

Programowanie sieciowe - opis przedmiotu

| Informacje ogólne | |
|---------------------|--|
| Nazwa przedmiotu | Programowanie sieciowe |
| Kod przedmiotu | 11.3-WI-INFD-PS |
| Wydział | Wydział Nauk Inżynieryjno-Technicznych |
| Kierunek | Informatyka |
| Profil | ogólnoakademicki |
| Rodzaj studiów | drugiego stopnia z tyt. magistra inżyniera |
| Semestr rozpoczęcia | semestr zimowy 2021/2022 |

| Informacje o przedmiocie | |
|---------------------------------|-----------------------------|
| Semestr | 3 |
| Liczba punktów ECTS do zdobycia | 4 |
| Typ przedmiotu | obowiązkowy |
| Język nauczania | polski |
| Sylabus opracował | • dr inż. Tomasz Gratkowski |

| Formy zajęć | | | | | |
|--------------|---|--|--|---|---------------------|
| Forma zajęć | Liczba godzin w semestrze (stacjonarne) | Liczba godzin w tygodniu (stacjonarne) | Liczba godzin w semestrze (niestacjonarne) | Liczba godzin w tygodniu (niestacjonarne) | Forma zaliczenia |
| Laboratorium | 30 | 2 | 18 | 1,2 | Zaliczenie na ocenę |
| Wykład | 15 | 1 | 9 | 0,6 | Zaliczenie na ocenę |

Cel przedmiotu

- zapoznanie studentów z podstawowymi technikami wytwarzaniem aplikacji sieciowych
- ukształtowanie podstawowych umiejętności w zakresie projektowania i wytwarzania aplikacji sieciowych w języku Java

Wymagania wstępne

Zakres tematyczny

Wprowadzenie: Wysokopoziomowy mechanizm dostępu do zasobów sieci globalnej - Internet. Programy sieci WWW. Interaktywne aplety Javy. Dynamiczne strony WWW. Stosowanie serwerów aplikacyjnych. Sieciowe rozwiązania korporacyjne. Obiekty zasobów URL. Połączenia sieciowe wykorzystujące interfejs programowy URL, URLConnection, HttpURLConnection.

Połączenia komunikacyjne niezawodnym strumieniem TCP. Model interakcji klient-serwer. Pojęcie gniazd sieciowych - interfejs Socket, ServerSocket. Komunikacja z wykorzystaniem protokołu UDP. Programy klient - serwer wykorzystujące UDP. Gniazda UDP - interfejs DatagramSocket. Pojęcie pakietu datagramu - interfejs DatagramPacket. Pojęcie Broadcastingu i Multicatingu - interfejs MulticastSocket. Zasady tworzenie protokołów sieciowych warstw aplikacji modelu TCP/IP.

Programowanie usług sieci Internet. Wykorzystanie dedykowanych pakietów języka Java do budowy rozproszonych usług sieciowych. Architektura oparta na usługach (SOA) oraz użycie stylu architektonicznego REST (JAX-RS). Zastosowanie protokołu SOAP (JAX-WS). Interakcyjne używanie odległych maszyn.

Dostęp do sieciowych zasobów danych. Zasady budowania wielowarstwowych systemów sieciowych.

Metody kształcenia

wykład: wykład konwencjonalny/tradycyjny

laboratorium: ćwiczenia laboratoryjne

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

| Opis efektu | Symbol e efektów | Metody weryfikacji | Forma zajęć |
|--|--------------------|---|----------------|
| Potrąfi wytłumaczyć mechanizm działania gniazd sieciowych. | • K_W11 | • zaliczenie - ustne, opisowe, testowe i inne | • Wykład |
| Potrąfi korzystać ze standardowych protokołów sieciowych. | • K_W11 • K_U14 | • bieżąca kontrola na zajęciach | • Laboratorium |
| Potrąfi dobrać odpowiedni protokół sieciowy w celu zoptymalizowania działania tworzonej aplikacji sieciowej. | • K_U01 • K_U14 | • bieżąca kontrola na zajęciach | • Laboratorium |
| Potrąfi opracować własny protokół sieciowy. | • K_W11 • K_U14 | • bieżąca kontrola na zajęciach | • Laboratorium |

| Opis efektu | Symbole efektów | Metody weryfikacji | Forma zajęć |
|---|--|---|--|
| Student potrafi wyjaśnić sposób dostępu do zasobów sieci globalnej Internet przy wykorzystaniu języka Java. | <ul style="list-style-type: none"> • K_W11 • K_U01 | <ul style="list-style-type: none"> • zaliczenie - ustne, opisowe, testowe i inne | <ul style="list-style-type: none"> • Wykład |
| Potrafi zaprojektować i utworzyć aplikację sieciową w języku obiektowym Java. | <ul style="list-style-type: none"> • K_U14 | <ul style="list-style-type: none"> • bieżąca kontrola na zajęciach | <ul style="list-style-type: none"> • Laboratorium |

Warunki zaliczenia

Wykład - warunkiem zaliczenia jest uzyskanie pozytywnej oceny z egzaminu pisemnego lub testu.

Laboratorium - warunkiem zaliczenia jest uzyskanie pozytywnych ocen ze wszystkich ćwiczeń laboratoryjnych, przewidzianych do realizacji w ramach programu laboratorium.

Składowe oceny końcowej = wykład: 50% + laboratorium: 50%

Literatura podstawowa

1. Stevens W.R.: UNIX. Programowanie usług sieciowych. Tom 1 - API: gniazda i XTI; Wydawnictwa Naukowo-Techniczne, 2002.
2. Burns B.: Projektowanie systemów rozproszonych. Wzorce i paradygmaty dla skalowalnych, niezawodnych usług, Helion, 2018
3. Harold E. R., Java Network Programming, 4th Edition, Developing Networked Applications. O'Reilly, 2013

Literatura uzupełniająca

1. Horstmann C. S., Cornell G.: Java. Podstawy. Wydanie IX, Helion, 2013.
2. Horstmann C. S., Cornell G.: Java. Techniki zaawansowane. Wydanie IX, Helion, 2013.
3. The Java Tutorial / ORACLE.
4. The Java EE Tutorial / ORACLE.
5. Stevens W.R: TCP/IP. Tom 1: Protokoły - Biblia; Oficyna Wydawnicza READ ME, 1998.
6. Comer D. E., Sieci komputerowe i internecie. Wydanie V, Helion, 2012.
7. Rychlicki-Kicior K., Java EE 6. Programowanie aplikacji WWW, Helion, 2012.
8. Erl, Th., SOA., Koncepcje, technologie i projektowanie, 2014.
9. Rotem-Gal-Oz A., Wzorce SOA, Helion, 2013
10. Sharma S., Mikrouslugi w Javie. Poradnik eksperta, Helion, 2017
11. Rhuan Rocha, Joao Purificacao J.: Java EE 8. Wzorce projektowe i najlepsze praktyki, Helion, 2019
12. Burns B.: Tao mikrouslug. Projektowanie i wdrażanie, Helion, 2019

Uwagi

Zmodyfikowane przez prof. dr hab. inż. Andrzej Obuchowicz (ostatnia modyfikacja: 20-04-2021 08:48)

Wygenerowano automatycznie z systemu SyllabUZ