

<b>KARTA OPISU MODUŁU KSZTAŁCENIA</b>		
Nazwa modułu/przedmiotu <b>Projektowanie i konstrukcja systemów rozproszonych 2</b>		Kod <b>1010515321010511657</b>
Kierunek studiów <b>Informatyka</b>	Profil kształcenia (ogólnoakademicki, praktyczny) <b>ogólnoakademicki</b>	Rok / Semestr <b>1 / 2</b>
Ścieżka obieralności/specjalność <b>Sieci komputerowe</b>	Przedmiot oferowany w języku: <b>polski</b>	Kurs (obligatoryjny/obieralny) <b>obieralny</b>
Stopień studiów: <b>II stopień</b>	Forma studiów (stacjonarna/niestacjonarna) <b>niestacjonarna</b>	
Godziny Wykłady: <b>8</b> Ćwiczenia: - Laboratoria: <b>24</b> Projekty/seminaria: -		Liczba punktów <b>4</b>
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) <b>kierunkowy</b>		(ogólnouczelniany, z innego kierunku) <b>z danego kierunku</b>
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki <b>nauki techniczne</b>		Podział ECTS (liczba i %) <b>4 100%</b>
<b>Odpowiedzialny za przedmiot / wykładowca:</b>		
<p>dr inż. Łukasz Piątkowski                      email: Lukasz.Piatkowski@put.poznan.pl                      tel. 61 6652950                      Instytut Informatyki                      ul. Piotrowo 2, 60-965 Poznań</p>		
<b>Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:</b>		
1	<b>Wiedza:</b>	Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę w zakresie: systemów operacyjnych, technologii sieciowych, przetwarzania rozproszonego, bezpieczeństwa systemów informatycznych oraz baz danych.
2	<b>Umiejętności:</b>	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. Student musi posiadać podstawowe umiejętności związane z tworzeniem oprogramowania obiektowego dla środowisk zarządzanych.
3	<b>Kompetencje społeczne</b>	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 oraz umiejętność treściwej prezentacji indywidualnie zdobytej wiedzy i doświadczenia w formie technicznych prezentacji multimedialnych.
<b>Cel przedmiotu:</b>		
<ol style="list-style-type: none"> <li>Przekazanie studentom podstawowej wiedzy z zakresu konstrukcji nowoczesnych i skalowalnych systemów rozproszonych w architekturze zorientowanej na usługi (SOA).</li> <li>Rozwijanie u studentów umiejętności rozwiązywania problemów związanych z projektowaniem i implementacją usług sieciowych oraz efektywnym zarządzaniem projektem.</li> <li>Kształtowanie u studentów umiejętności pracy zespołowej.</li> <li>Przekazanie studentom podstawowej wiedzy dotyczącej symulacyjnej oceny systemów rozproszonych.</li> <li>Analiza właściwości oprogramowania typu distributed middleware w oparciu o istniejące rozwiązania.</li> </ol>		
<b>Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia</b>		
<b>Wiedza:</b>		

<p>1. ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie architektury systemów komputerowych, technologii sieciowych, baz danych; - [K_W4]</p> <p>2. ma podbudowaną teoretycznie szczegółową wiedzę związaną z wybranymi zagadnieniami z zakresu informatyki, takimi jak: rozproszone systemy obiektowe, systemy symulacji środowisk rozproszonych, systemy realizacji programów współbieżnych - [K_W5]</p> <p>3. ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach w informatyce i w wybranych pokrewnych dyscyplinach naukowych, - [K_W6]</p> <p>4. ma podstawową wiedzę o cyklu życia systemów informatycznych sprzętowych lub programowych - [K_W7]</p> <p>5. zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich z wybranego obszaru informatyki, w szczególności obejmujące tworzenie oprogramowania rozproszonego - [K_W8]</p>
<p><b>Umiejętności:</b></p> <p>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]</p> <p>2. potrafi przygotować i przedstawić, w języku ojczystym i angielskim, prezentację ustną, dotyczącą szczegółowych zagadnień z zakresu informatyki - [K_U4]</p> <p>3. potrafi określić kierunki dalszego uczenia się i zrealizować proces samokształcenia - [K_U5]</p> <p>4. potrafi posługiwać się technikami informacyjno-komunikacyjnymi wykorzystywanymi przy realizacji przedsięwzięć informatycznych - [K_U7]</p> <p>5. potrafi planować i przeprowadzać eksperymenty, w tym pomiary i symulacje komputerowe, interpretować uzyskane wyniki i wyciągać wnioski - [K_U8]</p> <p>6. potrafi wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych metody analityczne, symulacyjne oraz eksperymentalne - [K_U9]</p> <p>7. 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]</p> <p>8. potrafi formułować i testować hipotezy związane z problemami inżynierskimi i prostymi problemami badawczymi - [K_U12]</p> <p>9. potrafi ocenić przydatność i możliwość wykorzystania nowych osiągnięć (metod i narzędzi) oraz nowych produktów informatycznych - [K_U13]</p> <p>10. potrafi poprawnie użyć przynajmniej jedną metodę szacowania pracochłonności wytwarzania oprogramowania - [K_U15]</p> <p>11. potrafi ocenić architekturę oprogramowania z punktu widzenia wymagań pozafunkcjonalnych - [K_U18]</p> <p>12. potrafi efektywnie uczestniczyć w inspekcji oprogramowania - [K_U19]</p> <p>13. ma umiejętność systematycznego przeprowadzania testów funkcjonalnych - [K_U20]</p> <p>14. potrafi zaproponować ulepszenia (usprawnienia) istniejących rozwiązań technicznych - [K_U21]</p> <p>15. potrafi sformułować wymagania pozafunkcjonalne dla wybranych charakterystyk jakościowych - [K_U23]</p> <p>16. 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]</p> <p>17. 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]</p>
<p><b>Kompetencje społeczne:</b></p> <p>1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K_K1]</p> <p>2. potrafi współdziałać i pracować w grupie, przyjmując w niej różne role - [K_K5]</p> <p>3. potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania - [K_K6]</p>

**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 projektu z zakresu współbieżności (waga do oceny końcowej: 1)
  - ocenę projektu polegającego na implementacji zadania w jednym z poznanych środowisk (waga do oceny końcowej: 2)
  - ocenę prezentacji multimedialnej dotyczącej wybranego zagadnienia technicznego powiązanego z przedmiotem (waga do oceny końcowej: 1)

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

- 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

Program wykładu obejmuje następujące zagadnienia:

1. Symulacyjna ocena wydajności systemu rozproszonego: idea symulatorów bazujących na zdarzeniach w czasie dyskretnym, opis architektury symulacji, konstrukcja modułów symulacyjnych, możliwości środowiska, narzędzie symulowania obciążenia, zbieranie danych statystycznych.
2. Wprowadzenie do środowiska .NET: historia środowiska, architektura i składowe środowiska, język c# w wersjach 1.0-4.0, biblioteki systemowe
3. Programowanie współbieżne i rozproszone: model obsługi wątków, gniazd sieciowych oraz mechanizmów synchronizacji w .NET, zasada działania metod asynchroniczne, zaawansowane mechanizmy komunikacji w .NET Web Services i Remoting
4. Bazy danych ?NoSQL?. Idea nierelacyjnych baz danych, ich podstawowe cechy, podział na rodzaje wraz z ich przykładami, wady i zalety baz nierelacyjnych.
5. Wzorce projektowe dla systemów rozproszonych: idea wzorców projektowych, wzorce projektowe dla systemów rozproszonymi związane z modelowaniem architektury aplikacji, jej projektu, projektowanie modelu dziedziny rozwiązania, interakcja ze źródłami danych, wzorce dla systemów z wywołaniami zdalnymi
6. Środowisko tworzenia aplikacji rozproszonych ICE: architektura środowiska ICE, jego założenia oraz możliwości, przedstawienie narzędzi i usług środowiska, język definiowania interfejsów zdalnych, implementacja z wykorzystaniem ICE
7. Środowisko Windows Communication Foundation - aplikacje Service Oriented Architecture: architektura i założenia paradygmatu Service Oriented Architecture, ogólne cechy rozwiązania Windows Communication Foundation, kontrakty danych, akcji i wiadomości, konfiguracja usług, konfiguracja zachowania usług, bezpieczeństwo w WCF

Zajęcia laboratoryjne prowadzone są w formie 3-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. Biblioteka ADO.NET: model dostępu do danych w bibliotece ADO.NET, integracja ADO.NET z komponentami interfejsu użytkownika SWF, implementacja dostępu do bazy danych z wykorzystaniem wysokopoziomowych funkcji ADO.NET i automatycznych dowiązań, integracja dostawców danych z graficznym interfejsem użytkownika
2. Niskopoziomowy dostęp do baz danych: interfejs dostawców źródeł danych dla baz PostgreSQL i SQLite, narzędzia do ręcznego sterowania połączeniami, zapytaniami i transakcjami, ręczna integracja źródeł danych z komponentami GUI
3. Narzędzia programistyczne: przedstawienie i omówieni idei testów jednostkowych oraz systemów typu Mock, projektowanie testów jednostkowych, narzędzie NUnit, problemy z zależnościami w testach jednostkowych - systemy mock
4. Środowisko symulacji rozproszonych OMNeT++ i jego integracja z mono: przedstawienie zasady działania symulatorów bazujących na zdarzeniach w czasie dyskretnym, omówienie języka opisu architektury symulacji - język NED, przykładowe badania symulacyjne, zbieranie i wstępna analiza danych statystycznych, integracja z środowiskiem mono
5. Gniazda sieciowe i narzędzia zarządzania współbieżnością i wątkami w .NET: model wątków w .NET, podstawowe i zaawansowane klasy do obsługi synchronizacji procesów, metody asynchroniczne, implementacja podstawowych problemów współbieżnych
6. Bazy danych NoSQL, przykłady wykorzystania ze środowiska .NET
7. Środowisko tworzenia aplikacji rozproszonych ICE: wprowadzenie do środowiska, narzędzia i usługi zawarte w środowisku, integracja ICE z .NET, projekt i implementacja przykładowej aplikacji rozproszonej
8. Środowisko tworzenia usług sieciowych .NET WCF, podstawowe narzędzia, projekt i implementacja przykładowej aplikacji
9. Prezentacje studentów dotyczące projektów działających w środowisku .NET o zastosowaniach związanych z systemami rozproszonymi i biznesowymi

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

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:**

1. Enterprise Solution Patterns Using Microsoft .NET, David Trowbridge et. al., 2003, MS Corp
2. Patterns of Enterprise Application Architecture, Martin Fowler, Addison-Wesley Professional; 2002

**Literatura uzupełniająca:**

1. Programming Indigo, David Pallman, 2005, MS Press
2. Developing XML Web Services and Service Components, MS Corp, MS Press

**Bilans nakładu pracy przeciętnego studenta**

Czynność	Czas (godz.)
----------	--------------

1. udział w zajęciach laboratoryjnych:.	24
2. przygotowanie do ćwiczeń laboratoryjnych:	20
3. udział w konsultacjach (mogą być realizowane drogą elektroniczną) związanych z realizacją procesu kształcenia, w szczególności ćwiczeń laboratoryjnych / projektu	2 8
4. udział w wykładach	22
5. napisanie programu / programów, uruchomienie i weryfikacja (czas poza zajęciami laboratoryjnymi)	25
6. zapoznanie się ze wskazaną literaturą / materiałami dydaktycznymi (10 stron tekstu naukowego = 1 godz.), 250 stron	
<b>Obciążenie pracą studenta</b>	
<b>forma aktywności</b>	<b>godzin</b>
<b>ECTS</b>	
Łączny nakład pracy	101
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	34
Zajęcia o charakterze praktycznym	66