



WYŻSZA SZKOŁA
INFORMATYKI I
ZARZĄDZANIA Z
SIEDZIBĄ W
RZESZOWIE



Studia podyplomowe "Inżynieria oprogramowania"

Numer usługi 2024/12/04/14073/2446985

7 200,00 PLN brutto

7 200,00 PLN netto

32,73 PLN brutto/h

32,73 PLN netto/h

📍 Rzeszów / mieszana (stacjonarna połączona z usługą zdalną w czasie rzeczywistym)

📖 Studia podyplomowe

🕒 220 h

📅 11.10.2025 do 30.06.2026

Informacje podstawowe

Kategoria	Informatyka i telekomunikacja / Bazy danych
Identyfikator projektu	Małopolski Pociąg do kariery
Sposób dofinansowania	wsparcie dla osób indywidualnych wsparcie dla pracodawców i ich pracowników
Grupa docelowa usługi	Studia adresowane są do osób, które planują rozszerzyć swoje kwalifikacje o umiejętność tworzenia oprogramowania i wiązą swoją przyszłość z pracą jako programista. Usługa również adresowana dla Uczestników Projektu "Małopolski pociąg do kariery - sezon 1" i/lub dla Uczestników Projektu "Nowy start w Małopolsce z EURESem"
Minimalna liczba uczestników	18
Maksymalna liczba uczestników	30
Data zakończenia rekrutacji	10-10-2025
Forma prowadzenia usługi	mieszana (stacjonarna połączona z usługą zdalną w czasie rzeczywistym)
Liczba godzin usługi	220
Podstawa uzyskania wpisu do BUR	art. 163 ust. 1 ustawy z dnia 20 lipca 2018 r. Prawo o szkolnictwie wyższym i nauce (t.j. Dz. U. z 2023 r. poz. 742, z późn. zm.)

Cel

Cel edukacyjny

Studia podyplomowe Inżynieria oprogramowania wraz z egzaminem potwierdzają przygotowanie do pracy w zespołach wytwarzających oprogramowanie od projektowania, poprzez programowanie, a na testowaniu i dokumentowaniu aplikacji skończywszy.

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Charakteryzuje istotne fakty, pojęcia, zasady i teorie na których bazuje programowanie, w tym elementy zarządzania informacją, jej gromadzenia oraz przetwarzania.	Wyjaśnia pojęcia dotyczące złożoności obliczeniowej i definiowania algorytmów oraz określić złożoność algorytmu.	Test teoretyczny
Dobiera oraz wykorzystuje standardy i normy pozwalające na skuteczne programistyczne rozwiązywanie zadań naukowych, technicznych, organizacyjnych oraz z obszaru zarządzania.	Tworzy prostą aplikację w języku Java w oparciu o wytyczne	Prezentacja
Rozróżnia zagadnienia w zakresie metod, technik i narzędzi stosowanych podczas programistycznego rozwiązywania zadań naukowych, technicznych oraz z obszaru zarządzania.	Używa wybrane techniki i metody programowania obiektowego	Obserwacja w warunkach rzeczywistych
Charakteryzuje zagadnienia związane rolą i organizacją poszczególnych etapów cyklu życia projektu informacyjnego, a także szacowania kosztów jego projektowania, budowy oraz eksploatacji.	Sprawnie posługuje się technikami i metodami programowania obiektowego	Obserwacja w warunkach rzeczywistych
Stosuje zasady, sposoby oraz metody projektowania aplikacji informatycznych, a także pojmując znaczenie kompromisów w fazie wyboru efektywnego rozwiązania projektowego.	Wymienia wszystkie dyscypliny procesu produkcyjnego oprogramowania	Test teoretyczny
Definiuje zagadnienia w zakresie algorytmiki oraz metod i technik programowania, w różnych obszarach zawodowej i społecznej aktywności człowieka.	Wyjaśnia zagadnienia w zakresie algorytmiki oraz metod i technik programowania, w różnych obszarach zawodowej i społecznej aktywności człowieka.	Test teoretyczny

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Charakteryzuje metodykę wdrażania i eksploatacji systemów informacyjnych, pojmując zależności między komponentami i ich wpływ na cykl życia produktu.	Wykorzystuje techniki programistyczne dostosowane do danego zagadnienia	Test teoretyczny
Projektuje założenia systemu informacyjnego, bazującego na dostępnych narzędziach i technologiach.	Buduje strukturę aplikacji z podziałem na warstwy i komponenty	Test teoretyczny
Definiuje specyfikację funkcjonalną systemu informacyjnego z uwzględnieniem dostępnych technologii, aspektów prawnych, w tym ochrony własności intelektualnej oraz innych względów pozatechnicznych, takich jak: uwarunkowania środowiskowe, społeczne i ekonomiczne.	Przeprowadza analizę wymagań. Buduje analityczny model dziedziny oraz model przypadków użycia.	Test teoretyczny
Rozwiązuje zadania informatyczne, świadomie i skutecznie korzystając ze standardowych narzędzi oraz ogólnodostępnych zasobów programistycznych.	Projektuje bazę danych dla systemu informatycznego	Test teoretyczny
Ocenia i porównuje aspekty sprzętowe i programowe systemów informacyjnych ze względu na zadane kryteria użytkowe i ekonomiczne (możliwości funkcjonalne, szybkość działania, koszty zakupu oraz eksploatacji itp.).	Wykorzystuje cechy języka programowania, w szczególności standardowe struktury danych do rozwiązania postawionego zagadnienia.	Test teoretyczny
Planuje procedurę testowania systemu informacyjnego oraz określić modyfikacje mające na celu jego dostosowanie do konkretnych potrzeb użytkowników.	Projektuje i implementuje program dla zadanego problemu z wykorzystaniem instrukcji złożonych oraz zweryfikować poprawność jego działania.	Test teoretyczny
Opracowuje dokumentację użytkową opisującą pracę z przygotowanym rozwiązaniem informatycznym.	Projektuje na poziomie komponentów aplikację trójwarstwową.	Obserwacja w warunkach rzeczywistych
Uświadamia społeczne znaczenie swojego zawodu, wykazuje wysoką motywację do prowadzenia działalności zawodowej.	Prezentuje wyniki swoich prac i badań	Wywiad swobodny

Kwalifikacje

Kompetencje

Usługa prowadzi do nabycia kompetencji.

Warunki uznania kompetencji

Pytanie 1. Czy dokument potwierdzający uzyskanie kompetencji zawiera opis efektów uczenia się?

Tak. Absolwent studiów podyplomowych uzyskuje świadectwo zgodnie z obowiązującym rozporządzeniem ministerialnym oraz zaświadczenie o osiągniętych efektach uczenia się.

Pytanie 2. Czy dokument potwierdza, że walidacja została przeprowadzona w oparciu o zdefiniowane w efektach uczenia się kryteria ich weryfikacji?

Tak. Każdy przedmiot kończy się zaliczeniem, zaliczeniem na ocenę lub egzaminem zgodnie z wytycznymi zawartymi w kartach przedmiotów.

Pytanie 3. Czy dokument potwierdza zastosowanie rozwiązań zapewniających rozdzielenie procesów kształcenia i szkolenia od walidacji?

Tak. Po uzyskaniu zaliczeń i zdaniu egzaminów przedmiotowych oraz zakończeniu zajęć dydaktycznych słuchacz zdaje egzamin końcowy w formie ustnej wypowiedzi przed powołaną komisją.

Program

Program studiów zawiera następujące zagadnienia:

Algorytmy i struktury danych

1. Podstawowe zasady analizy algorytmów:
 1. poprawność
 2. złożoność obliczeniowa algorytmu (pesymistyczna, oczekiwana)
 3. koszt amortyzowany: metoda potencjału
2. Podstawowe techniki i struktury:
 1. metoda dziel i zwyciężaj
 2. metoda zachłanna
 3. programowanie dynamiczne
 4. transformacyjna konstrukcja algorytmu
 5. elementarne struktury danych: stosy, kolejki, listy
3. Sortowanie:
 1. sortowanie przez porównania (InsertionSort, QuickSort, MergeSort)
 2. proste kolejki priorytetowe: kopce binarne
 3. HeapSort
 4. sortowanie pozycyjne
 5. złożoność problemu sortowania
4. Selekcja:
 1. algorytm Hoare'a
 2. algorytm magicznych piątek
5. Wyszukiwanie i proste słowniki:
 1. wyszukiwanie liniowe i binarne
 2. prosty słownik: drzewa poszukiwań binarnych
 3. haszowanie
6. Efektywne implementacje słownika:
 1. drzewa AVL

2. drzewa typu splay
3. B-drzewa
7. Złożone struktury danych:
 1. wzmocnione kolejki priorytetowe: kolejki dwumianowe, kopce Fibonacciego
 2. efektywne sumowanie zbiorów rozłącznych
8. Algorytmy grafowe:
 1. DFS i jego zastosowania
 2. problemy ścieżkowe – Algorytm Dijkstry
 3. minimalne drzewo rozpinające
9. Wyszukiwanie wzorca w tekstach:
 1. prefikso-sufiksy
 1. algorytm Knutha-Morisa-Pratta
 2. Tekstowe struktury danych:
 1. tablice sufiksowe
 2. drzewa sufiksowe
 3. NP-zupełność:
 1. klasa NP.
 2. problemy NP-trudne i NP-zupełne.

Języki i metody programowania - cz.1:

1. Wprowadzenie do koncepcji JVM i środowiska Java
2. Podstawowe pojęcia programowania obiektowego
3. Język programowania Java – składnia, zmienne, operatory, tablice, instrukcje sterujące
4. Klasy, obiekty, metody, konstruktory

Techniki wytwarzania złożonego oprogramowania:

1. Zadania inżynierii oprogramowania
2. Dyscypliny związane z tworzeniem oprogramowania
3. Podstawowe modele wytwarzania oprogramowania
4. Manifest zwinnego tworzenia oprogramowania
5. Od zbierania wymagań do wdrożenia - proces wytwórczy
6. Metodyka pracy w zespole SCRUM-owym – role i ich odpowiedzialność, podstawowe zasady współpracy
7. Cykl produkcyjny w podejściu SCRUM-owym, zdarzenia i artefakty
8. Wprowadzenie do programowania zorientowanego obiektowo
9. Notacja UML
10. Zasady GRASP
11. Zasady SOLID
12. Wzorce projektowe
13. Wprowadzenie do podejścia Domain-Driven Design
14. Wzorce strategiczne i taktyczne DDD

Bazy danych

1. Wstęp
 1. 1. Pojęcie relacyjnej bazy danych
 2. Model danych
 3. Język SQL – wstęp
2. Wyświetlanie danych z użyciem języka SQL
 1. Budowa zapytania SELECT w języku SQL
 2. Aliasy kolumn
 3. Wyrażenia arytmetyczne,
 4. Praca z wartościami NULL,
 5. Łączenie łańcuchów znakowych,
 6. Wyświetlanie struktury tabeli.
3. Ograniczenia i sortowanie danych
 1. Użycie klauzuli WHERE
 2. Operatory porównań
 3. Operatory BETWEEN, IN, LIKE i znaki zastępcze,

4. Warunek NULL
5. Operatory logiczne AND, OR i NOT
6. Sortowanie i klauzula ORDER BY
4. Użycie funkcji jednowierszowych do personalizacja wyjścia
 1. Typy funkcji SQL
 2. Funkcje znakowe
 3. Funkcje numeryczne
 4. Operacje na datach
5. Używanie funkcji konwertujących i wyrażenia warunkowe
 1. Funkcje TO_CHAR, TO_DATE i TO_NUMBER,
 2. Funkcje NVL, NVL2, NULLIF, COALESCE,
 3. Funkcje CASE i DECODE
6. Agregacja danych z wykorzystaniem funkcji grupujących
 1. Pojęcie funkcji grupującej
 2. Wykorzystanie funkcji AVG, SUM, MIN, MAX i COUNT,
 3. Wykorzystanie słowa kluczowe DISTINCT,
 4. Wykorzystanie klauzul HAVING i ORDER BY.
7. Wyświetlanie danych z wielu tabel
 1. Rodzaje połączeń tabel
 2. Konstrukcja SELECT w zapytaniu do wielu tabel i klauzule: ON, USING.
 3. Wykorzystanie słów INNER, LEFT, RIGHT, FULL, OUTER do tworzenia różnych rodzajów połączeń tabel
8. Tworzenie zapytań złożonych z wykorzystaniem podzapytań
 1. Rodzaje podzapytań
 2. Wykorzystanie słów kluczowych Having, ANY, ALL EXIST w podzapytaniach
9. Tworzenie zapytań złożonych
 1. Rodzaje zapytań złożonych
 2. Operatory zapytań złożonych UNION, UNION ALL, INTERSECT, MINUS
10. Zmiana danych w bazie danych
 1. Wyrażenia INSERT, UPDATE, DELETE i TRUNCATE,
 2. Pojęcie transakcji, słowa kluczowe COMMIT lub ROLLBACK.
11. Użycie wyrażen DDL do tworzenia i zarządzania tabelami:
 1. Tworzenie tabel z wykorzystaniem wyrażenia CREATE TABLE,
 2. Tworzenie więzów pomiędzy tabelami z wykorzystaniem: NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY.
12. Tworzenie pozostałych obiektów schematu bazy danych
 1. Widoki,
 2. Sekwencje,
 3. Indeksy,
 4. Synonimy.
13. Uprawnienia do obiektów bazy danych
 1. Tworzenie i kontrola uprawnień użytkowników,
 2. Wykorzystanie roli do kontroli praw dostępu.
14. Zarządzanie obiektami w schemacie bazy danych
 1. Modyfikacja tabel, więzów, indeksów.
15. Zarządzanie obiektami ze widokami słownika danych
16. Operowanie na dużych zestawach danych
17. Zarządzanie danymi w różnych strefach czasowych
18. Zawansowane tworzenie zapytań z wykorzystaniem podzapytań.
19. Wyrażenie regularne w Oracle.

Języki i metody programowania - cz. 2

1. Dziedziczenie, klasy abstrakcyjne, interfejsy, polimorfizm
2. Klasy generyczne i kolekcje
3. Wyjątki
4. Operacje wejścia/wyjścia

Aplikacje korporacyjne

1. Główne cechy oprogramowania korporacyjnego
2. Pojęcie architektury aplikacji i stylu architektonicznego, najważniejsze style architektoniczne
3. Budowanie aplikacji z wykorzystaniem podziału warstwowego i komponentowego
4. Technologia Java EE – najważniejsze zagadnienia
5. Budowanie aplikacji z wykorzystaniem Spring Framework
6. Utrwalanie danych z wykorzystaniem Hibernate
7. Współczesne technologie graficznego interfejsu użytkownika
8. Architektura SOA
9. Technologie usług sieciowych, protokół SOAP, styl REST
10. Komunikacja za pomocą kolejek
11. Wprowadzenie do złożonych technik przetwarzania – model aktorowy
12. Budowanie aplikacji korporacyjnej w środowisku Java EE z wykorzystaniem podejścia Domain-Driven Design

Studia trwają 2 semestry, umożliwiają uzyskanie 30 punktów ECTS. Zajęcia realizowane są w formie mieszanej, 96h zajęć prowadzonych jest w formie stacjonarnej. Zajęcia odbywają się średnio co 2 tygodnie w soboty i niedziele, średnio 6 - 8 godzin dziennie (godzina dydaktyczna - 45 minut).

Absolwent studiów podyplomowych uzyskuje świadectwo ukończenia studiów podyplomowych.

Harmonogram

Liczba przedmiotów/zajęć: 0

Przedmiot / temat zajęć	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin	Forma stacjonarna
Brak wyników.					

Cennik

Cennik

Rodzaj ceny	Cena
Koszt przypadający na 1 uczestnika brutto	7 200,00 PLN
Koszt przypadający na 1 uczestnika netto	7 200,00 PLN
Koszt osobogodziny brutto	32,73 PLN
Koszt osobogodziny netto	32,73 PLN

Prowadzący

Liczba prowadzących: 0

Brak wyników.

Informacje dodatkowe

Informacje o materiałach dla uczestników usługi

Zapewniamy uczestnikom studiów dostęp do materiałów przekazywanych przez wykładowców poszczególnych przedmiotów drogą elektroniczną oraz na platformie Moodle. Słuchacze otrzymują: prezentacje przygotowane przez wykładowców, skrypty, inne materiały opisowe przygotowane przez wykładowców, zestawy ćwiczeń.

Warunki uczestnictwa

Osoby z wykształceniem wyższym (I lub II stopnia). Rejestracja <https://podyplomowe.wsiz.pl/rekrutacja/>

Rejestracja na studia podyplomowe odbywa się w formie elektronicznej. Aby zarezerwować miejsce na studiach podyplomowych konieczne jest złożenie kompletu wymaganych dokumentów rekrutacyjnych. Zgłoszenie na studia tylko przez Bazę Usług Rozwojowych nie gwarantuje miejsca w grupie.

Informacje dodatkowe

Zajęcia odbywają się w soboty-niedziele co 2 tygodnie po około 6-8h lekcyjnych każdego dnia w formie stacjonarnej i zdalnej. Zajęcia zdalne realizowane są z wykorzystaniem platformy Cisco Webex. Czesne za studia wpisane w karcie usługi nie obejmuje opłaty rekrutacyjnej w wysokości 50 zł. Opłatę rekrutacyjną należy wnieść w chwili rejestracji na studia przez system rekrutacyjny uczelni.

Szczegółowy harmonogram zajęć dydaktycznych zostanie wprowadzony do karty usługi co najmniej 2 tygodnie przed terminem rozpoczęcia usługi.

Usługa skierowana do Uczestników Projektu MP.

Warunki techniczne

Zajęcia zdalne prowadzone są z użyciem platformy Cisco Webex. Słuchacz loguje się do platformy Cisco Webex ze swojego konta w Wirtualnej Uczelni. Słuchacz, aby skorzystać z zajęć online musi posiadać stanowisko pracy spełniające poniższe minimalne wymagania:

Komputer/laptop/ z zainstalowanym systemem:

Windows

- Windows 10 lub nowszym

Mac OS

- 10.15 lub nowszym

Urządzenia mobilne:

iOS

- 16 i nowsze

iPadOS

- 16 i nowsze

Android

- 10 i nowsze

Minimalna przepustowość połączenia internetowego:

- Download 4 Mb/s
- Upload 4 MB/s

Niezbędne oprogramowanie umożliwiające uczestnikom dostęp do prezentowanych treści i materiałów

- Przeglądarka internetowa (według wyboru słuchacza)

Adres

ul. mjr. Henryka Sucharskiego 2

35-225 Rzeszów

woj. podkarpackie

Zajęcia dydaktyczne realizowane będą w siedzibie uczelni przy ul. Sucharskiego 2. Informacje i sprawy bieżące - pokój nr 48 Centrum Studiów Podyplomowych.

Udogodnienia w miejscu realizacji usługi

- Klimatyzacja
- Wi-fi
- Laboratorium komputerowe

Kontakt



Marta Cisek-Babiarz

E-mail mcisek@wsiz.edu.pl

Telefon (+48) 17 8661 517