

## Specification for the book of courses

<b>Study program</b>		Electrical Engineering and Computer Science		
<b>Module</b>		Computing and Informatics		
<b>Type and level of studies</b>		Undergraduate Academic Studies		
<b>The name of the course</b>		Software Architecture and Design		
<b>Lecturer (for lectures)</b>		Stojanović H. Dragan, Predić B. Bratislav		
<b>Lecturer/associate (for exercises)</b>		Predić B. Bratislav, Mihajlović T. Vladan		
<b>Lecturer/associate (for OFE)</b>		Mihajlović T. Vladan		
<b>Number of ECTS</b>	5	<b>Course status (obligatory/elective)</b>	Elective	
<b>Prerequisites</b>				
<b>Course objectives</b>	Acquiring knowledge required for architecture development and design of large scale software systems using architectural styles and patterns, design patterns, software components, application frameworks and services.			
<b>Course outcomes</b>	Theoretical and practical knowledge about principles, methods and software tools for architecture development and design of large scale software systems.			
<b>Course outline</b>				
<b>Theoretical teaching</b>	Introduction to software architecture and design. Fundamental principles and methods of software architecture. Modularity of a software system, module coupling and cohesion. Quality attributes and software architecture. Architectural styles and patterns. Software components, middleware and application frameworks. Design, documenting and evaluation of software architecture. Service-oriented architecture and Web services. Model Driven Architecture (MDA) and software development. Basic principles and methods of software design. Software design using design patterns. Software design anti-patterns. Refactoring to design patterns. Software design and architecture patterns in contemporary information systems.			
<b>Practical teaching (exercises, OFE, study and research)</b>	Practical work on examples of software design using design patterns and refactorig of existing design to design patterns. Consideration of various architecture styles and patterns in large scale software system architecture. Specification of non-functional requirements and quality requirements and architecture design that satisfies these requirements. Development and documenting architecture and design of a real software system using appropriate styles, design patterns and UML diagrams.			
<b>Textbooks/references</b>				
1	Len Bass, Paul Clements Rick Kazman. Software Architecture in Practice, 3rd edition, Addison-Wesley Professional., 2012			
2	Ian Gorton, Essential Software Architecture, 2nd Edition, Springer, 2011.			
3	Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides. Design, Patterns - Elements of Reusable Object-Oriented Software. Addison-Wesley, 1995.			
4				
5				
<b>Number of classes of active education per week during semester/trimester/year</b>				
<b>Lectures</b>	<b>Exercises</b>	<b>OFE</b>	<b>Study and research work</b>	<b>Other classes</b>
2	2	1	0	0
<b>Teaching methods</b>	Lectures, auditive exercises, lab practicing, independent student work on assignments and projects, student seminars.			
<b>Grade (maximum number of points 100)</b>				
<b>Pre-exam duties</b>	<b>Points</b>	<b>Final exam</b>		<b>Points</b>
<b>Activity during lectures</b>		<b>Written exam</b>		40
<b>Exercises</b>		<b>Oral exam</b>		
<b>Colloquia</b>	40			
<b>Projects</b>	20			