

Study Programme: Information Technologies
Course Unit Title: Architecture, Design and Patterns
Course Unit Code: IT402
Name of Lecturer(s): Gordana Rakić
Type and Level of Studies: Bachelor Academic Degree
Course Status (compulsory/elective): Compulsory
Semester (winter/summer): Summer
Language of instruction: Serbian (primary), English (secondary)
Mode of course unit delivery (face-to-face/distance learning): Face-to-face
Number of ECTS Allocated: 7
Prerequisites: Object-Oriented Programming 1
<p>Course Aims:</p> <p>This course aims are to introduce students to a multitude of modelling techniques and designs to address the issue of software architecture in the context of object-oriented software development. Course covers all aspects of software design from architectural features (styles, models and views) to design models that could be described as "a common solution to common problems in a given context" on the lower level of abstraction.</p>
<p>Learning Outcomes:</p> <p><i>Minimal</i></p> <p>At the end of the course, it is expected that the successful student shows clear understanding of the impact of abstraction, modelling, architecture, and patterns in software product development and be able to critically discuss software architectures the key concepts, designs and patterns.</p> <p><i>Desirable</i></p> <p>At the end of the course, it is expected that the successful student is able to critically discuss the architectural alternatives and alternative designs, to generate a reasonable alternative for the problem and select between them, to identify an appropriate pattern for the problem and create it, and to apply practical skills in generating and developing software architecture and design based on functional requirements.</p>
<p>Syllabus:</p> <p><i>Theory</i></p> <p>Theoretical background of software architecture, analogy with architecture in general, the elements of software architecture, architectural styles (ABAS), architectural patterns (Event-based, Layered, Pipes & Filters, ...), architecture description languages, the interaction between requirements and architecture, master-plan vs. piecemeal growths, architecture analysis and evaluation (SAAM, Scenario-based evaluation), the architectural process and organization, model driven architecture, from the architecture to the model, re-usable architecture, design patterns, framework and tools.</p> <p><i>Practice</i></p> <p>Case study analysis</p>
<p>Required Reading:</p> <ol style="list-style-type: none"> 1. Flávio Oquendo, Jair Leite, Thais Batista, Software Architecture in Action Designing and Executing Architectural Models with SysADL Grounded on the OMG SysML Standard, Springer, 2016 2. Len Bass, Rick Kazman, Paul Clements, Software Architecture in Practice, Addison Wesley, second edition.

3. M. Shaw and D. Garlan, Software Architecture. Prentice Hall 1996
4. Erich Gamma, Richard Helm, Ralph Johnson, John M. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison Wesley.
5. 4. H. Rumbaugh, M. Blaha, W. Premarlani, F. Eddy, W. Lorensen: Object-Oriented Modelling and Design, Prentice-Hall
6. G. Booch: Object-Oriented Analysis and Design with Applications, Addison-Wesley, 1994 (2nd ed.)

Weekly Contact Hours: 5	Lectures: 3	Practical work: 2
--------------------------------	--------------------	--------------------------

Teaching Methods:
 During the lectures classical methodology is applied, through usage of beam-projector and slides projector. During the exercises the case studies and examples are analyzed by using the traditional methods of teaching and using the projector. Practical skills in architecture and modelling are developed with the introduction of the recommended tools. Students build their knowledge incrementally in each research topic. Knowledge is checked through the realization of projects that are presented during and at the end of the course.

Knowledge Assessment (maximum of 100 points):

Pre-exam obligations	points	Final exam	points
Active class participation	10	written exam	0
Practical work	60	oral exam	30
Preliminary exam(s)	0	
Seminar(s)			

The methods of knowledge assessment may differ; the table presents only some of the options: written exam, oral exam, project presentation, seminars, etc.