



## Subject card

Subject name and code	Quality and Testing Engineering, PG_00063899						
Field of study	Informatics						
Date of commencement of studies	February 2027		Academic year of realisation of subject		2027/2028		
Education level	second-cycle studies		Subject group		Optional subject group Specialty 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	1		Language of instruction		Polish		
Semester of study	2		ECTS credits		2.0		
Learning profile	general academic profile		Assessment form		assessment		
Conducting unit	Department of Software Engineering -> Faculty of Electronics Telecommunications and Informatics -> Faculties of Gdańsk University of Technology						
Name and surname of lecturer (lecturers)	Subject supervisor		dr inż. Aleksander Jarzębowicz				
	Teachers		dr inż. Aleksander Jarzębowicz mgr inż. Rafał Dobrosielski				
Lesson types	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	15.0	0.0	0.0	15.0	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		4.0		16.0	50
Subject objectives	The goal of this course is to introduce the principles of Quality Engineering according to TMAP Body of Knowledge and to develop skills in proactive quality assurance within the context of DevOps culture and modern cross-functional teams.						

Learning outcomes	Course outcome	Subject outcome	Method of verification
	[K7_U08] while identifying and formulating engineering tasks specifications and solving these tasks, can: - apply analytical, simulation and experimental methods, - notice their systemic and non-technical aspects, - make a preliminary economic assessment of suggested solutions and engineering work	The student employs the Value Proposition Canvas for analyzing a real business case. They use TMAP tools like VOICE (Value, Objectives, Indicators, Confidence, Experience) to define team objectives and quality expectations. By formulating a Quality Engineering and Risk Strategy within the Lean Startup framework, they leverage iterative learning and feedback to increase the likelihood of project success.	[SU1] Assessment of task fulfilment
	[K7_W101] is able to make an in-depth identification of key objects and phenomena related to the field of study, as well as theories that describe them and applicable analytical and design methods	Students can use Value Proposition Canvas to design solutions that address customer needs and VOICE, KANO, and ISO 25010 to prioritize quality attributes, ensuring alignment between business value and technical design.	[SW3] Assessment of knowledge contained in written work and projects
	[K7_W10] knows and understands, to an increased extent, the basic processes occurring in the life cycle of equipment, objects and technical systems, as well as methods of supporting processes and functions, specific to the field of study	Students has a deeper understanding of the fundamental processes throughout the life cycle of technical systems. They will explore advanced methodologies for supporting these processes, particularly within high-performance development environments like DevOps. This knowledge encompasses effective quality engineering practices, continuous integration/continuous delivery (CI/CD) pipelines, and an understanding of how to implement agile methodologies in conjunction with robust testing frameworks.	[SW1] Assessment of factual knowledge
	[K7_U07] can apply advanced methods of process and function support, specific to the field of study	Student is able to apply advanced process and function support methods, focusing on preventive, detective, and corrective quality engineering tools. Key practices include pair programming, Behavior-Driven Development (BDD), Test-Driven Development (TDD), and non-functional testing. The emphasis is on implementing quality measures, such as test design techniques and coverage analysis, all within the context of DevOps culture.	[SU1] Assessment of task fulfilment

Subject contents	<p>Course content – lecture</p> <p><b>Lectures</b></p> <p><b>1. Introduction to Quality Engineering:</b></p> <ul style="list-style-type: none"> <li>Principles of Quality Engineering (TMAP).</li> <li>Shared responsibility for quality among teams.</li> <li>Quality measures: preventive, detective, and corrective.</li> </ul> <p><b>2. Continuous Quality Engineering in DevOps:</b></p> <ul style="list-style-type: none"> <li>Cross-functional teams in Quality Engineering.</li> <li>Dynamics of DevOps culture.</li> <li>Continuous improvement and testing methods.</li> </ul> <p><b>3. CI/CD Pipeline and Automation:</b></p> <ul style="list-style-type: none"> <li>Components of CI/CD in DevOps.</li> <li>Test automation and continuous testing principles.</li> <li>Team vs. business testing in the pipeline.</li> </ul> <p><b>4. Quality Assurance Methodologies:</b></p> <ul style="list-style-type: none"> <li>Testing methodologies: BDD, TDD.</li> <li>Static and dynamic testing techniques.</li> <li>Effective test coverage design techniques.</li> </ul> <p><b>5. Quality Monitoring and Control:</b></p> <ul style="list-style-type: none"> <li>Monitoring quality in DevOps projects.</li> <li>Understanding anomalies and incident reporting.</li> <li>Communication tools for quality status.</li> </ul> <p><b>Semester Assignment: Quality Engineering Strategy Development</b></p> <ul style="list-style-type: none"> <li><b>Goal:</b> <ul style="list-style-type: none"> <li>Apply knowledge of business value design and quality engineering in a practical project setting.</li> </ul> </li> <li><b>Group Work:</b> <ul style="list-style-type: none"> <li>Teams of 4-5 students will collaborate on a quality engineering strategy.</li> </ul> </li> <li><b>Key Activities:</b> <ul style="list-style-type: none"> <li>Design a business value proposition using tools like the Design Value Proposition and VOICE Model.</li> <li>Conduct KANO analysis to understand customer expectations.</li> <li>Develop a Quality Engineering and Testing Strategy with risk analysis.</li> </ul> </li> <li><b>Outcome:</b> <ul style="list-style-type: none"> <li>A comprehensive proposal that includes project goals, quality attributes, metrics, and acceptance criteria.</li> <li>Practical experience in applying quality tools and methodologies.</li> </ul> </li> <li><b>Skills Gained:</b> <ul style="list-style-type: none"> <li>Enhanced ability to integrate business analysis and quality engineering.</li> <li>Experience in prioritizing goals and managing project quality effectively.</li> </ul> </li> </ul>											
Prerequisites and co-requisites	Basics of software engineering and software development models.											
Assessment methods and criteria	<table border="1"> <thead> <tr> <th data-bbox="453 1561 794 1592">Subject passing criteria</th> <th data-bbox="799 1561 1141 1592">Passing threshold</th> <th data-bbox="1145 1561 1485 1592">Percentage of the final grade</th> </tr> </thead> <tbody> <tr> <td data-bbox="453 1599 794 1630">task in the project</td> <td data-bbox="799 1599 1141 1630">50.0%</td> <td data-bbox="1145 1599 1485 1630">50.0%</td> </tr> <tr> <td data-bbox="453 1637 794 1668">test of knowledge</td> <td data-bbox="799 1637 1141 1668">50.0%</td> <td data-bbox="1145 1637 1485 1668">50.0%</td> </tr> </tbody> </table>			Subject passing criteria	Passing threshold	Percentage of the final grade	task in the project	50.0%	50.0%	test of knowledge	50.0%	50.0%
Subject passing criteria	Passing threshold	Percentage of the final grade										
task in the project	50.0%	50.0%										
test of knowledge	50.0%	50.0%										
Recommended reading	<p>Basic literature</p> <p>Supplementary literature</p> <p>eResources addresses</p>	<p>R. Marselis et al. Quality for DevOps Teams, Sogeti, 2020</p> <p>A. Osterwalder et al. "Value Proposition Design: How to Create Products and Services Customers Want", Wiley, 2014.</p> <p>B. Jayaswal, P. Patton "Design for Trustworthy Software: Tools, Techniques, And Methodology of Developing Robust Software", Prentice Hall, 2006</p>										
Example issues/ example questions/ tasks being completed	N/A											

Practical activities within the subject	Not applicable
---	----------------

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