

# WROCŁAWSKA WYŻSZA SZKOŁA INFORMATYKI STOSOWANEJ

## KARTA OPISU PRZEDMIOTU

<b>Wydział</b>		<b>Informatyki</b>	
<b>Kierunek</b>		<b>Informatyka</b>	
<b>Specjalność</b>		<b>Zarządzanie projektami IT</b>	
<b>Semestr</b>	<b>IV</b>	<b>Program studiów, dla którego obowiązuje sylabus</b>	<b>2023/2024</b>
<b>Stopień studiów</b>	<b>I</b>		

Nazwa przedmiotu	Inżynieria oprogramowania w PM			
Kod przedmiotu	IOWPM			
Łączna liczba godzin	30	Tryb	stacjonarny	niestacjonarny
Profil kształcenia	Ogólnoakademicki (A)		Praktyczny (P)	
Forma zajęć	laboratorium			
Język przedmiotu	polski			
Liczba punktów ECTS	2			

<b>Prowadzący zajęcia</b>	
<b>Forma prowadzonych zajęć</b>	<b>Laboratorium</b>
<b>Wymiar zajęć</b>	<b>30 h</b>
<b>Stopień (tytuł) naukowy</b>	
<b>Imię</b>	
<b>Nazwisko</b>	

<b>Wymagania wstępne</b>	Znajomość podstaw inżynierii oprogramowania i cyklu życia projektu IT, podstawowa znajomość wybranych języków programowania.
<b>Założenia i cele przedmiotu</b>	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.
<b>Metody dydaktyczne</b>	<ol style="list-style-type: none"> <li>1. Prezentacje multimedialne.</li> <li>2. Pokazy przykładowych rozwiązań problemów.</li> <li>3. Rozwiązywanie zadań praktycznych.</li> </ol>

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

**WROCŁAWSKA WYŻSZA SZKOŁA INFORMATYKI STOSOWANEJ**

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

Lp.	Tematyka zajęć	Liczba godzin
<b>Forma zajęć – laboratorium</b>		

## WROCŁAWSKA WYŻSZA SZKOŁA INFORMATYKI STOSOWANEJ

1	Zastosowania metodyk zwinnych (Agile): tworzenie planów iteracji, prowadzenie spotkań Scrum.	6
2	Zarządzanie wymaganiami w praktyce. Symulacje wywiadów z klientami, ćwiczenia z tworzenia backlogu produktu.	6
3	Projektowanie oprogramowania. Implementacja wzorców projektowych w małych projektach.	6
4	Tworzenie przypadków testowych, przeprowadzanie testów jednostkowych i integracyjnych.	6
5	Kontrola wersji i ciągła integracja. Ćwiczenia z systemem kontroli wersji Git, konfiguracja ciągłej integracji. Zaliczenie.	6

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

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

Nakład pracy studenta	
	Liczba godzin
Zajęcia dydaktyczne	30
Przygotowanie się do zajęć	5
Studiowanie literatury	5
Udział w konsultacjach	2
Przygotowanie projektu / eseju / prezentacji itp.	18
Przygotowanie się do egzaminu / zaliczenia	-
Inne	-
<b>ŁĄCZNY nakład pracy studenta w godz.</b>	<b>60</b>
<b>Liczba punktów ECTS</b>	<b>2</b>