STUDY MODULE DESCRIPTION FORM

Name of the module/subject	Code
Quality Management and Experimental Software Eng.	**1010512321010517900**

Field of study	Profile of study (general academic, practical)	Year /Semester
Computing	**general academic**	**1 / 2**
Elective path/specialty	Subject offered in:	Course (compulsory, elective)
Software Engineering	**English**	**obligatory**

Cycle of study:	Form of study (full-time,part-time)
Second-cycle studies	**full-time**

No. of hours					No. of credits
Lecture: **-**	Classes: **30**	Laboratory: **-**	Project/seminars: **30**		**4**

Status of the course in the study program (Basic, major, other)	(university-wide, from another field)
major	**from field**

Education areas and fields of science and art	ECTS distribution (number and %)
technical sciences	**4 100%**

Responsible for subject / lecturer:

Mirosław Ochodek, PhD
email: Miroslaw.Ochodek@put.poznan.pl
tel. 61 6652944
Institute of Computing Science
Piotrowo 2 Str., 60-965 Poznan

Prerequisites in terms of knowledge, skills and social competencies:

1	**Knowledge**	Student starting this module should have a basic knowledge regarding software engineering (K_W4) and basic knowledge in the field of mathematics (K_W1).
2	**Skills**	Should have necessary skills to formulate and test simple statistical hypotheses (K_U12), skills necessary to prepare a short scientific report (K_U3) and skills necessary to acquire information from given sources of information.
3	**Social competencies**	Student should understand the need to extend his/her competences. In addition, in respect to the social skills the student should show attitudes as honesty, responsibility, perseverance, curiosity, creativity, manners, and respect for other people.

Assumptions and objectives of the course:

1. Provide knowledge regarding experimental software engineering, especially related to empirical research methods and their theoretical foundations,
2. Provide knowledge regarding quality management, especially related to quality management systems, assessment of processes maturity and their continuous improvement,
3. Develop students? skills in solving problems related to evaluation of methods, tools, and phenomena in software engineering using empirical methods,
4. Develop students? skills in solving problems related to evaluation and improvement of software development processes within an organization.

Study outcomes and reference to the educational results for a field of study

Knowledge:

1. has detailed knowledge in selected areas of statistics: measurement scales, testing hypotheses, power analysis, descriptive statistics and analysis of probability distributions. - [K_W3]
2. has well-established theoretical knowledge of software engineering regarding experimental software engineering and quality management. - [K_W4]
3. has detailed theoretical knowledge related to the selected areas of computer science including: experimental evaluation of tools and methods used for software development. - [K_W6]
4. has basic knowledge related to quality management, including basic knowledge of ISO 9000 standards. - [K_W12]

Skills:

1. is able to acquire, combine, interpret and evaluate information from literature, databases and other information sources (in mother tongue and English); draw conclusions, and formulate opinions based on it. - [K_U1]
2. is able to prepare a treatise in mother tongue and a short research report in English. - [K_U3]
3. is able to prepare and give an oral presentation in mother tongue and English regarding results of empirical research. - [K_U4]
4. has language skills at B2+ level in accordance with the requirements set out for level B2+ Common European Framework of Reference for Languages - [K_U6]
5. is able to plan and conduct experiments and is able to analyze and interpret their outcomes. - [K_U8]
6. is able to employ experimental methods to formulate and solve engineering tasks and basic research problems. - [K_U9]
7. is able to formulate and test hypotheses regarding engineering problems and basic research problems. - [K_U12]
8. is able to experimentally assess usefulness and possibility of employing new developments (methods and tools) and new IT products. - [K_U13]
9. is able to experimentally evaluate usefulness of methods and tools (also to identify their limitations) used to solve engineering tasks, i.e., building IT systems or their components. - [K_U24]
Social competencies:
1. knows examples and understands the causes of the failures of IT systems that have led to major financial or social losses, or caused damage to health or even death. - [K_K4]
2. is aware of the social role of technical university graduates, and especially understands the need of informing the society (especially through mass-media) about new developments in engineering and other aspects of engineering activity; endeavors to disseminate such information, providing also rationale for various viewpoints, in a commonly understandable manner. - [K_K9]
3. Understands the importance of reporting threats to validity of empirical study together with its results. - [K_K9]

Assessment methods of study outcomes
Formative assessment: - based on the answers provided by students during the seminars, - based on the regular assessment of the current status of research projects. Summative assessment: - knowledge regarding experimental software engineering and quality management is assessed based on the results of a test including multiple choice questions (with the possibility of more than one correct answer) and problem solving tasks. Each topic covered within the course is represented by a single question / task. Each task is worth two times more points than a multiple-choice question. Multiple choice questions are graded in the following way (max - maximum points for a question; T - the number of correct answers; F - the number of incorrect answers): for each correctly marked answer student receives max/T points; for each wrongly marked answer student looses 1.5*max/F points; the total number of points for a question cannot be lesser than zero. In order to pass, a student needs to obtain a minimum of 50% of points. (K_W3, K_W4, K_W5, K_W12, K_K4) - skills are verified based on the evaluation of the progress of empirical research project together with the assessment of the final report summarizing the project (K_U1, K_U3, K_U8, K_U9, K_U12, K_U13, K_U24), and based on the oral presentation regarding the project outcomes given during seminars (K_U4, K_U6)..
Course description

The course includes seminars and project classes.

During the seminars students learn and discuss about quality management and experimental engineering. A part of the seminar classes has a form of showcase or individual tasks that are performed by students. The following topics are covered in details:

- Quality Management
- quality management (definition of quality, importance of quality, cost of quality),
- continuous improvement paradigm (Plan-Do-Check-Act, QIP, Experience Factory, and TQM),
- quality management systems and ISO 9000 (definition of quality management system, the structure of ISO 9001, the quality management principles in ISO 9001),
- assessment of processes maturity based on CMMI and SCAMPI,
- good practices of service management with ITIL,
- Reviews and inspections (definition of review and inspection, inspection and review processes and their results).
- Experimental software engineering
- empirical research in software engineering (the role of empirical research in evaluation of tools and methods used in software engineering; relationships between observations, laws, and theories; formulating hypotheses and research questions; quantitative and qualitative methods),
- measurement scales (definition and properties of nominal, ordinal, interval, and ratio scales; types of measurement errors),
- controlled experiments (the goals of controlled experiments; experiment definition; context selection; formulating experiment hypotheses; variables selection: dependent and independent variables; sampling; experiment assumptions; data collection; data validation with the use of statistical methods; data visualization and analysis of probability distributions; testing hypotheses with statistical tests; power analysis; software tools supporting the analysis of the experiment data; interpretation and analysis of the results; classification of threats to validity),
- case studies (goals of cases studies; planning a case study; data and evidence collection; analysis of the data collecting within case study; reporting the results of case study),
- meta-analysis (the goals of meta-analysis; systematic literature reviews; planning and preparation of the review protocol; performing a review; documenting results of review),
- surveys (goals of surveys; types of surveys; preparing surveys; evaluating a survey instrument; the analysis of survey results),

During the project classes? student is running a research project that has to employ at least one of the following empirical methods: controlled experiment, case study, survey, or meta-analysis. The goal of each project is to find an answer to a given research question. The results of the project are described in a report and they presented during the seminars.

Learning methods:

1. The seminars: multimedia presentation, discussion, task solving, and showcase.
2. The project classes: performing an experiment, discussion, and teamwork.

Basic bibliography:

1. C. Wohlin, P. Runeson, M. Host, M. Ohlsson, B. Regnell, and A. Wesslen: Experimentation in Software Engineering: An Introduction, Kluwer Academic Publishers, 2000.

2. Gordon G. Schulmeyer: Handbook of Software Quality Assurance, ISBN-13: 978-1596931862, Artech House Publishers, 2007.

Additional bibliography:

1. F. Shull, J. Singer, D. Sjoberg Guide to advanced empirical software engineering, Springer Verlag, 2007.

Result of average student's workload

Activity	Time (working hours)
1. participating in seminars: 15 x 2 hours,	30
2. participating in project classes: 15 x 2 hours,	30
3. consulting issues related to the subject of the course.	10
4. preparing assumptions of research project, its execution, preparation of the report, and presentation of the results: 15 x 1 hour,	15
5. studying literature / learning aids (10 pages = 1 hour), 150 pages.	15

Student's workload

Source of workload	hours	ECTS
Total workload	100	4

Contact hours	70	3
Practical activities	85	3