



## COURSE DESCRIPTION CARD - SYLLABUS

Course name

Software Engineering [N1IZar1>IO]

### Course

Field of study

Engineering Management

Year/Semester

2/4

Area of study (specialization)

–

Profile of study

general academic

Level of study

first-cycle

Course offered in

Polish

Form of study

part-time

Requirements

compulsory

### Number of hours

Lecture

10

Laboratory classes

0

Other (e.g. online)

0

Tutorials

0

Projects/seminars

14

### Number of credit points

2,00

### Coordinators

dr hab. Grzegorz Pawłowski

grzegorz.pawlowski@put.poznan.pl

### Lecturers

### Prerequisites

Basic course in the design of IT management systems. Efficient use of design support tools (Visio) and database design skills. Understanding the need for design skills and management for the implementation of management information systems.

### Course objective

To familiarize students with the methods and CASEs of software engineering used in the design of information management systems.

### Course-related learning outcomes

Knowledge:

The student explains and compares various methods and tools for collecting, processing, selecting, and distributing information in the context of software engineering [P6S\_WG\_08].

The student describes the software product lifecycle, including stages such as creation, implementation, operation, and modification [P6S\_WG\_15].

The student identifies and classifies basic methods, techniques, tools, and materials used in software engineering, such as tools for requirements management, prototyping, and cost estimation

[P6S\_WG\_16].

The student lists basic principles of safety and hygiene and describes their application in the work environment of a software engineer [P6S\_WG\_18].

Skills:

The student designs and conducts experiments and computer simulations, analyzes, and interprets results in the context of software development and testing [P6S\_UW\_09].

The student applies analytical, simulation, and experimental methods to formulate and solve problems in software engineering, including in the process of requirements engineering [P6S\_UW\_10].

The student takes responsibility for individual and group IT projects, adhering to teamwork principles and project management methodologies, such as the P-CMM model [P6S\_UO\_01].

Social competences:

The student analyzes and evaluates cause-and-effect relationships in the software creation process, making decisions regarding task prioritization and resource management [P6S\_KK\_02].

## Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Lectures: problem tasks to do during the lecture, exam

Project: assessment of current work on the design of the application logical model project, ready documentation of the application logical model

## Programme content

The program covers the following issues: creation, implementation, operation and modification of an IT system, integration of IT systems, functional and non-functional requirements, requirements engineering process, requirements management, software engineering tools, software prototyping using the UML.

## Course topics

Lecture:

1. Analysis of the IT system - stages of design, construction and operation of the IT System, high- and low-level CASE tools, software requirements, requirements management, FURPS scheme, Cascade model
2. Iterative modeling - Evolutionary model, Incremental model, Spiral model, RUP methodology: phases and cycles of software development, Agile methodology: extreme, Scrum, Kanban
3. Software development - structural, object-oriented and generic approach
- 4-7. UML language course - history of object-oriented modeling, characteristics of the UML language, modeling of business and IT systems, Model dynamics and structure diagrams, Use case diagram (4), Activity diagram (5), Sequence diagram (6), Class and object diagram (7)

Project:

Initial modeling of the requirements of the selected software using a brainstorming diagram (e.g. FURPS diagram) and creation of a set of UML diagrams including: use case diagram, activity diagram, sequence diagram, class diagram.

## Teaching methods

Lecture - informative lecture, seminar, case study

Laboratories - laboratory method, project method, brainstorming, demonstration method

## Bibliography

Basic:

Borucki A. (2012). E-Biznes. Wydawnictwo Politechniki Poznańskiej. Poznań.

Kolbusz E., Olejniczak W., Szyjewski Z. (2005). Inżynieria systemów informatycznych w e-gospodarce. PWE. Warszawa.

Sommerville I. (2003). Inżynieria oprogramowania. WNT. Warszawa.

Jaszkiewicz A. (1997). Inżynieria oprogramowania. Helion. Gliwice.

Additional:

Szpringer W. (2012). Innowacyjne modele e-biznesu. Difin. Warszawa.  
Flasiński M.(2008). Zarządzanie projektami informatycznymi.PWN

### Breakdown of average student's workload

	Hours	ECTS
Total workload	50	2,00
Classes requiring direct contact with the teacher	28	1,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	22	1,00