

# KARTA PRZEDMIOTU

## 1. Informacje ogólne

|   |   |
|---|---|
| <b>Nazwa przedmiotu i kod (wg planu studiów):</b> | Zaawansowane techniki programowania aplikacji sieciowych w języku C++<br><b>D1_10</b> |
| <b>Nazwa przedmiotu (j. ang.):</b>                | Advanced Paradigms of Network Application Development in C++ Language.                |
| <b>Kierunek studiów:</b>                          | Informatyka   |
| <b>Specjalność/specjalizacja:</b>                 | Sieciowe Systemy Informatyczne  |
| <b>Poziom kształcenia:</b>                        | studia I stopnia  |
| <b>Profil kształcenia:</b>                        | praktyczny (P)  |
| <b>Forma studiów:</b>                             | studia stacjonarne / studia niestacjonarne  |
| <b>Obszar kształcenia:</b>                        | nauki techniczne  |
| <b>Dziedzina:</b>                                 | nauki techniczne  |
| <b>Dyscyplina nauki:</b>                          | Informatyka   |
| <b>Koordynator przedmiotu:</b>                    | dr inż. Piotr Wais  |

## 2. Ogólna charakterystyka przedmiotu

|   |   |
|---|---|
| <b>Przynależność do modułu:</b>                                 | specjalizacyjnego do wyboru   |
| <b>Status przedmiotu:</b>                                       | obowiązkowy   |
| <b>Język wykładowy:</b>   | polski  |
| <b>Rok studiów, semestr:</b>                                    | II, 4   |
| <b>Forma i wymiar zajęć według planu studiów:</b>               | stacjonarne - wykład 15 h, ćw. laboratoryjne 30 h<br>niestacjonarne - wykład 15 h, ćw. laboratoryjne 15 h |
| <b>Interesariusze i instytucje partnerskie (nieobowiązkowe)</b> |   |
| <b>Wymagania wstępne / Przedmioty wprowadzające:</b>            | Programowanie w języku C++ / Programowanie II   |

### 3. Bilans punktów ECTS

| Całkowita liczba punktów ECTS (wg planu studiów; 1 punkt =25-30 godzin pracy studenta, w tym praca na zajęciach i poza zajęciami):   | 6<br>(A + B)                             |             |                |
|--|--|-------------|----------------|
|  |  | stacjonarne | Niestacjonarne |
| <b>A. Liczba godzin wymagających bezpośredniego udziału nauczyciela</b> (kontaktowych, w czasie rzeczywistym, w tym testy, egzaminy etc) <b>z podziałem na typy zajęć oraz całkowita liczba punktów ECTS osiągniętych na tych zajęciach</b>  | obecność na wykładach                    | 15          | 15             |
|  | obecność na ćwiczeniach laboratoryjnych  | 30          | 15             |
|  | egzamin                                  | 2           | 2              |
|  | udział w konsultacjach                   | 5           | 5              |
|  | <b>w sumie:</b>                          | 52          | 37             |
|  | ECTS                                     | 2           | 1,5            |
| <b>B. Poszczególne typy zadań do samokształcenia studenta</b> (niewymagających bezpośredniego udziału nauczyciela) <b>wraz z planowaną średnią liczbą godzin na każde i sumaryczną liczbą ECTS</b> (np. praca w bibliotece, w sieci, na platformie e-learningowej, w laboratorium, praca nad projektem końcowym, przygotowanie ogólne; suma poszczególnych godzin powinna zgadzać się z liczbą ogólną) | przygotowanie do ćwiczeń laboratoryjnych | 20          | 20             |
|  | wykonanie projektu                       | 20          | 20             |
|  | praca w sieci                            | 15          | 20             |
|  | przygotowanie do egzaminu                | 15          | 15             |
|  | przygotowanie do konsultacji             | 10          | 10             |
|  | uzupełnienie/studiowanie notatek         | 5           | 8              |
|  | studiowanie zalecanej literatury         | 15          | 20             |
|  | <b>w sumie:</b>                          | 100         | 113            |
| ECTS   | 4  | 4,5         |                |
| <b>C. Liczba godzin praktycznych/laboratoryjnych w ramach przedmiotu oraz związana z tym liczba punktów ECTS</b> (ta liczba nie musi być powiązana z liczbą godzin kontaktowych, niektóre zajęcia praktyczne/laboratoryjne mogą odbywać się bez udziału nauczyciela):  | udział w ćwiczeniach laboratoryjnych     | 30          | 15             |
|  | wykonanie projektu                       | 20          | 25             |
|  | praca praktyczna samodzielna             | 12          | 22             |
|  | <b>w sumie:</b>                          | 62          | 62             |
|  | ECTS                                     | 2,5         | 2,5            |

### 4. Opis przedmiotu

|   |
|---|
| <p><b>Cel przedmiotu:</b><br/>Osiągnięcie podstawowej wiedzy z zakresu programowania aplikacji sieciowych w środowisku MS Visual Studio C++.</p>  |
| <p><b>Metody dydaktyczne:</b> wykład informacyjny, pokaz, ćwiczenia laboratoryjne, metoda projektów</p> <p><i>np. podające (wykład), problemowe (konwersatorium, seminarium), aktywizujące (symulacja, metoda przypadków itp. ), eksponujące (pokaz, film), praktyczne (ćwiczenia, metoda projektów itp) – pełniejszy wykaz poniżej (prosimy wybrać najstosowniejsze - jedną lub więcej, można dodać własne metody)</i></p>   |
| <p><b>Treści kształcenia</b> (w rozbiciu na formę zajęć (jeśli są różne formy) i najlepiej w punktach):</p> <p><b>Wykłady:</b></p> <ol style="list-style-type: none"> <li>1. Visual Studio C++ - Platforma .NET – przegląd komponentów biblioteki Windows Forms, projektowanie interfejsu aplikacji, menu główne, menu kontekstowe, metody zdarzeniowe, okna dialogowe i pliki tekstowe, edycja i korzystanie ze schowka, drukowanie, ekran powitalny, przygotowanie ikony w obszarze powiadamiania, odtwarzanie pliku dźwiękowego, wczytywanie obrazu, ustawienia aplikacji.</li> <li>2. Programowanie sieciowe – sieci komputerowe, protokoły TCP i UDP, protokół IP i adresy MAC, programowanie klient- serwer i peer-to-peer.</li> <li>3. Aplikacje TCP i UDP - Połączenie TCP – klient, serwer, odczytanie adresu IP przyłączonego hosta.</li> </ol> |

Połączenie UDP – klient, serwer. Asynchroniczne połączenie TCP. Prosty skaner otwartych portów hosta zdalnego. Skaner otwartych portów lokalnego hosta. Sprawdzenie adresu IP naszego komputera. Komplet informacji na temat połączeń sieciowych. Ping. Ping - przeciwdziałanie zablokowaniu interfejsu. NetDetect - sprawdzanie dostępnych komputerów w sieci. Traceroute - śledzenie drogi pakietu ICMP.

4. Remoting - Serwer i klient TCP.

5. ASP.NET – ping, wysyłanie wiadomości e-mail, pobieranie plików na serwer

6. Web Services – usługa sieciowa

7. WCF - Podstawy działania. WCF = A + B + C. Definiowanie kontraktu. Udostępnianie usługi. Tworzenie klienta.

### Ćwiczenia laboratoryjne:

1. Projektowanie aplikacji w środowisku Visual Studio C++ - Platforma .NET – komponenty biblioteki Windows Forms

2. Projektowanie aplikacji w środowisku Visual Studio C++:

- Aplikacje TCP i UDP,
- Remoting - Serwer i klient TCP.
- ASP.NET – ping, wysyłanie wiadomości e-mail, pobieranie plików na serwer
- Web Services – usługa sieciowa
- WCF - Definiowanie kontraktu. Udostępnianie usługi. Tworzenie klienta.

## 5. Efekty kształcenia i sposoby weryfikacji

**Efekty kształcenia** (w sumie wymienić ok. od 3 do 9 efektów - podać numery efektów z listy dla danego kierunku/specjalności – opublikowane na stronie uczelni; podać TYLKO te efekty (tam gdzie to możliwe i stosowne w trzech kategoriach, np. kompetencje społeczne mogą nie być realizowane w tym przedmiocie), na których osiągnięcie kładzie się nacisk w ramach przedmiotu, wybrane efekty kierunkowe powinny być bardziej szczegółowo sformułowane niż te dla całej specjalności, tak aby były weryfikowalne – dlatego mają osobne symbole jako efekty przedmiotu)

| Efekt przedmiotu (kod przedmiotu + kod efektu kształcenia) | Student, który zaliczył przedmiot (spełnił minimum wymagań)  | Efekt kierunkowy                                   |
|--|--|--|
| D1_10_W01<br>D1_10_W02<br>D1_10_W03                        | <b>Wiedza:</b> <ol style="list-style-type: none"> <li>1. Student zna narzędzia i mechanizmy potrzebne do zbudowania aplikacji Windows Forms w środowisku MS Visual Studio C++.</li> <li>2. Student potrafi programować aplikacje w środowisku MS Visual Studio C++ wykorzystując technologie sieciowe.</li> <li>3. Student wie jak programować dostosowując swój projekt do ciągle zmieniających się trendów i możliwości.</li> </ol>  | K_W08<br>K_W16<br>K_W07                            |
| D1_10_U01<br>D1_10_U02<br>D1_10_U03                        | <b>Umiejętności:</b> <ol style="list-style-type: none"> <li>1. Student potrafi poszerzać i aktualizować swoją wiedzę niezbędną do zbudowania aplikacji w środowisku MS Visual Studio C++ wykorzystując komponenty biblioteki Windows Forms zgodnie z obowiązującymi standardami i rozwiązaniami.</li> <li>2. Student potrafi na podstawie algorytmu (specyfikacji) stworzyć prostą aplikację sieciową w środowisku MS Visual Studio C++ wykorzystując technologie sieciowe.</li> <li>3. Student potrafi zarządzać danymi z poziomu aplikacji.</li> </ol> | K_U03<br>K_U04<br>K_U10<br>K_U17<br>K_U18<br>K_U11 |

|           |   |  |
|-----------|---|--|
| D1_10_K01 | <b>Kompetencje społeczne</b><br><b>1.</b> Student rozumie potrzebę poznawania nowych narzędzi programistycznych wykorzystywanych w środowisku MS Visual Studio C++. | K_K01  |
| D1_10_K02 |   | <b>2.</b> Student potrafi pracując w zespole zaprojektować i zaimplementować aplikację sieciową w środowisku MS Visual Studio C++. |

**Sposoby weryfikacji efektów kształcenia:**

(np. dyskusja, gra dydaktyczna, zadanie e-learningowe, ćwiczenie laboratoryjne, projekt indywidualny/grupowy, zajęcia terenowe, referat studenta, praca pisemna, kolokwium, test zaliczeniowy, egzamin, opinia eksperta zewnętrznego, etc. Dodać do każdego wybranego sposobu symbol zakładanego efektu, jeśli jest ich więcej)

| Lp. | Efekt przedmiotu   | Sposób weryfikacji                                       | Ocena formująca   | Ocena końcowa  |
|-----|--|--|---|--|
| 1   | D1_10_W01<br>D1_10_W02<br>D1_10_W03<br>D1_10_U01<br>D1_10_U02<br>D1_10_U03 | Egzamin  | ocena z egzaminu – sprawdzian wiedzy i umiejętności             | Ocena końcowa z egzaminu   |
| 2   | D1_10_U01<br>D1_10_U02<br>D1_10_U03<br>D1_10_K01<br>D1_10_K02              | ćwiczenia laboratoryjne,<br>projekt indywidualny/grupowy | ocena z projektu końcowego,<br>ocena zaangażowania na zajęciach | Ocena końcowa z laboratorium - średnia z ocen formujących sprawdzających nabyte umiejętności i kompetencje |

**Kryteria oceny** (oceny 3,0 powinny być równoważne z efektami kształcenia, choć mogą być bardziej szczegółowo opisane):

| w zakresie wiedzy |   | Efekt kształcenia                   |
|-------------------|---|-------------------------------------|
| Na ocenę 3,0      | Student uzyskał min. 50% wymaganej wiedzy w zakresie obowiązującego materiału. Student:<br>- zna podstawowe narzędzia potrzebne do zbudowania aplikacji Windows Forms – MS Visual Studio C++, oraz mechanizmy, takie jak: kreator projektu,<br>- zna podstawowe pojęcia w zakresie programowania aplikacji sieciowych w środowisku MS Visual Studio C++ wykorzystując technologie sieciowe.<br>- wie jak aktualizować środowisko programistyczne potrzebne do programowania aplikacji w obecnie obowiązującej wersji. | D1_10_W01<br>D1_10_W02<br>D1_10_W03 |
| Na ocenę 5,0      | Student zdobył powyżej 95% wymaganej wiedzy w zakresie obowiązującego materiału. Student:<br>- zna dodatkowe mechanizmy, takie jak: debugger,<br>- zna podstawowe i zaawansowane pojęcia w zakresie programowania aplikacji sieciowych w środowisku MS Visual Studio C++ wykorzystując technologie sieciowe   | D1_10_W01<br>D1_10_W02<br>D1_10_W03 |

|   |  |                                     |
|---|--|-------------------------------------|
|   | - zna sposoby wykorzystania dodatkowych funkcji środowiska programistycznego.  |                                     |
| <b>w zakresie umiejętności</b>  |  |                                     |
| Na ocenę 3,0  | Student uzyskał min. 50% wymaganych umiejętności w zakresie obowiązującego materiału. Student potrafi:<br>- stworzyć nowy projekt zgodnie z obowiązującymi standardami i rozwiązaniami.<br>- na podstawie algorytmu (specyfikacji) stworzyć prostą aplikację w środowisku MS Visual Studio C++ wykorzystując komponenty biblioteki Windows Forms.<br>- stworzyć prostą aplikację sieciową w środowisku MS Visual Studio C++ wykorzystując technologie sieciowe.  | D1_10_U01<br>D1_10_U02<br>D1_10_U03 |
| Na ocenę 5,0  | Student uzyskał powyżej 95% umiejętności w zakresie obowiązującego materiału. Student umie:<br>- stworzyć nowy projekt zgodnie z obowiązującymi standardami i rozwiązaniami oraz dołączyć dodatkowe biblioteki do projektu<br>- na podstawie algorytmu (specyfikacji) stworzyć zaawansowaną aplikację w środowisku MS Visual Studio C++ wykorzystując komponenty biblioteki Windows Forms.<br>- stworzyć zaawansowaną aplikację sieciową w środowisku MS Visual Studio C++ wykorzystując technologie sieciowe. | D1_10_U01<br>D1_10_U02<br>D1_10_U03 |
| <b>w zakresie kompetencji społecznych</b>   |  |                                     |
| Na ocenę 3,0  | Student osiągnął wymagane kompetencje społeczne na poziomie min. 50%.  | D1_10_K01<br>D1_10_K02              |
| Na ocenę 5,0  | Student osiągnął wymagane kompetencje społeczne na poziomie wyższym niż 90%.   | D1_10_K01<br>D1_10_K02              |
| <p><b>Kryteria oceny końcowej</b> (zaleca się podział procentowy poszczególnych kryteriów składających się na ocenę końcową, który może współgrać z powyższymi kryteriami: np. aktywność za zajęciami... %, kolokwia ...%, samodzielne ćwiczenia ...%, laboratoria ... % ocena z projektu (szczególnie istotna)- ...%, zajęcia terenowe...%, zaliczenie, egzamin pisemny... %, opinia eksperta zewnętrznego ...% itp. )</p> <p>ocena z egzaminu: 100%</p> <p>ocena z laboratorium:<br/>ocena z projektu: 80 %<br/>samodzielne wykonanie ćwiczeń laboratoryjnych: 15%<br/>aktywność na zajęciach: 5%</p> |  |                                     |
| <b>Zalecana literatura</b>  |  |                                     |
| <p><b>Literatura podstawowa:</b></p> <ol style="list-style-type: none"> <li>1. Matulewski J., Visual Studio 2013: podręcznik programowania w C# z zadaniami</li> <li>3. Maciej Grabek, WCF od podstaw. Komunikacja sieciowa nowej generacji, Helion 2012</li> <li>4. msdn.microsoft.com</li> <li>5. cnap.pwsz.krosno.pl</li> </ol> <p><b>Literatura uzupełniająca:</b></p> <ol style="list-style-type: none"> <li>1. Sharp, John, Microsoft Visual C# 2013: Krok po kroku, APN Promise, 2014</li> </ol>   |  |                                     |

**Informacje dodatkowe:**

|  |
|--|
| Dodatkowe obowiązki prowadzącego wraz z szacowaną całkowitą liczbą godzin: |
| Przygotowanie do wykładów i ćwiczeń laboratoryjnych – 30 godzin            |
| Konsultacje – 10 godzin  |
| Poprawa prac projektowych – 15 godzin                                      |
| Przygotowanie i poprawa egzaminu – 10 godzin                               |
| W sumie: 65 godzin   |