**Nazwa przedmiotu:**

Narzędzia typu RAD

**Koordynator przedmiotu:**

Julian MYRCHA

**Status przedmiotu:**

Fakultatywny dowolnego wyboru

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Przedmioty techniczne

**Kod przedmiotu:**

NTR

**Semestr nominalny:**

6 / rok ak. 2018/2019

**Liczba punktów ECTS:**

5

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

120
w tym:
30 godzin wykładu
30 godzin zajęć laboratoryjnych
30 godzin przygotowania do zajęć laboratoryjnych
30 godzin pracy własnej (samokształcenie) w obszarach poruszonych na wykładzie

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

30 godzin wykładu
30 godzin zajęć laboratoryjnych
w sumie 60 godzin co daje 2,5 ECTS

**Język prowadzenia zajęć:**

polski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

30 godzin zajęć laboratoryjnych
30 godzin przygotowania do zajęć laboratoryjnych
w sumie 60 godzin co daje 2,5 ECTS

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład:  | 30h |
| Ćwiczenia:  | 0h |
| Laboratorium:  | 30h |
| Projekt:  | 0h |
| Lekcje komputerowe:  | 0h |

**Wymagania wstępne:**

Wymagana jest podstawowa umiejętność programowania i projektowania aplikacji.

**Limit liczby studentów:**

24

**Cel przedmiotu:**

Celem wykładu jest przedstawienie metod i narzędzi służących do szybkiego tworzenia aplikacji. Zostanie przedstawiona architektura aplikacji opartych na komponentach oraz kilka popularnych frameworków. Studenci zapoznają się z budową własnych komponentów. Zostaną przedstawione metody dostępu do danych w aplikacjach webowych.

**Treści kształcenia:**

Wprowadzenie. Pojęcia podstawowe, przedstawienie istniejących
architektur programów w środowiskach graficznych, cechy wyróżniające
narzędzia RAD.

Język C# jako przykład nowoczesnego języka do tworzenia
aplikacji. Prezentacja tych mechanizmów języka C#, które powstały w
celu wspomagania szybkiego tworzenia aplikacji oraz dostępu do danych.

Technologia ASP.NET. Programowanie aplikacji w technologii
ASP.NET w języku C#.

LINQ oraz Entity Framework. Omówienie sposobów dostępu do baz
danych z aplikacji webowych ASP.Net.

Testowanie. Metody projektowania aplikacji z uwzględnieniem
testów.

MVC.Net jako alternatywna architektura aplikacji webowych.

Python i Django. Przykładowy framework umożliwiający szybkie
tworzenie aplikacji webowych.

Ruby i Rails. Podobne rozwiązanie w języku Ruby.

Zagadnienia omawiane są na przykładzie szeregu technologii firmy
Microsoft (C#, Asp.Net, MVC.Net, SQL Server) oraz (w drugiej części
wykładu) języków Ruby (framework Rails) oraz Python (framework Django).

Zakres laboratorium
Zrealizowanie prostego programu ASP.Net.

Zrealizowanie własnych komponentów ASP.Net.

Realizacja aplikacji w technologii MVC.Net z wykorzystaniem
Entity Framework.

Opracowanie złożonej aplikacji - część 1.

Opracowanie złożonej aplikacji - część 2.

**Metody oceny:**

W trakcie wykładu są 2 laboratoria po 20 pkt, z laboratorium jest do uzyskania 60 pkt. Warunkiem zaliczenia przedmiotu jest uzyskanie co najmniej 51 punktów oraz zaliczenie wszystkich tematów laboratoryjnych.

**Egzamin:**

nie

**Literatura:**

1. Steven Sanderson : Pro ASP.NET MVC 2 Framework, Apress 2010
2. Julia Lerman : Programming Entity Framework, O'Reilly 2010
3. Adam Freeman, Joseph C. Rattz, Jr : Pro LINQ: Language Integrated Query in C# 2010, Apress 2010
4. Tim Patrick : Microsoft ADO.NET 4 Step by Step, Microsoft Press 2010
5. Rob Cameron, Dale Michalk : Pro ASP.NET 3.5 Server Controls and AJAX Components, Apress 2008
6. Jeff McWherter, Ben Hall : Testing ASP.NET Web Applications, Wrox 2010
7. Pierre Henri Kuate, Tobin Harris : NHibernate in Action, Manning 2009

**Witryna www przedmiotu:**

https://usosweb.usos.pw.edu.pl/kontroler.php?\_action=katalog2/przedmioty/pokazPrzedmiot&prz\_kod=103D-INIIT-ISP-NTR

**Uwagi:**

## Charakterystyki przedmiotowe

### Profil ogólnoakademicki - wiedza

**Charakterystyka 01:**

Potrafi samodzielnie zaprojektować i wykonać komponent integrujący się ze środowiskiem typu RAD

Weryfikacja:

sprawdzian 1, Laboratorium 2

**Powiązane charakterystyki kierunkowe:** K\_W09, K\_W16

**Powiązane charakterystyki obszarowe:** I.P6S\_WG

**Charakterystyka 02:**

Potrafi zaprojektować i zrealizować aplikację z wykorzystaniem komponentów

Weryfikacja:

sprawdzian 1, Laboratorium 1

**Powiązane charakterystyki kierunkowe:** K\_W09, K\_W12, K\_W14, K\_W19

**Powiązane charakterystyki obszarowe:** I.P6S\_WG, III.P6S\_WG.o

**Charakterystyka 03:**

Potrafi zaprojektować i wykonać wielodostępną aplikację webową wykorzystującą bazę danych

Weryfikacja:

sprawdzian 2, laboratoria 3, 4 i 5

**Powiązane charakterystyki kierunkowe:** K\_W09, K\_W12, K\_W14, K\_W16, K\_W17, K\_W18, K\_W19

**Powiązane charakterystyki obszarowe:** I.P6S\_WG, III.P6S\_WG.o

**Charakterystyka 04:**

Potrafi opracować metodykę automatycznego testowania aplikacji webowej

Weryfikacja:

sprawdzian 2, laboratorium 6

**Powiązane charakterystyki kierunkowe:** K\_W13, K\_W15

**Powiązane charakterystyki obszarowe:** I.P6S\_WG, III.P6S\_WG.o

### Profil ogólnoakademicki - umiejętności

**Charakterystyka NTR\_U01:**

Umie zaprojektować aplikację typu biznesowego za pomocą nowoczesnych technik

Weryfikacja:

Ocena laboratoriów L5 i L7

**Powiązane charakterystyki kierunkowe:** K\_U13, K\_U15, K\_U16, K\_U20, K\_U21

**Powiązane charakterystyki obszarowe:** I.P6S\_UW, III.P6S\_UW.4.o, III.P6S\_UW.3.o

**Charakterystyka NTR\_U02:**

Umie zaprojektować zestaw testów i wykonać go w sposób automatyczny

Weryfikacja:

Sprawdzenie wyników L8

**Powiązane charakterystyki kierunkowe:** K\_U21, K\_U19, K\_U20

**Powiązane charakterystyki obszarowe:** I.P6S\_UW, III.P6S\_UW.4.o, III.P6S\_UW.3.o