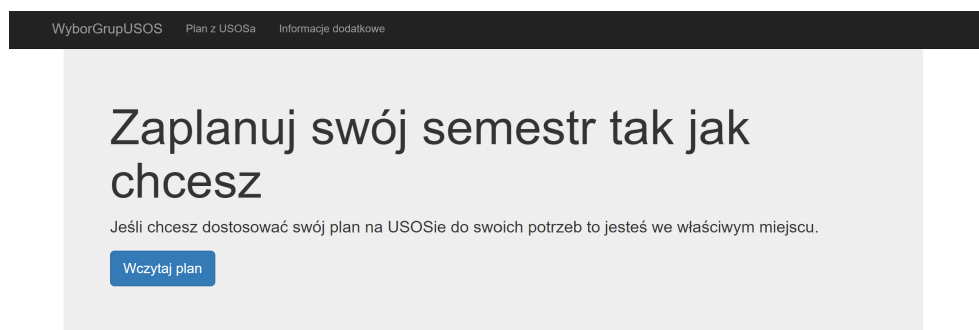
Po przejściu na witrynę główną aplikacji użytkownik zobaczy ekran główny opisujący po krótce aplikację oraz pozwalający na przejście do ekranu wczytania planu. Ekran ten jest widoczny na figurze 3.

Po przejściu dalej zostanie wyświetlony ekran pozwalający na przekazanie aplikacji linku do planu. Ekran zawiera też rozwijaną część opisującą dokładniej szczegóły działania oraz wskazujący przykładowy plan. Możliwe jest wczytanie wskazanego planu lub plany przykładowego. Ekran ten przedstawiają figury 4 i 5.

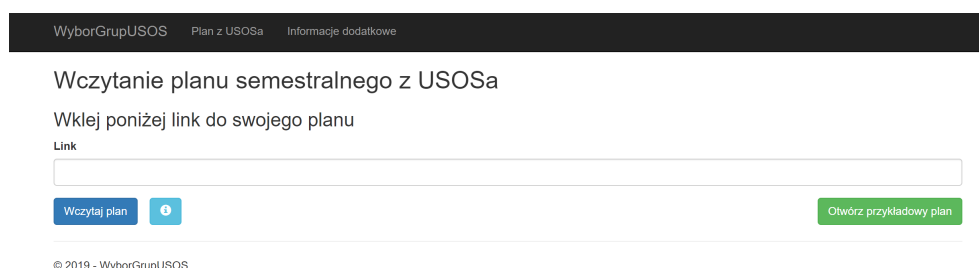
Po wczytaniu planu użytkownik zostanie przekierowany na ekran widoku planu. W górnej jego części znajduje się podgląd aktualnej konfiguracji planu. Na dole wyświetlane są kontrolki pozwalające na zmianę grupy. Przedmioty wyświetlane na planie są pokolorowane zgodnie z ich kategorią.

- żółty – ćwiczenia,
- niebieski – wykład,
- różowy – konwersatorium,
- czerwony – **konflikt**.

Listy wyboru grup zawierają wartości wczytane z planu. W sytuacji gdy więcej niż jedno zajęcie jest ustawione na wybraną godzinę pole zmieni dynamicznie kolor na czerwony.



Rysunek 3. Ekran startowy



Rysunek 4. Ekran wczytania bez dodatkowych informacji

Ekran ten przedstawia figura numer 6.

5. Publikacja w chmurze

W celu udostępnienie aplikacji w Internecie skonfigurowałem infrastrukturę w chmurze *Microsoft Azure*. Dzięki temu aplikacja jest dostępna pod adresem `usos-plan.azurewebsites.net`.

5.1. Podjęte kroki

1. Założyłem subskrypcję studencką przy pomocy konta uczelnianego.
2. Utworzyłem grupę zasobów, powiązaną z subskrypcją, pozwalającą na łączne zarządzanie serwisami.
3. Skonfigurowałem *App Service plan* tak by aplikacja była dostępna 24 godziny na dobę.
4. Utworzyłem *App Service* pozwalający na przekazanie aplikacji webowej do chmury.
5. Wgrałem kod aplikacji do chmury przy pomocy środowiska *Visual Studio*.

Utworzona grupa zasobów jest pokazana na figurze 7

WyborGrupUSOSPlan z USOSaInformacje dodatkowe

Wczytanie planu semestralnego z USOSa

Wklej poniżej link do swojego planu

Link

Wczytaj plan

Otwórz przykładowy plan

O jaki link chodzi

Potrzebny jest odnośnik do planu przedstawiającego rozkład wszystkich przedmiotów i grup na wybrany semestr.

[Przykładowy plan](#)

Gdzie szukać linku do planu?

Odnośnik do planu może być zamieszczony na stronie Twojego wydziału.

© 2019 - WyborGrupUSOS

Rysunek 5. Ekran wczytania z dodatkowymi informacjami

6. Szczegóły techniczne

6.1. Schemat działania

Po wczytaniu linku wprowadzonego przez użytkownika modyfikowana jest część query string tak by otrzymać od serwisu *USOS* cały plan w jednym pliku. Dzięki temu nie trzeba pobierać wielu stron opisujących szczegóły każdego z przedmiotów.

Następnie z odpowiedzi od serwera wyodrębniana jest tabela zawierająca plan. Jest ona parsowana wiersz po wierszu by określić informacje takie jak dzień, w którym organizowane są określone zajęcia.

Po zakończeniu parsowania odpowiedzi serwera tworzony jest model planu zajęć. Na tym etapie wydzielane są grupy zajęć stanowiących wybór oraz zajęcia, na które student musi chodzić i nie ma żadnego wyboru.

Po utworzeniu modelu przekazywany jest on do widoku gdzie zgodnie ze wzorcem MVVM tworzony jest model widoku uwzględniający elementy takie jak aktualnie wygrane zajęcia dla każdej grupy.

Następnie model jest rysowany w postaci tabeli. W reakcji na zmianę grupy przez użytkownika model i wyświetlany widok są aktualizowane.

6.2. Walidacja danych

Lik wprowadzany przez użytkownika jest walidowany po stronie serwera. W tym celu zostały zastosowane adnotacje z przestrzeni nazw `System.ComponentModel.DataAnnotations`.

Sprawdzone jest czy wartość została wprowadzona adnotacją `Required` oraz poprawność linku adnotacją `RegularExpression`. Dzięki zastosowaniu tego rozwiązania mogą być automatycznie generowane odpowiednie komunikaty o błędach.

6.3. Zastosowane mechanizmy

Podczas pisania projektu poznałem wiele mechanizmów, które starałem się zastosować w projekcie.

linq

Zastosowałem bibliotekę `linq` rozszerzającą metody kolekcji w celu filtrowania danych, mapowania oraz wybierania odpowiednich elementów ze zbiorów. Stosowałem za równo składnię `metod` jak i `zapytań`.

Plan

Poniedziałek	Wtorek	Środa	Czwartek	Piątek
	Podstawy geoinformatyki II gr. 3	Teledetekcja środowiska gr. 1	Zastosowania matematyki i statystyki w geografii gr. 1	Współczesne procesy urbanizacji gr. 1
Geografia rozwoju gr. 1	Geoeekologia gr. 1			Biogeografia gr. 3
	Geoeekologia gr. 2			
Rozwój regionalny i polityka regionalna gr. 1		Zastosowania matematyki i statystyki w geografii gr. 1 Biogeografia gr. 1	Współczesne procesy urbanizacji gr. 1	Rozwój regionalny i polityka regionalna gr. 1
Teledetekcja środowiska gr. 4				

Podstawy geoinformatyki II	3
Zastosowania matematyki i statystyki w geografii	1
Teledetekcja środowiska	4
Geoeekologia	2
Współczesne procesy urbanizacji	1
Biogeografia	3
Rozwój regionalny i polityka regionalna	1

© 2019 - WyborGrupUSOS

Rysunek 6. Ekran wczytania z dodatkowymi informacjami

async

Dzięki zastosowaniu programowania współbieżnego mogłem zwolnić aktualnie czekający wątek maszyny podczas wczytywania danych z zewnętrznych źródeł oraz zrównoleglić wykonywanie pracy.

Json.NET

Przy pomocy tej biblioteki firmy *Newtonsoft* mogłem z łatwością prasować i serializować obiekty do formatu JSON. Było to szczególnie przydatne przy konwersji modelu C# na modle w języku JavaScript.

XPath

Ponieważ dokument html jest poniekąd dokumentem xml to mogłem zastosować składnię XPath do wybierania odpowiednich elementów podczas przetwarzania odpowiedzi od serwisu *USOS*.

Animacje CSS

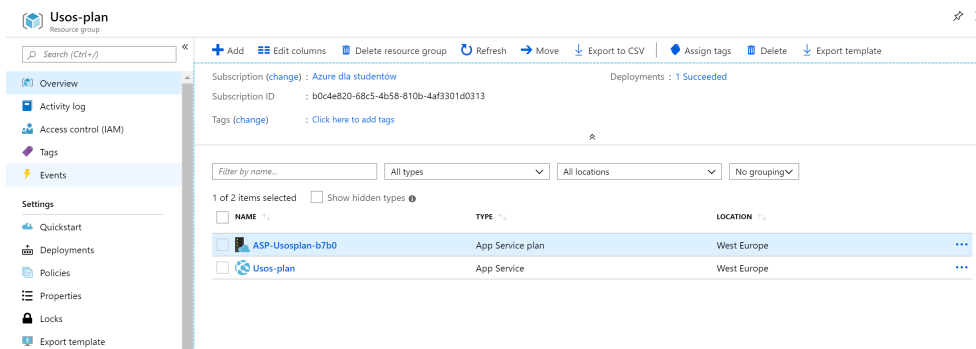
By nadać płynność elementowi ładowania pojawiającemu się podczas ładowania strony zastosowałem animacje opisywane przez język CSS. Opisują one w wektorowy sposób jakie transformacje mają zostać wykonane na obiekcie.

Wiązania danych

Dzięki wiązaniom danych z biblioteki *Knockout.js* można powiązać widok z modelem widoku realizując wzorzec MVVM. Pozwala to na dynamiczną aktualizację powiązanych wartości oraz na łatwe stosowanie wzorca Obserwator.

7. Możliwości rozwoju

W przyszłości do programu możnaby oddać następujące funkcjonalności.



Rysunek 7. Grupa zasobów na Azure

Zapis stanu

Do widoku planu można dodać przycisk pozwalający na zachowanie aktualnego ustawienia planu zajęć. W tym celu należałoby zapisać model widoku w bazie na serwerze. Aby to osiągnąć można zastosować blob storage z platformy *Azure*.

Automatyczne dopasowanie konfiguracji planu

Przydatny dla użytkownika mógłby okazać się algorytm, który automatycznie wybierałby grupy zajęć tak by osiągnąć wskazany przez użytkownika cel. Na przykład minimalizację okienek czy dopasowanie planu do grafiku pracy.

Wstawianie własnych zajęć

Możliwość by użytkownik mógł dodawać własne dodatkowe zajęcia do wczytanego planu.

Usuwanie przedmiotów

Możliwość by użytkownik mógł zrezygnować z wyświetlania zajęć z wybranego przedmiotu.

Konta użytkowników Dzięki systemowi logowań można by powiązać dane takiej jak plany z poprzednich semestrów użytkownika czy zapisane konfiguracje z kontem danej osoby.