

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Projektowanie i konstrukcja systemów rozproszonych 1		Kod 1010515311010511656
Kierunek studiów Informatyka	Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki	Rok / Semestr 1 / 1
Ścieżka obieralności/specjalność Sieci komputerowe	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obligatoryjny
Stopień studiów: II stopień	Forma studiów (stacjonarna/niestacjonarna) niestacjonarna	
Godziny Wykłady: 16 Ćwiczenia: - Laboratoria: 16 Projekty/seminaria: -		Liczba punktów 3
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) kierunkowy		(ogólnouczelniany, z innego kierunku) z danego kierunku
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne		Podział ECTS (liczba i %) 3 100%
Odpowiedzialny za przedmiot / wykładowca:		
<p>dr inż. Cezary Sobaniec email: Cezary.Sobaniec@put.poznan.pl tel. 61 6652370 Instytut Informatyki ul. Piotrowo 2, 60-965 Poznań</p>		
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę w zakresie: systemów operacyjnych, technologii sieciowych, przetwarzania rozproszonego, bezpieczeństwa systemów informatycznych oraz baz danych.
2	Umiejętności:	Powinien posiadać umiejętność pozyskiwania informacji z literatury oraz innych źródeł (w języku ojczystym i angielskim); powinien potrafić wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych metody analityczne, symulacyjne oraz eksperymentalne; powinien potrafić integrować wiedzę z różnych obszarów informatyki (a w razie potrzeby także wiedzę z innych dyscyplin naukowych) oraz zastosować podejście systemowe, uwzględniające także aspekty pozatechniczne.
3	Kompetencje społeczne	Student powinien również rozumieć konieczność poszerzania swoich kompetencji. Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.
Cel przedmiotu:		
<p>1. Przekazanie studentom podstawowej wiedzy z zakresu konstrukcji nowoczesnych i skalowalnych systemów rozproszonych w architekturze zorientowanej na usługi (SOA) z wykorzystaniem usług sieciowych Web Services i REST. Jednocześnie w ramach przedmiotu przekazywana jest podstawowa wiedza z zakresu zarządzania projektem (m.in. metodyka PRINCE2) oraz norm ISO9000.</p> <p>2. Rozwijanie u studentów umiejętności rozwiązywania problemów związanych z projektowaniem i implementacją usług sieciowych oraz efektywnym zarządzaniem projektem.</p> <p>3. Kształtowanie u studentów umiejętności pracy zespołowej.</p>		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		

1. ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie algorytmów i złożoności, architektury systemów komputerowych, systemów operacyjnych, technologii sieciowych, baz danych; - [K_W4]
2. ma podbudowaną teoretycznie szczegółową wiedzę związaną z wybranymi zagadnieniami z zakresu informatyki, takimi jak: usługi katalogowe, zarządzanie oprogramowaniem, archiwizacja i odtwarzanie, rozproszone systemy plików, systemy klastrowe - [K_W5]
3. ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach w informatyce i w wybranych pokrewnych dyscyplinach naukowych, - [K_W6]
4. ma podstawową wiedzę o cyklu życia systemów informatycznych sprzętowych lub programowych - [K_W7]
5. zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich z wybranego obszaru informatyki, w szczególności obejmujące zarządzanie systemami uniksowymi, monitorowanie usług systemowych, diagnostykę sieci komputerowych - [K_W8]
6. ma podstawową wiedzę dotyczącą zarządzania jakością, w tym podstawową wiedzę nt. standardów serii ISO 9000 - [K_W12]

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 - [K_U1]
2. potrafi określić kierunki dalszego uczenia się i zrealizować proces samokształcenia - [K_U5]
3. potrafi posługiwać się technikami informacyjno-komunikacyjnymi wykorzystywanymi przy realizacji przedsięwzięć informatycznych - [K_U7]
4. potrafi wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych metody analityczne oraz eksperymentalne - [K_U9]
5. potrafi - przy formułowaniu i rozwiązywaniu zadań inżynierskich - integrować wiedzę z różnych obszarów informatyki (a w razie potrzeby także wiedzę z innych dyscyplin naukowych) oraz zastosować podejście systemowe, uwzględniające także aspekty pozatechniczne - [K_U10]
6. potrafi przeprowadzić analizę ryzyka związanego z przedsięwzięciem informatycznym - [K_U11]
7. potrafi formułować i testować hipotezy związane z problemami inżynierskimi i prostymi problemami badawczymi - [K_U12]
8. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych - [K_U13]
9. potrafi poprawnie użyć przynajmniej jedną metodę szacowania pracochłonności wytwarzania oprogramowania - [K_U15]
10. potrafi efektywnie uczestniczyć w inspekcji oprogramowania - [K_U19]
11. ma umiejętność systematycznego przeprowadzania testów funkcjonalnych - [K_U20]
12. potrafi zaproponować ulepszenia (usprawnienia) istniejących rozwiązań technicznych - [K_U21]
13. potrafi sformułować wymagania pozafunkcyjne dla wybranych charakterystyk jakościowych - [K_U23]
14. potrafi ocenić przydatność metod i narzędzi służących do rozwiązania zadania inżynierskiego, polegającego na budowie lub ocenie systemu informatycznego lub jego składowych, w tym dostrzec ograniczenia tych metod i narzędzi - [K_U24]
15. potrafi - zgodnie z zadaną specyfikacją, uwzględniającą aspekty pozatechniczne - zaprojektować złożone urządzenie, system informatyczny lub proces oraz zrealizować ten projekt - co najmniej w części - używając właściwych metod, technik i narzędzi, w tym przystosowując do tego celu istniejące lub opracowując nowe narzędzia - [K_U27]

Kompetencje społeczne:

1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K_K1]
2. potrafi inspirować i organizować proces uczenia się innych osób - [K_K2]
3. zna przykłady i rozumie przyczyny wadliwie działających systemów informatycznych, które doprowadziły do poważnych strat finansowych, społecznych lub też do poważnej utraty zdrowia, a nawet życie - [K_K4]
4. potrafi współdziałać i pracować w grupie, przyjmując w niej różne role - [K_K5]
5. potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania - [K_K6]
6. prawidłowo identyfikuje i rozstrzyga dylematy związane z wykonywaniem zawodu - [K_K7]
7. potrafi myśleć i działać w sposób przedsiębiorczy - [K_K8]

Sposoby sprawdzenia efektów kształcenia

<p>Ocena formująca:</p> <p>a) w zakresie wykładów:</p> <ul style="list-style-type: none">- na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach, <p>b) w zakresie laboratoriów / ćwiczeń:</p> <ul style="list-style-type: none">- na podstawie oceny bieżącego postępu realizacji zadań, <p>Ocena podsumowująca:</p> <p>a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:</p> <ul style="list-style-type: none">- ocenę wiedzy i umiejętności wykazanych na kolokwium pisemnym o charakterze problemowym (5 pytań otwartych, każde dające 1 pkt., zaliczenie od 50%)- omówienie wyników kolokwium, <p>b) w zakresie laboratoriów / ćwiczeń weryfikowanie założonych efektów kształcenia realizowane jest przez:</p> <ul style="list-style-type: none">- ocenę i obronę przez studenta sprawozdania z realizacji dwóch projektów,- ocenianie ciągle, na każdych zajęciach stopnia realizacji ćwiczeń laboratoryjnych, <p>Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:</p> <ul style="list-style-type: none">- przygotowanie i poprowadzenie na forum grupy warsztatów prezentujących uzgodnioną wcześniej usługę lub środowisko,- umiejętność współpracy w ramach zespołu praktycznie realizującego zadanie szczegółowe w laboratorium,- uwagi związane z udoskonaleniem materiałów dydaktycznych,- wskazywanie trudności percepcyjnych studentów umożliwiające bieżące doskonalenia procesu dydaktycznego.
Treści programowe
<p>Program wykładu obejmuje następujące zagadnienia:</p> <ol style="list-style-type: none">1. Języki znacznikowe: HTML4, XHTML, HTML5, ewolucja języków znacznikowych, geneza HTML5, Web APIs, obiekt Canvas, WebGL2. Standard XML, aplikacje języka XML (MathML, DocBook, SVG, Open Document Format, Office Open XML), XML Information Set, prezentacja dokumentów XML, standard Extensible Stylesheet Language, XSL Formatting Objects, XHTML3. Protokół HTTP: żądania i odpowiedzi, uwierzytelnianie, buforowanie podręczne.4. Technologie dynamiczne po stronie serwera: CGI, FastCGI, moduły wbudowane w serwer, SSI, PHP, ASP, Java EE, .Net5. Asynchroniczna komunikacja z serwerem: AJAX, WebSocket6. Architektura SOA: definicja usługi, motywacja dla SOA, definicja architektury, założenia SOA, magistrala ESB, usługi sieciowe Web Services: przegląd standardów WS-*, protokół SOAP, format komunikatów SOAP, wiązanie SOAP z protokołami transportowymi, standard opisu usług WSDL, profile WS-I (Basic, Security),7. Usługi sieciowe: usługi sieciowe REST: problem adresacji usług sieciowych, styl architektoniczny REST, zasoby i ich reprezentacje, metody i kody błędów protokołu HTTP, projektowanie usług. <p>Zajęcia laboratoryjne prowadzone są w formie 2-godzinnych ćwiczeń, odbywających się w laboratorium. Ćwiczenia realizowane są indywidualnie lub w zespołach 2 osobowych w zależności od charakteru ćwiczeń. Program laboratorium obejmuje następujące zagadnienia:</p> <ol style="list-style-type: none">1. Warstwa prezentacyjna: HTML4, HTML5, CSS2/3, JavaScript2. Protokół HTTP: analiza i generowanie żądań i odpowiedzi, uwierzytelnianie, przekierowania, negocjacja treści, ciasteczka, serwery pośredniczące3. Skrypty CGI: konfiguracja, tworzenie dynamicznych odpowiedzi, dostęp do baz danych, języki Perl, Python, budowa prostej aplikacji4. PHP: elementy języka programowania: tryby pracy, instrukcje sterujące, współpraca z serwerem WWW, dostęp do baz danych, programowanie obiektowe, przetwarzanie formularzy, mechanizm sesji, projekt aplikacji webowej5. XML: poprawność formatowania i poprawność dokumentów, DTD, projekt aplikacji języka XML, prezentacja dokumentów XML6. XSLT: transformacje dokumentów do HTML offline i w przeglądarce, łączenie ze stylem kaskadowym, tworzenie odnośników <p>Metody dydaktyczne:</p> <ol style="list-style-type: none">1. wykład: prezentacja multimedialna, demonstracja oprogramowania2. ćwiczenia laboratoryjne: opracowywanie niewielkich przykładowych programów testujących poznawane mechanizmy, eksperymenty funkcjonalne, dyskusja, pokaz multimedialny, studium przypadków, demonstracja oprogramowania
<p>Literatura podstawowa:</p> <ol style="list-style-type: none">1. Christopher Schmitt, Kyle Simpson. HTML5 Cookbook. O'Reilly Media, 2011.2. Leonard Richardson, Sam Ruby. RESTful Web Services. O'Reilly Media 2007.

Literatura uzupełniająca:		
1. Michael Rosen, Boris Lublinsky, Kevin T. Smith, Marc J. Balcer. Applied SOA Service-Oriented Architecture and Design Strategies. Wiley, 2012.		
2. Jim Webber, Savas Parastatidis, Ian Robinson. REST in Practice Hypermedia and Systems Architecture. O'Reilly Media, 2010.		
Bilans nakładu pracy przeciętnego studenta		
Czynność	Czas (godz.)	
1. udział w zajęciach laboratoryjnych:	16	
2. przygotowanie do ćwiczeń laboratoryjnych:	8	
3. udział w konsultacjach związanych z realizacją procesu kształcenia, w szczególności ćwiczeń laboratoryjnych / projektu	4	
4. udział w wykładach	16	
5. napisanie programu / programów, uruchomienie i weryfikacja (czas poza zajęciami laboratoryjnymi)	8	
6. zapoznanie się ze wskazaną literaturą / materiałami dydaktycznymi (10 stron tekstu naukowego = 1 godz.), 100 stron	10	
7. przygotowanie do kolokwium i obecność na kolokwium: 12 godz. + 1 godz.	13	
Obciążenie pracą studenta		
forma aktywności	godzin	ECTS
Łączny nakład pracy	75	3
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	37	1
Zajęcia o charakterze praktycznym	32	1