



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Aplikacje internetowe [S2ETI1>AI]

Przedmiot

Kierunek studiów

Edukacja techniczno-informatyczna

Rok/Semestr

1/1

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

drugiego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

Liczba godzin

Wykład

30

Laboratorium

30

Inne (np. online)

0

Ćwiczenia

0

Projekty/seminaria

0

Liczba punktów ECTS

4,00

Koordynatorzy

dr inż. Marcin Borowski

marcin.borowski@put.poznan.pl

Wykładowcy

dr inż. Marcin Borowski

marcin.borowski@put.poznan.pl

Wymagania wstępne

W zakresie wiedzy, student powinien posiadać podstawową wiedzę z zakresu programowania strukturalnego oraz obiektowego, podstawowej wiedzy na temat technologii internetowych oraz podstawową wiedzę z zakresu projektowania i używania baz danych. Powinien posiadać umiejętność rozwiązywania podstawowych problemów związanych z procesem projektowania systemów informatycznych oraz umiejętność pozyskiwania informacji ze wskazanych źródeł. W zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.

Cel przedmiotu

1. Przekazanie studentom podstawowej wiedzy dotyczącej projektowania aplikacji internetowych, przepływu informacji w takich aplikacjach, technologii wykorzystywanych przy ich budowie. 2. Rozwijanie u studentów umiejętności rozwiązywania problemów związanych z projektowaniem aplikacji internetowych, wykorzystywania rozwiązań typu Open Source, framework'ów oraz bibliotek wspomagających budowę tego typu rozwiązań. 3. Kształtowanie u studentów umiejętności pracy zespołowej oraz samodzielności w rozwiązywaniu problemów.

Przedmiotowe efekty uczenia się

Wiedza:

1. ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie algorytmów, architektury systemów komputerowych, systemów operacyjnych, technologii sieciowych, języków programowania, grafiki, sztucznej inteligencji, baz danych, wspomagania decyzji, systemów uczących się i inżynierii oprogramowania [k2_w10].
2. ma podbudowaną teoretycznie szczegółową wiedzę związaną z wybranymi zagadnieniami z zakresu informatyki [k2_w11].
3. zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich z wybranego obszaru informatyki i techniki [k2_w07].

Umiejętności:

1. potrafi pozyskiwać informacje z literatury, baz danych oraz innych źródeł (w języku ojczystym i angielskim), integrować je, dokonywać ich interpretacji i krytycznej oceny, wyciągać wnioski oraz formułować i wyczerpująco uzasadniać opinie, w szczególności korzystać z różnego rodzaju dokumentacji technicznych oraz api narzędzi wykorzystywanych w trakcie zajęć praktycznych dotyczących budowy aplikacji internetowych oraz mobilnych [k2_u04].
2. potrafi wykorzystać nabytą wiedzę matematyczną do opisu procesów, tworzenia modeli oraz zapisu algorytmów [k2_u01].
3. potrafi zgodnie z zadaną specyfikacją zaprojektować urządzenie, system informatyczny lub proces używając właściwych metod, technik i narzędzi [k2_u22].

Kompetencje społeczne:

1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - rozumie potrzebę uczenia się przez całe życie. potrafi inspirować i organizować proces uczenia się innych osób [k2_k01].
2. potrafi współdziałać i pracować w grupie, przyjmując w niej różne role w szczególności podczas realizacji projektów informatycznych [k2_k03].
3. potrafi ustalać priorytety służące realizacji określonego przez siebie lub innych zadania w szczególności podczas realizacji projektów informatycznych [k2_k04].

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

1. w zakresie wykładów:
 - a. na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach,
2. w zakresie laboratoriów / ćwiczeń:
 - a. na podstawie oceny bieżącego postępu realizacji zadań,

Ocena podsumowująca:

a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:

1. ocenę wiedzy i umiejętności wykazanych w kolokwium o charakterze problemowym (student może korzystać z dowolnych materiałów dydaktycznych) / w formie testu wyboru, (30 pytań za łączną liczbę 60 punktów (2 punkty za prawidłową odpowiedź), zaliczenie od 60%+1)

• omówienie wyników kolokwium,

b) w zakresie laboratoriów / ćwiczeń weryfikowanie założonych efektów kształcenia realizowane jest przez:

1. ocenę i „obronę” przez studenta sprawozdania z realizacji projektu,

Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:

1. omówienia dodatkowych aspektów zagadnienia,

• efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanego problemu,

1. uwagi związane z udoskonaleniem materiałów dydaktycznych,

wskazywanie trudności percepcyjnych studentów umożliwiające bieżące doskonalenia procesu dydaktycznego.

Treści programowe

Program wykładu obejmuje następujące zagadnienia:

Podstawy technologii XML, DTD, XML Scheme. Strukturę oraz składnię języka HTML5, CSS3. Podstawy języka JavaScript, wykorzystanie biblioteki jQuery. Programowanie w języku PHP. Omówienie

framework'a Yii2 oraz tworzenia aplikacji w oparciu o ten framework.

Ćwiczenia laboratoryjne prowadzone są w formie piętnastu dwugodzinnych zajęć odbywających się w laboratorium komputerowym. Pierwsze zajęcia przeznaczone są na zapoznanie studentów z zasadami użytkowania laboratorium i zaliczenia ćwiczeń. Ćwiczenia realizowane są przez dwuosobowe zespoły studentów. Program zajęć laboratoryjnych obejmuje następujące zagadnienia:

Tworzenie stron w HTML5. Tworzenie arkuszy styli CSS3. Wykorzystywanie bibliotek JavaScript.

Programowanie w języku PHP5. Budowa aplikacji z wykorzystaniem framework'u Yii2.

Metody dydaktyczne

1. wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy, rozwiązywanie zadań, pokaz multimedialny, demonstracja.

2. ćwiczenia laboratoryjne: ćwiczenia praktyczne, dyskusja, praca w zespole, pokaz multimedialny, studium przypadków, demonstracja.

Literatura

Podstawowa

1. jQuery Documentation - <http://api.jquery.com>

2. Yii2 Framework - <http://www.yiiframework.com>

3. Bootstrap Framework – <http://getbootstrap.com>

Uzupełniająca

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	77	4,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	65	2,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwii/egzaminu, wykonanie projektu)	40	1,00