



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Zaawansowane aplikacje internetowe

Przedmiot

Kierunek studiów

Informatyka

Studia w zakresie (specjalność)

Poziom studiów

pierwszego stopnia

Forma studiów

niestacjonarne

Rok/semestr

3/5

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obieralny

Liczba godzin

Wykład

16

Ćwiczenia

Laboratoria

16

Projekty/seminaria

Inne (np. online)

Liczba punktów ECTS

4

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

dr. inż. Maciej Piernik

email: maciej.piernik@cs.put.poznan.pl

tel: (+48 61) 665-30-57

wydział: Wydział Informatyki i Telekomunikacji

adres: ul. Piotrowo 2, 60-965 Poznań

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Paweł Boiński

email: Pawel.Boinski@cs.put.poznan.pl

tel: 61 6652965

wydział: Wydział Informatyki i Telekomunikacji

adres: ul. Piotrowo 3 60-965 Poznań

Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z zakresu protokołów sieciowych, systemów baz danych i programowania zorientowanego obiektowo. Powinien posiadać umiejętność programowania prostych aplikacji z wykorzystaniem zintegrowanych środowisk programistycznych. Powinien również rozumieć konieczność poszerzania swoich kompetencji / mieć gotowość do podjęcia współpracy w ramach zespołu.

Cel przedmiotu

1. Uzyskanie wiedzy dotyczącej podstawowych pojęć i koncepcji z zakresu implementacji dokumentów WWW i aplikacji WWW, niezbędnej do rozróżniania podstawowych architektur aplikacji internetowych oraz metod implementacji ich modułów.



2. Uporządkowanie wiedzy z zakresu architektur sieciowych, protokołów sieciowych, bezpieczeństwa systemów rozproszonych.

3. Nabycie umiejętności implementacji aplikacji WWW z wykorzystaniem podstawowych technologii implementacji interfejsu użytkownika, m.in. HTML, CSS, JavaScript, XML+XSL, technologii implementacji logiki prezentacji, m.in. serwlety Java, JavaServer Pages, PHP, Active Server Pages, technologii implementacji logiki biznesowej, m.in. JavaBeans, biblioteki znaczników JSP.

4. Pozyskanie kompetencji społecznych dotyczących zespołowego przygotowania projektu, w tym organizacji pracy zespołowej, a zwłaszcza przywództwa oraz komunikatywności w procesie grupowego rozwiązywania problemu.

Przedmiotowe efekty uczenia się

Wiedza

ma uporządkowaną wiedzę z zakresu protokołów sieciowych i bezpieczeństwa systemów rozproszonych (K1st_W4)

ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie architektur aplikacji WWW wykorzystujących bazy danych (K1st_W4)

zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu projektowania, implementacji i wdrażania aplikacji WWW (K1st_W7)

Umiejętności

ma umiejętność formułowania algorytmów i ich programowania w językach stosowanych dla aplikacji internetowych z użyciem przynajmniej jednego z popularnych narzędzi (K1st_U11)

potrafi zgodnie z zadaną specyfikacją zaprojektować oraz zrealizować aplikację internetową, używając właściwych metod, technik i narzędzi (K1st_U10)

ma umiejętność tworzenia prostych aplikacji internetowych, wykorzystujących systemy baz danych oraz interakcyjne interfejsy użytkownika (K1st_U12)

potrafi wybrać technologię implementacji aplikacji WWW odpowiednią do zadanego obszaru zastosowań (K1st_U18)

potrafi zaprojektować dobry interfejs użytkownika dla różnych klas systemów informatycznych, w tym aplikacji internetowych (K1st_U14)

Kompetencje społeczne

ma świadomość znaczenia wiedzy inżynierskiej w rozwiązywaniu problemów inżynierskich, zna przykłady wadliwie działających systemów oraz jest w stanie wskazać przyczyny tych wad (K1st_K2)

potrafi uczestniczyć w zespołowym przygotowaniu projektu, w tym zaplanować organizację pracy zespołowej oraz komunikować się w procesie grupowego rozwiązywania problemu (K1st_K1)



potrafi myśleć w sposób przedsiębiorczy, znajdując komercyjne zastosowanie dla tworzonych oprogramowania (K1st_K3)

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

- w zakresie wykładów: - na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach;
- w zakresie laboratorium: - na podstawie oceny bieżącego postępu realizacji zadań.

Ocena podsumowująca:

Sprawdzanie założonych efektów kształcenia realizowane jest przez:

- w zakresie laboratorium - ocenę i obronę zadań wykonanych przez studenta w trakcie semestru oraz projektu końcowego
- w zakresie wykładu - ocenę wiedzy i umiejętności wykazanych na zaliczeniu pisemnym o formie testu.

Treści programowe

1. Różne podejścia do zagadnienia logiki prezentacji na różnych platformach takich jak ASP.NET i Java EE
2. Sposoby implementacji logiki biznesowej umożliwiającej obsłużenie wielu typów aplikacji i interfejsów na najpopularniejszych platformach
3. Projektowanie aplikacji opartych o dane; organizacja warstwy dostępu do danych w sposób umożliwiający jej ponowne wykorzystanie, skalowalność, wydajność i bezpieczeństwo; przykłady na różnych platformach
4. Infrastruktura aplikacji www; najpopularniejsze architektury
5. Servery HTTP
6. Zaawansowane zagadnienia interfejsu użytkownika: preprocesory CSS, responsive web design, frameworki front-end'owe
7. Tworzenia aplikacji typu Single Page Application wykorzystując najpopularniejsze frameworki, zaawansowane zagadnienia JavaScript, przetwarzanie asynchroniczne; projektowanie i implementacja usług sieciowych typu RESTful
8. Uwierzytelnianie i autoryzacja w aplikacjach webowych; najważniejsze ataki oraz jak się przed nimi bronić
9. Testowanie aplikacji webowych oraz ich wydajności

Cześć wymienionych wyżej treści programowych realizowana jest w ramach pracy własnej studenta.

Metody dydaktyczne



1. wykład: prezentacja multimedialna, prezentacja ilustrowana przykładami podawanymi na tablicy, rozwiązywanie zadań, demonstracja.
2. ćwiczenia laboratoryjne: rozwiązywanie zadań, ćwiczenia praktyczne, warsztaty

Literatura

Podstawowa

1. J. Duckett, Web Design with HTML, CSS, JavaScript and jQuery, Wiley, 2014.
2. B. Sholtz, A. Tijms, The Definitive Guide to JSF in Java EE 8: Building Web Applications with JavaServer Faces, Apress, 2018.
3. K. Hadlock, Ajax for Web Application Developers, Sams Publishing, 2006.
4. J. Liberty, D. Hurwitz, B. MacDonald, Learning ASP.NET 2.0 with AJAX: A Practical Hands-on Guide, O'Reilly, 2007.

Uzupełniająca

1. E. Jendrock, I. Evans, D. Gollapudi, K. Haase, C. Srivathsa, The Java EE 6 Tutorial, Oracle, 2010.

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	98	4,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	34	1,5
Praca własna studenta (przygotowanie do ćwiczeń laboratoryjnych, napisanie programów, uruchomienie, weryfikacja, przygotowanie do zaliczenia wykładów i zadań na laboratoriach, przygotowanie do oddawania projektu) ¹	64	2,5

¹ niepotrzebne skreślić lub dopisać inne czynności