

KARTA PRZEDMIOTU**I. Dane podstawowe**

Nazwa przedmiotu	Practice of programming
Nazwa przedmiotu w języku angielskim	
Kierunek studiów	Informatyka
Poziom studiów (I, II, jednolite magisterskie)	II stopnia
Forma studiów (stacjonarne, niestacjonarne)	Stacjonarne
Dyscyplina	Informatyka, informatyka techniczna i telekomunikacja
Język wykładowy	Angielski

Koordinator przedmiotu/osoba odpowiedzialna	mgr inż. Kamil Zieliński
---------------------------------------------	--------------------------

Forma zajęć (<i>katalog zamknięty ze słownika</i>)	Liczba godzin	Semestr	Punkty ECTS
wykład			3
konwersatorium			
ćwiczenia			
laboratorium	30	6	
warsztaty			
seminarium			
proseminarium			
lektorat			
praktyki			
zajęcia terenowe			
pracownia dyplomowa			
translatorium			
wizyta studyjna			

Wymagania wstępne	Fundamentals of algorithms and programming Object-oriented programming
-------------------	---------------------------------------------------------------------------

II. Cele kształcenia dla przedmiotu

Getting to know SCRUM and meaning SCRUM in software development process
Recognizing methodology for working with IT development projects
Getting to know tools for team work

III. Efekty uczenia się dla przedmiotu wraz z odniesieniem do efektów kierunkowych

Symbol	Opis efektu przedmiotowego	Odniesienie do efektu kierunkowego
WIEDZA		
W_01	Students have knowledge of software development with Scrum	K_W01, K_W04, K_W06
W_02	Students have knowledge of rules of teamwork	K_W04, K_W06
W_03	Students have knowledge of tools for software development in teams	K_W04, K_W06
UMIEJĘTNOŚCI		
U_01	Students are able to use SCRUM in software development process	K_U04, K_U14
U_02	Students are able to make decomposition of IT system	K_U02, K_U04
U_03	Students are able to use tools for teamwork (IDE, Version Control System)	K_U12, K_U13
KOMPETENCJE SPOŁECZNE		
K_01	Students are able to work with team during software development process	K_K02, K_K04

IV. Opis przedmiotu/ treści programowe

<p>Scrum as methodology for software development Tools for teamwork Software implementation in teams Preparing clean code</p>

V. Metody realizacji i weryfikacji efektów uczenia się

Symbol efektu	Metody dydaktyczne <i>(lista wyboru)</i>	Metody weryfikacji <i>(lista wyboru)</i>	Sposoby dokumentacji <i>(lista wyboru)</i>
WIEDZA			
W_01	Guided practice	Preparation / implementation of the project	Project rating card
W_02	Guided practice	Preparation / implementation of the project	Project rating card
W_03	Guided practice	Preparation / implementation of the project	Project rating card
UMIEJĘTNOŚCI			
U_01	Project - based learning	Preparation / implementation of the	Project rating card

		project	
U_02	Project - based learning	Preparation / implementation of the project	Project rating card
U_03	Project - based learning	Preparation / implementation of the project	Project rating card
KOMPETENCJE SPOŁECZNE			
K_01	Discussion, PBL	Preparation / implementation of the project	Project rating card

VI. Kryteria oceny, wagi...

Based on software project implementation.

VII. Obciążenie pracą studenta

Forma aktywności studenta	Liczba godzin
Liczba godzin kontaktowych z nauczycielem	50
Liczba godzin indywidualnej pracy studenta	40

VIII. Literatura

Literatura podstawowa
Sutherland J., Schwaber Ken, Scrum Guide Martin R.C., A Handbook of Agile Software Craftsmanship Martin R.C., The Clean Coder: A Code of Conduct for Professional Programmers
Literatura uzupełniająca
Hunt A., Thomas D., The Pragmatic Programmer: From Journeyman to Master Martin R.C., Agile Software Development, Principles, Patterns, and Practices