STUDY MODULE DESCRIPTION FORM

Name of the module/subject		Code
Software engineering		**1010331461010330109**

Field of study	Profile of study (general academic, practical)	Year /Semester
Information Engineering	**(brak)**	**3 / 6**

Elective path/specialty	Subject offered in:	Course (compulsory, elective)
Safety of Computer Systems	**polish**	**obligatory**

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

No. of hours				No. of credits
Lecture: **2**	Classes: **-**	Laboratory: **-**	Project/seminars: **1**	**4**

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

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

Responsible for subject / lecturer:

dr hab. inż. Barbara Begier
email: Barbara.Begier@put.poznan.pl
tel. (61) 665-3724
Wydział Elektryczny
ul. Piotrowo 3A 60-965 Poznań

Prerequisites in terms of knowledge, skills and social competencies:

1	**Knowledge**	Student has also theoretical and partially practical knowledge concerning: programming constructions, implementation of algorithms, programming styles, verification of software correctness, formal languages, compilers, and platforms. Knowledge learnt during the first semester of software engineering course.
2	**Skills**	Student is able to find information from professional literature, databases and other sources; he/she can also integrate and correctly interpret the gained information and then to coclude and formulate his/her own opinions. Student can create object models (use cases, objects, and classes) using the UML standard.
3	**Social competencies**	Student is aware of an importance of non-technical aspects and then consequences of software engineer's activities; he/she understands his/her responsibility for his/her decisions. Student understands a need to to constant learning, including improvement of using foreign languages and other professional and social competencies.

Assumptions and objectives of the course:

Quality aspects in software production. Software quality assessment.

Agile methods applied in software development.

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

Knowledge:

1. . Student has basic knowledge concerning methodologies applied in software engineering: software quality characteristics, overview of agile methodologies: XP, TDD, AMDD, FDD, BDD, and Scrum. - [K_W12]

2. Student is knowledgeable with the state of art and modern trends in software engineering and computing - [K_W19]

Skills:

1. Student is able to formulate requiremets, to build an object model, and assess a simple information system, its functions, and components - [K_U16]

2. Student is able to document an implementation of a software prodcut and also to discuss results of his/her engineering task. - [K_U06]

Social competencies:

1. Student is aware of an importance of a precise implementation of a software product, using the design standards, and preparing the correct documentation. - [K_K07]

2. Student is aware of his/her responsibility for the work done. He/she points out his/her readyness to take part in a team work and to be responsible for results of tasks realized in team. - [K_K04]

Assessment methods of study outcomes

The content of lectures presented in the first and second semester of the software engineering course is a subject of an exam.

During the project classes student shows his/her skills in object modeling, including software object behaviour and interactions. Marks are given for each UML diagram separately, and also for the test plan document.

Course description

Specification of software quality and its criteria according to the standards of ISO 9126 and ISO 25010. Software quality policies.

Values and objectives in agile (soft) methodologies. Roles of various stakeholders in the software process. Principles expressed in the Agile Manifesto. Review of agile methodologies: XP (eXtreme Programming), TDD (Test Driven Development), AMDD (Agile Model Driven Development), FDD (Feature Driven Development), BDD (Behavior Driven Development), and Scrum. Human aspects in software development. Software user satisfaction with a software product, EUCS (End User Computing Satisfaction) model.

Project. Development of an object model using the UML standard (modeling of software object behaviours). Development of the test plan and/or the quality tree related to the developed product.

Basic bibliography:

1. Martin R., Martin M., Agile. Programowanie zwinne. Zasady, wzorce i praktyki zwinnego wytwarzania oprogramowania w C?, Helion, Gliwice 2008

2. . Wrycza St., Marcinkowski B., Wyrzykowski K., Język UML 2.0 w modelowaniu systemów informatycznych, Helion, Gliwice 2005.

3. Recommended materials from Internet

Additional bibliography:

1. Begier B., Inżynieria oprogramowania - problematyka jakości, Wydawnictwo Politechniki Pozn., Poznań 1999.

2. Hnatkowska B., Huzar Z., Inżynieria oprogramowania ? metody wytwarzania i wybrane zagadnienia, PWN, Warszawa 2008.

3. Pilone D., Pitman N., UML 2.0 almanach, Helion, Gliwice 2007.

4. Subieta K., Wprowadzenie do inżynierii oprogramowania, Wydawnictwo PJWSTK, Warszawa 2002.

Result of average student's workload

Activity	Time (working hours)
1. Participation in lectures	30
2. Participation in project labs	15
3. Project modeling and design	20
4. Exam, consultations	10
5. Preparation for the exam	25

Student's workload

Source of workload	hours	ECTS
Total workload	100	4
Contact hours	55	2
Practical activities	35	1