

Programowanie aplikacji internetowych - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Programowanie aplikacji internetowych
Kod przedmiotu	11.3-WF-FizD-PrApl-L-S14_genM2BEX
Wydział	Wydział Fizyki i Astronomii
Kierunek	Fizyka / Fizyka komputerowa
Profil	ogólnoakademicki
Rodzaj studiów	drugiego stopnia z tyt. magistra
Semestr rozpoczęcia	semestr zimowy 2016/2017

Informacje o przedmiocie	
Semestr	3
Liczba punktów ECTS do zdobycia	4
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	

Formy zajęć					
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Wykład	15	1	-	-	Egzamin
Laboratorium	30	2	-	-	Zaliczenie na ocenę

Cel przedmiotu

Celem przedmiotu jest wprowadzenie do programowania aplikacji internetowych zarówno w warstwie prezentacji danych (programowanie po stronie przeglądarki) jak i w warstwie analizy, gromadzenia i przetwarzania danych (programowanie po stronie serwera z wykorzystaniem baz danych). Oprócz rozwijania umiejętności programowania aplikacji internetowych, celem przedmiotu jest również rozwijanie świadomości wolnego oprogramowania i umiejętności wykorzystywania go w tworzeniu profesjonalnych aplikacji internetowych.

Wymagania wstępne

Znajomość podstaw programowanie w języku Python z uwzględnieniem programowania obiektowego, znajomość podstaw relacyjnych baz danych

Zakres tematyczny

1. HTML

- struktura dokumentu
- elementy blokowe i „liniowe”
- prezentacja treści
- odnośniki
- grafika
- listy
- tabele
- formularze
- html 5

2. CSS

- selektory
- formatowanie tekstu, grafiki, tabel, list, linków i formularzy
- model pudełkowy
- pozycjonowanie elementów
- układy stron – tzw. layouts
- menu

3. Framework JQuery

- wstęp do JavaScript (składnia, instrukcje, DOM)
- JQuery – wstęp
- JQuery UI

- Pluginy JQuery
- Ajax

4. Framework Django

- przypomnienie wiadomości o programowaniu w Pythonie
- instalacja i konfiguracja frameworku Django
- widoki i adresy
- szablony html
- modele i praca z bazami danych
- panel administracyjny
- przetwarzanie formularzy

Metody kształcenia

Wykład:

Wykład konwencjonalny, problemowy, pokaz, dyskusja, warsztaty (testowanie aktualnie omawianych fragmentów kodu).

Laboratoria:

Ćwiczenia laboratoryjne, metoda projektu, praca w grupie, giełda pomysłów, burza mózgów, prezentacja, praca z dokumentacją, samodzielne pozyskiwanie wiedzy.

Efekty uczenia się i metody weryfikacji osiągania efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Student potrafi dobrać i skorzystać z gotowych bibliotek, frameworków i rozwiązań bez naruszania cudzej własności intelektualnej i z zachowaniem zapisów licencji. Potrafi wskazać wolne oprogramowanie stanowiące profesjonalną alternatywę dla oprogramowania komercyjnego. Jest świadomy miejsca wolneg oprogramowania w rozwoju aplikacji internetowych i jego znaczenia na rynku pracy	<ul style="list-style-type: none"> • K2_W09 • K2_U09 • K2_U10 • K2_K04 	<ul style="list-style-type: none"> • bieżąca kontrola na zajęciach • dyskusja • projekt 	<ul style="list-style-type: none"> • Wykład • Laboratorium
Student potrafi stworzyć stronę internetową spełniającą standardy html i css zgodnie ze standardami w3c, potrafi oddzielić warstwę danych od warstwy prezentacji i omówić konieczność takiego podziału	<ul style="list-style-type: none"> • K2_U09 • K2_U10 	<ul style="list-style-type: none"> • bieżąca kontrola na zajęciach • dyskusja • egzamin - ustny, opisowy, testowy i inne • projekt • Student potrafi stworzyć stronę internetową spełniającą standardy html i css zgodnie ze standardami w3c, potrafi oddzielić warstwę danych od warstwy prezentacji i omówić konieczność takiego podziału 	<ul style="list-style-type: none"> • Wykład • Laboratorium
Student potrafi zaprojektować bazodanowy serwis internetowy i zakodować go używają frameworku Django. Potrafi omówić rolę bazodanowych serwisów internetowych oraz jest świadomy ich znaczenia na współczesnych rynkach pracy	<ul style="list-style-type: none"> • K2_U09 • K2_U10 • K2_K04 	<ul style="list-style-type: none"> • bieżąca kontrola na zajęciach • dyskusja • egzamin - ustny, opisowy, testowy i inne • projekt 	<ul style="list-style-type: none"> • Wykład • Laboratorium

Warunki zaliczenia

Ocena końcowa jest oceną wykonania oraz omówienia projektu końcowego wybranego z listy zaproponowanych projektów. W projekcie końcowym oceniane jest właściwe użycie HTML +CSS, zastosowanie JQuery (razem stanowiące 40% oceny), zaprojektowanie modeli, widoków i zaimplementowanie serwisu w frameworku Django (40% oceny), omówienie projektu i technik w nim zastosowanych (20% oceny)

Przed przystąpieniem do egzaminu student musi uzyskać zaliczenie z ćwiczeń laboratoryjnych.

Ocena końcowa: średnia ważona ocen z egzaminu (60%) i ćwiczeń (40%).

Literatura podstawowa

Literatura uzupełniająca

Uwagi

Zmodyfikowane przez prof. dr hab. Mirosław Dudek (ostatnia modyfikacja: 30-09-2016 20:57)

Wygenerowano automatycznie z systemu SyllabUZ