

Karta przedmiotu

Nazwa i kod przedmiotu	Konteneryzacja oprogramowania (Ćw. laboratoryjne), PG_00193537						
Kierunek studiów	Bioinformatyka (O)						
Data rozpoczęcia studiów	październik 2026 r.	Rok akademicki realizacji przedmiotu			2028/2029		
Poziom kształcenia	I stopnia - licencjackie	Grupa zajęć			Grupa zajęć fakultatywnych		
Forma studiów	stacjonarne	Sposób realizacji			na uczelni		
Rok studiów	3	Język wykładowy			polski		
Semestr studiów	5	Liczba punktów ECTS			3.0		
Profil kształcenia	ogólnoakademicki	Forma zaliczenia			zaliczenie		
Jednostka prowadząca	Rektor -> Międzyuczelniany Wydział Biotechnologii UG i GUMed						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot		mgr Marcel Thiel				
	Prowadzący zajęcia z przedmiotu						
Formy zajęć	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	0.0	0.0	30.0	0.0	0.0	30
	W tym liczba godzin zajęć na odległość: 0.0						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów		Udział w konsultacjach		Praca własna studenta	RAZEM
	Liczba godzin pracy studenta	30		0.0		45.0	75
Cel przedmiotu	Celem przedmiotu jest zapoznanie studentów z technologiami związanymi z konteneryzacją oprogramowania wraz z uwzględnieniem ich praktycznego wykorzystania w dziedzinie bioinformatyki.						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[BIOINL3_W01] Ma wiedzę z zakresu technologii informatycznych, ze szczególnym uwzględnieniem programowania	zapoznanie z terminologią związaną z konteneryzacją oprogramowania - przegląd i omówienie najpopularniejszych narzędzi związanych z konteneryzacją oprogramowania - wykorzystanie platformy Docker: - tworzenie prostych kontenerów - pobieranie istniejących kontenerów z zewnętrznych repozytoriów (np. DockerHub) - tworzenie interakcji/połączeń pomiędzy konkretnymi kontenerami - zarządzanie procesami wewnątrz kontenerów - zarządzanie kontenerami - zapoznanie z terminologią potrzebną do utworzenia plików typu "Dockerfile" - implementacja własnych plików typu "Dockerfile" - uruchamianie własnych narzędzi w utworzonych kontenerach - walidacja oprogramowania	[SW3] opracowanie tekstowe/praca pisemna
	[BIOINL3_U01] Potrafi programować, wykorzystując nowoczesne narzędzia programistyczne, w tym narzędzia dedykowane bioinformatyce	zapoznanie z terminologią związaną z konteneryzacją oprogramowania - przegląd i omówienie najpopularniejszych narzędzi związanych z konteneryzacją oprogramowania - wykorzystanie platformy Docker: - tworzenie prostych kontenerów - pobieranie istniejących kontenerów z zewnętrznych repozytoriów (np. DockerHub) - tworzenie interakcji/połączeń pomiędzy konkretnymi kontenerami - zarządzanie procesami wewnątrz kontenerów - zarządzanie kontenerami - zapoznanie z terminologią potrzebną do utworzenia plików typu "Dockerfile" - implementacja własnych plików typu "Dockerfile" - uruchamianie własnych narzędzi w utworzonych kontenerach - walidacja oprogramowania	[SU3] opracowanie tekstowe/praca pisemna

Treści przedmiotu	<p>zapoznanie z terminologią związaną z konteneryzacją oprogramowania</p> <ul style="list-style-type: none"> - przegląd i omówienie najpopularniejszych narzędzi związanych z konteneryzacją oprogramowania - wykorzystanie platformy Docker: <ul style="list-style-type: none"> - tworzenie prostych kontenerów - pobieranie istniejących kontenerów z zewnętrznych repozytoriów (np. DockerHub) - tworzenie interakcji/połączeń pomiędzy konkretnymi kontenerami - zarządzanie procesami wewnątrz kontenerów - zarządzanie kontenerami - zapoznanie z terminologią potrzebną do utworzenia plików typu "Dockerfile" - implementacja własnych plików typu "Dockerfile" - uruchamianie własnych narzędzi w utworzonych kontenerach - walidacja oprogramowania 											
Wymagania wstępne i dodatkowe	wym. formalne: Zaliczenie z przedmiotu "Informatyka - wstęp"											
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 33%;">Sposób oceniania (składowe)</th> <th style="width: 33%;">Próg zaliczeniowy</th> <th style="width: 33%;">Składowa oceny końcowej</th> </tr> </thead> <tbody> <tr> <td>wejściówki z zajęć</td> <td>51.0%</td> <td>40.0%</td> </tr> <tr> <td>kolokwium</td> <td>51.0%</td> <td>60.0%</td> </tr> </tbody> </table>			Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej	wejściówki z zajęć	51.0%	40.0%	kolokwium	51.0%	60.0%
Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej										
wejściówki z zajęć	51.0%	40.0%										
kolokwium	51.0%	60.0%										
Zalecana lista lektur	<p>Podstawowa lista lektur</p> <p>Uzupełniająca lista lektur</p> <p>Adresy eZasobów</p>	<p>A. Literatura wymagana do ostatecznego zaliczenia zajęć (zdania egzaminu):</p> <p>A.1. wykorzystywana podczas zajęć</p> <ul style="list-style-type: none"> • Dokumentacja platformy Docker: docs.docker.com <p>A.2. studiowana samodzielnie przez studenta</p> <ul style="list-style-type: none"> • Jeff Nickoloff, Docker in Action, 2nd Edition, Manning Publications, 2019 <p>B. Literatura uzupełniająca</p> <ul style="list-style-type: none"> • Sean P. Kane, Docker: Up & Running, 2nd Edition, O'Reilly Media, 2018 • Elton Stoneman, Learn Docker in a Month of Lunches, Manning Publications, 2020 <p>n</p>										
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania	n											
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy											