



## Subject card

Subject name and code	Advanced Object-oriented Techniques, PG_00048008						
Field of study	Informatics						
Date of commencement of studies	October 2023	Academic year of realisation of subject			2026/2027		
Education level	first-cycle studies	Subject group			Optional subject group Subject group related to scientific research in the field of study		
Mode of study	Full-time studies	Mode of delivery			at the university		
Year of study	4	Language of instruction			Polish		
Semester of study	7	ECTS credits			2.0		
Learning profile	general academic profile	Assessment form			assessment		
Conducting unit	Department of Algorithms and Systems Modelling -> Faculty of Electronics, Telecommunications and Informatics						
Name and surname of lecturer (lecturers)	Subject supervisor	dr inż. Krzysztof Manuszewski					
	Teachers	dr inż. Krzysztof Manuszewski					
Lesson types and methods of instruction	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	15.0	0.0	15.0	0.0	0.0	30
	E-learning hours included: 0.0						
Learning activity and number of study hours	Learning activity	Participation in didactic classes included in study plan		Participation in consultation hours		Self-study	SUM
	Number of study hours	30		2.0		18.0	50
Subject objectives	The goal is to make student familiar with practical aspects of modern approaches like TDD/BDD/DDD and various tools/practices that support these approaches						

Learning outcomes	Course outcome	Subject outcome	Method of verification
	[K6_W03] Knows and understands, to an advanced extent, the construction and operating principles of components and systems related to the field of study, including theories, methods and complex relationships between them and selected specific issues - appropriate for the curriculum	Student is able to evaluate different approaches for desired effects in codebase	[SW1] Assessment of factual knowledge [SW3] Assessment of knowledge contained in written work and projects
	[K6_W06] Knows and understands the basic processes occurring in the life cycle of devices, facilities and systems specific to a given field of study.	Student understand importance and limitations of modern methodologies for process of software development. Student is able to evaluate different approaches for desired effects in codebase	[SW1] Assessment of factual knowledge [SW3] Assessment of knowledge contained in written work and projects
	[K6_U04] can apply knowledge of programming methods and techniques as well as select and apply appropriate programming methods and tools in computer software development or programming devices or controllers using microprocessors or programmable elements or systems specific to the field of study	Is able to manage the technical debt	[SU4] Assessment of ability to use methods and tools
	[K6_U09] can carry out a critical analysis of the functioning of existing technical solutions and assess these solutions, as well as apply experience related to the maintenance of technical systems, devices and facilities typical for the field of studies, gained in the professional engineering environment	Student is able to apply various technics typical for modern methodologies (i.e. UT, refactorisation) for purpose of improvement shape of implementation	[SU1] Assessment of task fulfilment [SU4] Assessment of ability to use methods and tools
	[K6_U41] can produce, test or evaluate software using modern programming platforms, tools, languages and paradigms of different levels, as well as use software packages supporting scientific and research processes as well as business decision-making processes and teamwork	Student is able to implement various design patterns	[SU1] Assessment of task fulfilment [SU4] Assessment of ability to use methods and tools
Subject contents	<ul style="list-style-type: none"> <li>• Technical Debt, refactoring and TDD/BDD</li> <li>• Elements of DDD, AoP</li> <li>• Construction of OO systems</li> <li>• Solid principles in practice</li> <li>• Importance of Design patterns</li> <li>• Classification of patterns</li> </ul>		
Prerequisites and co-requisites	Modern, high level object oriented languagess. Prefered C#.		
Assessment methods and criteria	Subject passing criteria	Passing threshold	Percentage of the final grade
	laboratories	50.0%	60.0%
	homework	50.0%	20.0%
	exam	50.0%	20.0%

Recommended reading	Basic literature	<p>Czysta architektura : struktura i design oprogramowania : przewodnik dla profesjonalistów, Robert C. Martin, Helion 2018</p> <p>Adaptywny kod zwinne programowanie, wzorce projektowe i SOLID-ne zasady, Gary McLean Hall, Helion 2018.</p> <p>R.Martin. Czysty kod, Helion 2014</p> <p>Wzorce projektowe, Eric Freeman, Elisabeth Freeman, Bert Bates, Kathy Sierra, Helion 2017</p>
	Supplementary literature	<p>Pragmatyczny programista : od czeladnika do mistrza, Andrew Hunt, David Thomas, Helion 2017</p> <p>Oprogramowanie łatwe w utrzymaniu : pisz kod podatny na przyszłe zmiany, Joost Visser,</p> <p>Helion. 2017</p> <p>Wzorce projektowe, E. Gamma, R. Helm, R. Johnson, J.M. Vlissides., WNT 2005</p>
	eResources addresses	Adresy na platformie eNauczanie:
Example issues/ example questions/ tasks being completed		
Work placement	Not applicable	