

Study programme(s): Informatics (IM)
Level: master
Course title: Requirements Engineering (IB121)
Lecturer: Mirjana K. Ivanović, Danijela N. Boberić-Krstićev
Status: obligatory for the <i>Software engineering</i> module; elective for other modules.
ECTS: 7.5
Requirements: none
Learning objectives Software requirements engineering is the science and discipline concerned with the process of forming and documenting software requirements. It deals with the techniques for elicitation, analysis, specification, verification and management of software requirements. On the other hand, system requirements engineering is the science and discipline concerned with the analysis and documentation of system requirements. It involves a transformation of operational needs into a system description, system performance parameters and a system configuration, that is accomplished through an iterative process of analysis, design, trade-off studies and prototyping. This course is intended to provide students with a comprehensive understanding and critical evaluation of software and system requirements engineering.
Learning outcomes <i>Minimal:</i> Students should be able to demonstrate comprehensive understanding of requirements engineering both for software and system requirements, to be able to critically evaluate basic management models and basics of requirements engineering, and to appreciate the essential issues of requirements engineering and architecture design. <i>Optimal:</i> Students should be able to practically apply techniques for elicitation, analysis, documentation, validation and tracing of requirements, and to critically evaluate the role of tools and methods in requirements engineering.
Syllabus <i>Theoretical instruction</i> Theoretical basis and methods of requirements engineering. Types and characteristics of requirements. Requirements structuring and their analysis. Relationships and distinctions among requirements in design process, their role and implementation in real software systems. Requirements quality. Relationships among users and requirements engineers. Techniques for requirements elicitation, analysis and negotiation. System theory, system engineering and concepts of system requirements. Transformation of operational requirements into technical specifications. Techniques for requirements documentation. Requirements management and tracing. Usage of appropriate formalisms and notations. Illustration of adequate requirements management tools. <i>Practical instruction</i> Analysis of study examples and individual work on requirements specifications for small real-world systems.
Literature 1. R. H. Thayer and M. Dorfman (ed.), <i>Software Requirement Engineering</i>, IEEE CS, 2000. 2. G. Kotonya and I. Sommerville, <i>Requirements Engineering: Processes and Techniques</i>, John Wiley & Sons, 2000. 3. I. Sommerville and P. Sawyer, <i>Requirements Engineering: A Good Practice Guide</i>, John Wiley & Sons, 1997. 4. Loucopoulos, P., and Karakostas, V., <i>System Requirements Engineering</i>, McGraw-Hill, 1995.

5. Derek J. Hatley et al., *Process of Systems Architecture and Requirements Engineering*, Dorset House, 2000.

Weekly teaching load				Other:
Lectures: 3	Exercises: 2 (1+1)	Other forms of teaching:	Student research:	
Teaching methodology Lectures are held using classical methods involving power-point presentations and a video-beam. During exercises, classical teaching methods involving power-point presentations and a video-beam are used to analyze study examples. Students complement their knowledge through supplementary research of materials presented, and are tested through preparation of seminar papers that are defended during and at the end of the course.				
Grading (maximum number of points 100)				
Pre-exam obligations	points	Final exam	points	
Active participation in lectures	6	Paper presentation	40	
Practical instruction	6			
Seminars	48			