GDAŃSK UNIVERSITY OF TECHNOLOGY

Subject card

| | | | |
|---|---|---|---|
| Subject name and code | Software Engineering, PG_00044135 | | |
| Field of study | Mathematics | | |
| Date of commencement of studies | October 2023 | Academic year of realisation of subject | 2023/2024 |
| Education level | second-cycle studies | Subject group | Optional subject group |
| 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 | 4.0 |
| Learning profile | general academic profile | Assessment form | assessment |
| Conducting unit | Department of Differential Equations and Mathematical Applications -> Faculty of Applied Physics and Mathematics | | |
| Name and surname of lecturer (lecturers) | Subject supervisor | dr hab. Paweł Pilarczyk | |
| | Teachers | dr hab. Paweł Pilarczyk | |

| Lesson types and methods of instruction | Lesson type | Lecture | Tutorial | Laboratory | Project | Seminar | SUM |
|---|---|---|---|---|---|---|---|
| | Number of study hours | 30.0 | 0.0 | 15.0 | 15.0 | 0.0 | 60 |
| | 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 | 60 | 5.0 | 35.0 | 100 |

| Subject objectives | |
|---|---|
| | Becoming familiar with basic methods of software engineering and acquiring the ability to apply these methods in practice. This concerns group work on a large scale IT project at all the stages of software production: beginning with requirements engineering, through requirements analysis, software design, implementation, testing, installation, to the stage of software maintenance. Also getting acquainted with basic issues regarding IT project management, such as quality management. |

## Learning outcomes

| Course outcome | Subject outcome | Method of verification |
|---|---|---|
| [K7_W11] Knows the mathematical foundations of information theory, the theory of algorithms and cryptography and their practical applications, i.a. in programming and computer science. | Knows how to proceed with software design and development. | [SW1] Assessment of factual knowledge<br>[SW3] Assessment of knowledge contained in written work and projects |
| [K7_U13] Understands the mathematical foundations of the analysis of algorithms and computational processes, can construct algorithms with good numerical properties, used to solve typical and unusual mathematical problems. | Can design and create software that satisfies given requirements. | [SU1] Assessment of task fulfilment<br>[SU4] Assessment of ability to use methods and tools |
| [K7_K03] Can work as a team; understands the necessity of systematic work on all projects that are long-term in nature, understands and appreciates the importance of intellectual honesty in one's own activities and the activities of other people; behaves ethically. | Describes selected software development processess (such as waterfall, iterative, agile).<br>Can develop software in a team.<br>Is able to conduct requirements solicitation, requirements analysis, and to design software.<br>Can use UML to model selected aspects of software.<br>Knows and understands ethics code of a software engineer. | [SK3] Assessment of ability to organize work<br>[SK4] Assessment of communication skills, including language correctness<br>[SK5] Assessment of ability to solve problems that arise in practice<br>[SK1] Assessment of group work skills |
| [K7_W08] Knows advanced computation techniques, supporting the work of a mathematician and understand their limitations. | Knows software engineering methods for designing, creating, testing and maintaining software. | [SW2] Assessment of knowledge contained in presentation<br>[SW3] Assessment of knowledge contained in written work and projects |

## Subject contents

Lecture: Introduction to software engineering. Ethics code. Software development processes. Requirements engineering. Structural and object-oriented methods. Using the UML in modelling. Testing software. IT project management, quality management. Software maintenance. Critical systems. Agile Manifesto and agile methods, including Scrum.

Lab and project: Applying the software engineering methods to designing specific software, including preparation of UML diagrams and user stories. Practical development of software prototypes.

## Prerequisites and co-requisites

Basic ability to write computer programs, e.g. in Python, C, or HTML/JavaScript.

## Assessment methods and criteria

| Subject passing criteria | Passing threshold | Percentage of the final grade |
|---|---|---|
| quizzes (in writing, 10 minutes each) | 60.0% | 50.0% |
| homework, project, presentations, etc. | 60.0% | 50.0% |

## Recommended reading

| | |
|---|---|
| Basic literature | Krzysztof Sacha, Inżynieria oprogramowania, PWN 2010.<br><br>Mariusz Chrapko, Scrum. O zwinnym zarządzaniu projektami. Wydanie II rozszerzone, Helion 2014. |
| Supplementary literature | Ian Sommerville, Software Engineering, Pearson, 10th edition, 2015.<br><br>Roger S. Pressman, Software Engineering: A Practitioner's Approach, McGraw Hill; 8th edition, 2014.<br><br>Russ Miles, Kim Hamilton, Learning UML 2.0: A Pragmatic Introduction to UML, O'Reilly and Associates, 2006.<br><br>Kenneth S. Rubin. Essential Scrum: A Practical Guide to the Most Popular Agile Process. Pearson Education, 2013. |
| eResources addresses | Adresy na platformie eNauczanie:<br>Inżynieria oprogramowania 2024 - Moodle ID: 31164<br>https://enauczanie.pg.edu.pl/moodle/course/view.php?id=31164 |

| | |
|---|---|
| Example issues/ example questions/ tasks being completed | Requirements engineering.<br><br>Software development processes.<br><br>Verification and validation.<br><br>Agile methods and the Manifesto.<br><br>Ethics code of a software engineer.<br><br>The INVEST features of user stories. |
| Work placement | Not applicable |