GDAŃSK UNIVERSITY OF TECHNOLOGY

Subject card

Subject name and code	Software engineering, PG_00045302		
Field of study	Data Engineering		
Date of commencement of studies	October 2025	Academic year of realisation of subject	2026/2027
Education level	first-cycle studies	Subject group	Obligatory subject group in the field of study 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	2	Language of instruction	English
Semester of study	3	ECTS credits	3.0
Learning profile	general academic profile	Assessment form	exam
Conducting unit	Department Of Software Engineering -> Faculty Of Electronics Telecommunications And Informatics -> Wydziały Politechniki Gdańskiej		
Name and surname of lecturer (lecturers)	Subject supervisor	dr inż. Aleksander Jarzębowicz	
	Teachers	dr inż. Aleksander Jarzębowicz	

Lesson types and methods of instruction

Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
Number of study hours	15.0	0.0	30.0	0.0	0.0	45

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	45	6.0	24.0	75

Subject objectives

The aim of the course is to introduce students to analysis and design as part of overall software project activities and to enable practical learning of UML as a tool for object-oriented analysis and design of IT systems.

Learning outcomes

Course outcome	Subject outcome	Method of verification
[K6_K03] demonstrates the ability to think critically and analytically and integrates knowledge from many disciplines in order to make effective decisions	Student works on a team project dedicated to a chosen topic, using the knowledge from lectures and the available tools (modeling language, Computer Aided Software Engineering system).	[SK1] Assessment of group work skills [SK2] Assessment of progress of work
[K6_W01] identifies conditioning of the processes occurring in the analyzed systems and selects methods for solving them, using the accumulated knowledge and taking into account the mutual relations between the analyzed phenomena	Student is able to design analytical models expressing different perspectives of the considered system (functional, structural, dynamic) and to maintain consistency between such models.	[SW3] Assessment of knowledge contained in written work and projects
[K6_U04] formulates logical solutions to complex or unstructured problems	Student is able to describe an organization acting as a client of an IT projekt, requirements related to the developed IT system and to express requirements using a structured modeling language.	[SU1] Assessment of task fulfilment [SU4] Assessment of ability to use methods and tools

<table>
<tr><td>Subject contents</td><td>1. Introduction
2. Scope and subject of software engineering. Essential motivations and concepts.
3. Areas and processes of software engineering
4. Software development models (software lifecycle models) overview
5. Software development methodologies (outline)
6. Conceptual modelling. Languages for modelling and specification.
7. Use cases
8. Object-oriented analysis using UML
9. Modelling of logical system structure: class diagrams
10. Modelling of system structure: other structural diagrams
11. Modelling system dynamics: sequence and communication diagrams
12. Modelling system dynamics: state and activity diagrams
13. Requirements engineering: requirements elicitation, analysis and validation
14. Requirements engineering: requirements specification
15. Design: system architecture
16. Design: system (high-level) design and class (low-level) design
17. Design: design patterns
18. Design: SOLID principles
19. Scrum method</td></tr>
<tr><td>Prerequisites and co-requisites</td><td></td></tr>
<tr><td>Assessment methods and criteria</td><td><table>
<tr><th>Subject passing criteria</th><th>Passing threshold</th><th>Percentage of the final grade</th></tr>
<tr><td>Exam</td><td>50.0%</td><td>50.0%</td></tr>
<tr><td>Lab</td><td>50.0%</td><td>50.0%</td></tr>
</table></td></tr>
<tr><td>Recommended reading</td><td><table>
<tr><td>Basic literature</td><td>1. Pressman R., Maxim B., Software Engineering: a Practitioners Approach, McGraw-Hill, 9th edition, 2019
2. Booch G., Rumbaugh J., Jacobsen I.: The Unified Modeling Language User Guide (2nd Edition), Addison-Wesley, 2005</td></tr>
<tr><td>Supplementary literature</td><td>1. Sommerville I., Software Engineering, 10th edition, Addison-Wesley, 2015
2. Maciaszek L.: Requirements analysis and system design, Addison-Wesley, 2007
3. Fowler M., Scott K.: UML distilled 3rd ed, Addison-Wesley, 2003
4. McLaughlin B., Pollice G., West D., Head First: Object-Oriented Analysis and Design, O'Reilly Media, 2006</td></tr>
<tr><td>eResources addresses</td><td>Adresy na platformie eNauczanie:</td></tr>
</table></td></tr>
<tr><td>Example issues/ example questions/ tasks being completed</td><td>• Draw a UML diagram (e.g. use case diagram, class diagram, state diagram) reflecting a given description of system requirements.
• Describe a given software development model and discuss its strong and weak aspects.
• Enumerate and describe requirements specification techniques.</td></tr>
<tr><td>Work placement</td><td>Not applicable</td></tr>
</table>

Document generated electronically. Does not require a seal or signature.