



Karta przedmiotu

Nazwa i kod przedmiotu	Zaawansowane techniki obiektowe, PG_00048008						
Kierunek studiów	Informatyka						
Data rozpoczęcia studiów	październik 2024 r.	Rok akademicki realizacji przedmiotu			2027/2028		
Poziom kształcenia	I stopnia - inżynierskie	Grupa zajęć			Grupa zajęć fakultatywnych Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademicki		
Forma studiów	stacjonarne	Sposób realizacji			na uczelni		
Rok studiów	4	Język wykładowy			polski		
Semestr studiów	7	Liczba punktów ECTS			2.0		
Profil kształcenia	ogólnoakademicki	Forma zaliczenia			zaliczenie		
Jednostka prowadząca	Wydział Elektroniki, Telekomunikacji i Informatyki -> Katedra Algorytmów i Modelowania Systemów						
Imię i nazwisko wykładowcy (wykładowców)	Opowiedzialny za przedmiot	dr inż. Krzysztof Manuszewski					
	Prowadzący zajęcia z przedmiotu	dr inż. Krzysztof Manuszewski					
Formy zajęć i metody nauczania	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	15.0	0.0	15.0	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	2.0	18.0	50		
Cel przedmiotu	Celem przedmiotu jest zapoznanie studentów z praktycznymi aspektami nowoczesnych metodologii jak TDD/BDD/DDD oraz różnymi narzędziami/praktykami wspierającymi te podejścia.						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[K6_W10] zna i rozumie w zaawansowanym stopniu podstawowe procesy zachodzące w cyklu życia urządzeń, obiektów i systemów technicznych oraz metody wspomaganie procesów i funkcji, specyficzne dla kierunku studiów	Potrafi zarządzać długim projektowym	[SW1] Ocena wiedzy faktograficznej
	[K6_U09] potrafi dokonać krytycznej analizy sposobu funkcjonowania istniejących rozwiązań technicznych związanych z kierunkiem studiów i ocenić te rozwiązania, a także wykorzystać zdobyte w środowisku zajmującym się zawodowo działalnością inżynierską doświadczenie związane z utrzymaniem urządzeń, obiektów i systemów technicznych typowych dla kierunku studiów	Potrafi praktycznie stosować techniki charakterystyczne dla nowoczesnych metodyk np. UT, refaktoryzację w celu poprawy rozwiązań implementacyjnych.	[SU1] Ocena realizacji zadania [SU4] Ocena umiejętności korzystania z metod i narzędzi
	[K6_W03] zna i rozumie w zaawansowanym stopniu budowę i zasady działania komponentów i systemów związanych z kierunkiem studiów, w tym teorie, metody i złożone zależności między nimi oraz wybrane zagadnienia szczegółowe – właściwe dla programu kształcenia	Potrafi i ocenić różne podejścia dla uzyskania pożądanych efektów w kodzie	[SW1] Ocena wiedzy faktograficznej [SW3] Ocena wiedzy zawartej w opracowaniu tekstowym i projektowym
[K6_U04] potrafi wykorzystywać posiadaną wiedzę z zakresu metod i technik programowania oraz dobrać i zastosować właściwe metody i narzędzia programistyczne w tworzeniu oprogramowania komputerów albo programowania urządzeń lub sterowników wykorzystujących mikroprocesory albo elementy lub układy programowalne, charakterystycznych dla danego kierunku studiów	Potrafi efektywnie stosować narzędzia i techniki w celu refaktoryzacji kodu	[SU4] Ocena umiejętności korzystania z metod i narzędzi	
Treści przedmiotu	<ul style="list-style-type: none"> <li>• Dług projektowy, Refaktoryzacja i podejścia TDD/BDD</li> <li>• Elementy DDD, AoP</li> <li>• Konstrukcja systemów obiektowych</li> <li>• Zasady SOLID w praktyce</li> <li>• Rola wzorców projektowych</li> <li>• Rodzaje wzorców projektowych</li> </ul>		
Wymagania wstępne i dodatkowe	Znajomość wysoko poziomowych obiektowych języków programowania. Preferowany C#		
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	Laboratorium	50.0%	60.0%
	Kolokwium	50.0%	20.0%
	Opracowanie pisemne	50.0%	20.0%
Zalecana lista lektur	Podstawowa lista lektur	<ul style="list-style-type: none"> <li>• Czysta architektura : struktura i design oprogramowania : przewodnik dla profesjonalistów, Robert C. Martin, Helion 2018</li> <li>•</li> <li>• Adaptacyjny kod zwinne programowanie, wzorce projektowe i SOLID-ne zasady, Gary McLean Hall, Helion 2018.</li> <li>• R.Martin. Czysty kod, Helion 2014</li> <li>• Wzorce projektowe, Eric Freeman, Elisabeth Freeman, Bert Bates, Kathy Sierra, Helion 2017</li> </ul>	

	Uzupełniająca lista lektur	Pragmatyczny programista : od czeladnika do mistrza, Andrew Hunt, David Thomas, Helion 2017  Oprogramowanie łatwe w utrzymaniu : pisz kod podatny na przyszłe zmiany, Joost Visser,  Helion. 2017  Wzorce projektowe, E. Gamma, R. Helm, R. Johnson, J.M. Vlissides., WNT 2005
	Adresy eZasobów	Adresy na platformie eNauczanie:
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania		
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy	

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.