

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: - Projekty/seminaria: 16		Liczba punktów 4
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) (ogólnouczelniany, z innego kierunku) kierunkowy z danego kierunku		
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne nauki techniczne		Podział ECTS (liczba i %) 4 100% 4 100%
Odpowiedzialny za przedmiot / wykładowca: dr inż. Cezary Sobaniec email: Cezary.Sobaniec@put.poznan.pl tel. 61 6652370 Instytut Informatyki ul. Piotrowo 2, 60-965 Poznań		
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: 1. Przekazanie podstawowej wiedzy z zakresu konstrukcji nowoczesnych i skalowalnych systemów rozproszonych korzystających z technologii webowych. 2. Przekazanie podstawowej wiedzy o założeniach architektury zorientowanej na usługi (SOA) oraz o technologiach związanych z konstrukcją usług sieciowych w modelach Web Services i REST. 3. Rozwijanie u studentów umiejętności rozwiązywania problemów związanych z projektowaniem i implementacją usług sieciowych oraz efektywnym zarządzaniem projektem.		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza: 1. ma zaawansowaną i pogłębioną wiedzę z zakresu technologii i standardów używanych do budowy webowych systemów informatycznych - [K2st_W1] 2. ma zaawansowaną wiedzę szczegółową dotyczącą konstrukcji usług sieciowych w modelu REST - [K2st_W3] 3. ma zaawansowaną i szczegółową wiedzę o procesach zachodzących w cyklu życia systemów informatycznych - [K2st_W5] 4. zna zaawansowane metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich - [K2st_W6]		
Umiejętności:		

1. 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 - [K2st_U5]
2. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych - [K2st_U6]
3. potrafi szacować pracochłonności wytwarzania oprogramowania - [K2st_U6]
4. potrafi dokonać krytycznej analizy istniejących rozwiązań technicznych oraz zaproponować ich ulepszenia (usprawnienia) - [K2st_U8]
5. 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; - [K2st_U9]
6. 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 - [K2st_U11]
7. potrafi współdziałać w zespole, przyjmując w nim różne role - [K2st_U15]

Kompetencje społeczne:

1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K2st_K1]
2. rozumie znaczenie wykorzystywania najnowszej wiedzy z zakresu informatyki w rozwiązywaniu problemów badawczych i praktycznych - [K2st_K2]

Sposoby sprawdzenia efektów kształcenia

Ocena formująca:

a) w zakresie wykładów:

- na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach,

b) w zakresie laboratoriów / ćwiczeń:

- na podstawie oceny bieżącego postępu realizacji zadań,

Ocena podsumowująca:

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

- ocenę wiedzy i umiejętności wykazanych na kolokwium pisemnym o charakterze problemowym (5 pytań otwartych, każde dające 1 pkt., zaliczenie od 50%)

- omówienie wyników kolokwium,

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

- ocenę i obronę przez studenta sprawozdania z realizacji kilku projektów,

- ocenianie ciągle, na każdych zajęciach stopnia realizacji ćwiczeń laboratoryjnych

Treści programowe

Program wykładu obejmuje następujące zagadnienia:

1. Języki znacznikowe: HTML4, XHTML, HTML5, ewolucja języków znacznikowych, geneza HTML5, Web APIs, obiekt Canvas, WebGL
2. 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, XHTML
3. Protokół HTTP: żądania i odpowiedzi, uwierzytelnianie, buforowanie podręczne.
4. Technologie dynamiczne po stronie serwera: CGI, FastCGI, moduły wbudowane w serwer, SSI, PHP
5. Asynchroniczna komunikacja z serwerem: AJAX, WebSocket
6. 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.

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:

1. Warstwa prezentacyjna: HTML4, HTML5, CSS2/3, JavaScript
2. Protokół HTTP: analiza i generowanie żądań i odpowiedzi, uwierzytelnianie, przekierowania, negocjacja treści, ciasteczka, serwery pośredniczące
3. Skrypty CGI: konfiguracja, tworzenie dynamicznych odpowiedzi, dostęp do baz danych, języki Perl, Python, budowa prostej aplikacji
4. Analiza funkcjonowania istniejącej usługi sieciowej REST.
5. Projekt i implementacja własnej usługi sieciowej REST.
6. XML: poprawność formatowania i poprawność dokumentów, DTD, projekt aplikacji języka XML, prezentacja dokumentów XML

Metody dydaktyczne:

1. wykład: prezentacja multimedialna, demonstracja oprogramowania
2. ćwiczenia laboratoryjne: opracowywanie niewielkich przykładowych programów testujących poznawane mechanizmy, eksperymenty funkcjonalne, dyskusja, pokaz multimedialny, studium przypadków, demonstracja oprogramowania

Literatura podstawowa:

Literatura uzupełniająca:

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)	10	
6. zapoznanie się ze wskazaną literaturą / materiałami dydaktycznymi (10 stron tekstu naukowego = 1 godz.), 100 stron	13	
7. przygotowanie do kolokwium i obecność na kolokwium: 12 godz. + 1 godz.		
Obciążenie pracą studenta		
forma aktywności	godzin	ECTS
Łączny nakład pracy	83	4
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	37	2
Zajęcia o charakterze praktycznym	40	2