

AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

KARTA OPISU PRZEDMIOTU

Wydział		Informatyki	
Kierunek		Informatyka	
Specjalność		Zarządzanie projektami IT	
Semestr	IV	Program studiów, dla którego obowiązuje sylabus	2025/2026
Stopień studiów	I		

Nazwa przedmiotu	Inżynieria oprogramowania w PM			
Kod przedmiotu	IOWPM			
Łączna liczba godzin	18	Tryb	stacjonarny	niestacjonarny
Profil kształcenia	Ogólnoakademicki (A)		Praktyczny (P)	
Forma zajęć	laboratorium			
Język przedmiotu	polski			
Liczba punktów ECTS	2			

Prowadzący zajęcia	
Forma prowadzonych zajęć	Laboratorium
Wymiar zajęć	18 h
Stopień (tytuł) naukowy	
Imię	
Nazwisko	

Wymagania wstępne	Znajomość podstaw inżynierii oprogramowania i cyklu życia projektu IT, podstawowa znajomość wybranych języków programowania.
Założenia i cele przedmiotu	Przedmiot ma na celu zaznajomienie studentów z praktycznym zastosowaniem metodyk zwinnych (Agile) w procesie wytwarzania oprogramowania, w tym tworzeniem iteracyjnych planów prac, efektywnym zarządzaniem wymaganiami oraz komunikacją z klientami. Studenci poznają wzorce projektowe, uczą się tworzyć przypadki testowe i przeprowadzać różne rodzaje testów, a także opanują pracę z systemami kontroli wersji i narzędziami ciągłej integracji. Dzięki realizacji przedmiotu absolwenci będą potrafili skutecznie uczestniczyć w projektach informatycznych realizowanych w metodykach zwinnych, dbając o jakość i efektywność tworzonego oprogramowania.
Metody dydaktyczne	<ol style="list-style-type: none"> 1. Prezentacje multimedialne. 2. Pokazy przykładowych rozwiązań problemów. 3. Rozwiązywanie zadań praktycznych.

Efekty uczenia się (odniesienie do charakterystyk poziomów Polskiej Ramy Kwalifikacji)	Odniesienie do efektów dla kierunku	Odniesienie do efektów uczenia się wg Polskiej Ramy
---	--	--

AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

			Kwalifikacji
WIEDZA – absolwent zna i rozumie:	<p>W01. Fundamenty metodyk zwinnych i ich zastosowanie w cyklu życia projektu programistycznego.</p> <p>W02. Metody analizy i dokumentowania wymagań, w tym sposoby tworzenia i zarządzania backlogiem produktu.</p> <p>W03. Rolę systemów kontroli wersji, takich jak Git, oraz ich znaczenie dla współpracy zespołowej i utrzymania jakości kodu.</p> <p>W04. Podstawowe wzorce projektowe i ich wpływ na elastyczność, skalowalność oraz czytelność kodu źródłowego.</p> <p>W05. Zasady przygotowywania i realizacji testów (jednostkowych, integracyjnych) oraz ich znaczenie w zapewnianiu jakości i stabilności oprogramowania.</p>	<p>K_W20 K_W21 K_W22</p>	<p>P6S_WG P6S_WG_INŻ</p>
UMIEJĘTNOŚCI – absolwent potrafi:	<p>U01. Tworzyć iteracyjne plany rozwojowe zgodne z metodykami zwinnymi, uwzględniając priorytety oraz ryzyka.</p> <p>U02. Przeprowadzać wywiady z klientami oraz na podstawie zebranych informacji formułować i priorytetyzować wymagania.</p> <p>U03. Stosować wzorce projektowe podczas implementacji oprogramowania, aby tworzyć kod lepszej jakości i łatwiejszy w utrzymaniu.</p> <p>U04. Opracować przypadki testowe oraz przeprowadzić testy jednostkowe i integracyjne w celu weryfikacji jakości tworzonego oprogramowania.</p> <p>U05. Zarządzać repozytorium kodu (Git), wdrażać procesy ciągłej integracji i automatyzować działania wytwórcze zespołu.</p>	<p>K_U01 K_U02 K_U03 K_U04 K_U09 K_U15 K_U21</p>	<p>P6S_UW P6S_UW_INŻ P6S_UO P6S_KK P6S_UK</p>
KOMPETENCJE SPOŁECZNE – absolwent jest gotów do	<p>K01. Pracy w zespole, przyjmując w nim różne role.</p> <p>K02. Krytycznej oceny możliwości oprogramowania i systemów dostępnych na rynku IT.</p> <p>K03. Ciągłego samokształcenia się w celu dostosowywania się do dynamicznie zmieniających się technologii.</p>	<p>K_K04 K_K05 K_K06</p>	<p>P6S_UO P6S_KR P6S_KK</p>

Lp.	Tematyka zajęć	Liczba godzin
Forma zajęć – laboratorium		

AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

1	Zastosowania metodyk zwinnych (Agile): tworzenie planów iteracji, prowadzenie spotkań Scrum.	2
2	Zarządzanie wymaganiami w praktyce. Symulacje wywiadów z klientami, ćwiczenia z tworzenia backlogu produktu.	4
3	Projektowanie oprogramowania. Implementacja wzorców projektowych w małych projektach.	4
4	Tworzenie przypadków testowych, przeprowadzanie testów jednostkowych i integracyjnych.	4
5	Kontrola wersji i ciągła integracja. Ćwiczenia z systemem kontroli wersji Git, konfiguracja ciągłej integracji. Zaliczenie.	4

Forma i warunki zaliczenia przedmiotu	Wykonanie projektów. Częstkowe prezentacje, zdawanie raportów, obrona projektów.	
Metody weryfikacji efektów uczenia się		Nr efektu uczenia się z sylabusu
	Ocena projektów i częściowych prezentacji.	W01-W05, U01-U05, K01-K03

Literatura podstawowa	<ol style="list-style-type: none"> 1. J. Roszkowski, <i>Analiza i projektowanie strukturalne</i>, Helion, Gliwice, 2004. 2. I. Sommerville, <i>Inżynieria oprogramowania</i>, WNT, Warszawa 2020. 3. M. Śmiałek, K. Rybiński, <i>Inżynieria oprogramowania w praktyce. Od wymagań do kodu z językiem UML</i>, Helion, Gliwice 2004. 4. M. Flasiński, <i>Zarządzanie projektami informatycznymi</i>, PWN, Warszawa 2006.
Literatura uzupełniająca	<ol style="list-style-type: none"> 1. <i>Przewodnik PMBOK</i>, red. A. Borawska-Reks, PMI Poland Chapter, Kraków 2022.

Nakład pracy studenta	
	Liczba godzin
Zajęcia dydaktyczne	18
Przygotowanie się do zajęć	9
Studiowanie literatury	9
Udział w konsultacjach	2
Przygotowanie projektu / eseju / prezentacji itp.	22
Przygotowanie się do egzaminu / zaliczenia	-
Inne	-
ŁĄCZNY nakład pracy studenta w godz.	60
Liczba punktów ECTS	2