

<b>Study Programme:</b> Information Technology - Software Engineering		
<b>Course Unit Title:</b> Software life cycle management		
<b>Course Unit Code:</b> OAS295		
<b>Name of Lecturer(s):</b> Associate Professor Zeljko Stojanov, PhD		
<b>Type and Level of Studies:</b> Bachelor Academic Degree		
<b>Course Status (compulsory/elective):</b> Compulsory		
<b>Semester (winter/summer):</b> Summer		
<b>Language of instruction:</b> English		
<b>Mode of course unit delivery (face-to-face/distance learning):</b> Face-to-face		
<b>Number of ECTS Allocated:</b> 6		
<b>Prerequisites:</b> None		
<b>Course Aims:</b> Acquisition of basic knowledge about software life cycle, as well as about basic principles of software engineering that relate to specific phases in software life cycle.		
<b>Learning Outcomes:</b> The acquired knowledge will enable students to understand the following aspects of software life cycle: life cycle models and phases, human roles in software life cycle, application of standards and recommendations of good practice, life cycle management, requirements management, software development management, software delivery management, software deployment, software maintenance and evolution, quality assurance, and process assessment and improvement. Students will be trained for practical independent and teamwork, as well as for solving problems.		
<b>Syllabus:</b> <i>Theory</i> Software life cycle. Software processes. Life cycle models and standards. Human factor - roles, teams, users. Standards and recommendations. Software change and configuration management, version control. Requirements management - collection, specification, prioritization, verification and validation. Software development - analysis, design, construction and testing. Software delivery - install, deploy and configure. Software maintenance - basics, processes and activities, techniques. Software evolution. Quality assurance - basic principles, ethical principles in software engineering, models and characteristics, quality improvement, quality measurement, security and reliability. Process assessment and improvement, metrics and measurement. <i>Practice</i> Mastering the basic principles and techniques for managing individual activities and phases in software life cycle through practical examples realized within computer laboratory exercises and seminar work.		
<b>Required Reading:</b> 1. Pierre Bourque and Richard E. (Dick) Fairley (Editors). Guide to the Software Engineering Body of Knowledge, Version 3.0, SWEBOOK. IEEE. 2014. 2. Ian Sommerville. Software Engineering, 9th edition. Addison-Wesley, Boston, MA, USA. 2011. 3. Hans van Vliet. Software Engineering: Principles and Practice, 3rd edition. John Wiley & Sons. Chichester, England. 2008.		
<b>Weekly Contact Hours:</b> 4	<b>Lectures:</b> 2	<b>Practical work:</b> 2

**Teaching Methods:**

Lectures. Illustrative teaching methods. Demonstration teaching methods. Case studies. Research.

**Knowledge Assessment (maximum of 100 points): 100**

<b>Pre-exam obligations</b>	points	<b>Final exam</b>	points
Active class participation	10	oral exam	30
Test I	30		
Seminar	30		