



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Programowanie aplikacji sieciowych w Javie i C#

Przedmiot

Kierunek studiów

Elektronika i Telekomunikacja

Studia w zakresie (specjalność)

Sieci komputerowe i technologie internetowe

Poziom studiów

pierwszego stopnia

Forma studiów

stacjonarne

Rok/semestr

3/6

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obieralny

Liczba godzin

Wykład

15

Ćwiczenia

Laboratoria

15

Projekty/seminaria

Inne (np. online)

Liczba punktów ECTS

3

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

dr hab. inż. Mariusz Żal,

mariusz.zal@put.poznan.pl

Odpowiedzialny za przedmiot/wykładowca:

Wymagania wstępne

Student przystępując do tego przedmiotu powinien posiadać podstawową wiedzę w zakresie sieci komputerowych, protokołów sieciowych oraz programowania w języku C#. Student potrafi posługiwać się językami programowania wysokiego poziomu C, C++, C. Potrafi pozyskiwać informacje z literatury i baz danych oraz innych źródeł w języku polskim lub angielskim; potrafi integrować uzyskane informacje, dokonywać ich interpretacji, wyciągać wnioski i uzasadniać opinie. Potrafi realizować projekty zespołowe.

Cel przedmiotu

Celem przedmiotu jest przekazanie studentowi wiedzy nt. mechanizmów wymiany wiadomości w sieciach komputerowych, zapoznanie z obsługą gniazd sieciowych w językach Java i C#. W ramach przedmiotu student zapoznaje się z zasadami tworzenia w C# i Java prostych aplikacji korzystających z protokołu TCP i UDP oraz protokołów warstwy aplikacyjnej. Student poznaje zasady tworzenia prostych aplikacji bazodanowych pracujących w środowisku sieciowym.

Przedmiotowe efekty uczenia się

Wiedza



1. Zna zasady konstrukcji programów komputerowych, posiada wiedzę z zakresu informatyki i zna składnię języków oprogramowania C# i Java z obsługą gniazd, pisanych zarówno dla komputerów PC jak również dla urządzeń o zmniejszonej ilości zasobów.
2. Ma uporządkowaną wiedzę w zakresie architektury komputerów
3. Ma podstawową wiedzę w zakresie podstaw budowy, standardów, architektury, protokołów sieciowych i działania rozległych i lokalnych sieci komputerowych, w szczególności urządzeń sieciowych.
4. Zna protokoły warstwy sieciowej, transportowej i aplikacji

Umiejętności

1. Potrafi przygotować aplikacje pozwalające na wymianę danych w środowisku sieciowym.
2. Potrafi dokonać wyboru protokołu warstwy transportowej i warstwy aplikacji w zależności od wymagań
3. Potrafi przygotować aplikację wielowątkową.
4. Potrafi porozumiewać się w języku polskim lub angielskim w środowiskach zawodowym i w innych środowiskach
5. Potrafi się dalej samodzielnie kształcić, poznawać nowe biblioteki i funkcje języków programowania

Kompetencje społeczne

1. Ma poczucie odpowiedzialności za zaprojektowane systemy elektroniczne i telekomunikacyjne i zdaje sobie sprawę z potencjalnych niebezpieczeństw dla innych ludzi lub społeczeństwa ich nieodpowiedniego wykorzystania. Zna zasady przechowywania informacji oraz określania dostępu do baz danych w celu zapewnienia bezpieczeństwa informacji w nich zawartej.
2. Posiada świadomość wpływu systemów i sieci telekomunikacyjnych i teleinformatycznych na kształtowanie społeczeństwa informacyjnego.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wiedza zdobyta w ramach wykładu weryfikowana jest przez egzamin w formie pisemnej lub ustnej. W formie pisemnej studenci muszą udzielić odpowiedzi na 7 - 10 pytań (testowych i otwartych) różnie punktowanych. Są trzy lub cztery grupy punktowe. Natomiast w przypadku egzaminu ustnego student losuje po jednym pytaniu z każdej grupy punktowej. W formie ustnej, do każdego wylosowanego pytania, student może otrzymać dodatkowe pytanie (związane z wylosowanym pytaniem). Ocena pytania (obejmuje odpowiedź zarówno na pytanie wylosowane jak i pytanie dodatkowe) obejmuje zakres odpowiedzi oraz głębię zrozumienia zagadnienia. Do każdego egzaminu przygotowanych jest 50 - 60 pytań. Warunkiem pozytywnego zaliczenia egzaminu otrzymanie minimum 50% punktów możliwych do zdobycia.

Umiejętności nabyte w ramach laboratorium weryfikowane są na podstawie zadania realizowanego na ostatnich zajęciach. Zadanie podzielny jest na 5 - 6 podzadań różnie punktowanych. Podzadania stanowią całość, możliwe jest jednak niezależne ich wykonanie. Brak wykonania podzadania nie wpływa na ocenę pozostałych podzadań. Próg zaliczeniowy: 50% punktów.

Kryteria oceny egzaminu i zaliczania:

liczba punktów	ocena
$\leq 50\%$	2,0



51% - 60%	3,0
61% - 70%	3,5
71% - 80%	4,0
81% - 90%	4,5
91% - 100%	5,0

Treści programowe

Wykłady:

1. Architektura aplikacji sieciowych w odniesieniu do 3 i 4 warstwy modelu OSI RM
2. Obsługa gniazd sieciowych oraz adresów w warstwie transportowej (porty)
3. Aplikacje synchroniczne i asynchroniczne. Aplikacje wielowątkowe.
4. Aplikacje warstwy TCP: FTP i TFTP.
5. Aplety
6. Serwlety
7. Programowanie protokołów warstwy aplikacji
8. Aplikacje bazodanowe
9. Kojarzenie elementów baz danych z kontrolkami aplikacji
10. Aplikacje rozgłoszeniowe

Ćwiczenia laboratoryjne:

1. Proste aplikacje sieciowe działające w sposób synchroniczny.
2. Sieciowe aplikacje asynchroniczne
3. Sieciowe aplikacje wielowątkowe
4. Aplikacje bazodanowe
5. Warstwa aplikacji: protokół http
6. Klienci FTP i TFTP
7. Tworzenie prostych serwletów

Metody dydaktyczne

1. Wykład:

- a) prezentacje multimedialne ilustrowane przykładami podawanymi na tablicy
- b) analiza przypadków dokonywana bezpośrednio w środowiskach uruchomieniowych dla języków programowania; studenci biorą czynny udział poprzez zadawanie pytań oraz sugerowanie wyników, które można uzyskać

2. Ćwiczenia laboratoryjne: wykonywanie zadań podanych przez prowadzącego (ćwiczenia praktyczne realizowane z wykorzystaniem działających środowisk uruchomieniowych dla języków programowania C# i Java) uzupełniane prezentacjami multimedialnymi.

Literatura



Podstawowa

1. Orłowski S., Grabek M., C# : tworzenie aplikacji sieciowych : gotowe projekty, Helion 2012
2. Hmrajani A., Bucka E., Java : tworzenie aplikacji sieciowych za pomocą Springa, Hibernate i Eclipse, Helion 2012
3. Reese G.. Java : aplikacje bazodanowe : najlepsze rozwiązania, Helion 2003.

Uzupełniająca

1. Richard Blum, C# Network Programming, Sybex 2003
2. Anthony Potts, David H. Friedel Jr, Java programming language handbook, Coriolis Group Inc. 1996

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	75	3,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	31	2,0
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych, przygotowanie do egzaminu) ¹	44	1,0

¹ niepotrzebne skreślić lub dopisać inne czynności