Study program: Information Technology

Course title: CONTEMPORARY SOFTWARE ARCHITECTURES

Teacher(s): Vladimir M. Mladenović

Course status: mandatory

Number of ECTS credits: 6

Prerequisite courses: none

Course objectives

Acquaintance with modern trends in software products and architectures and mastery of modern practical "exit" knowledge (at the exit from the Faculty) and skills in the field of software system architecture required for the realization of complex information systems.

Learning outcomes

They enable the student to effectively understand the structure of software and create software architecture, to be able to combine technologies for software development, to be creative in the part of process development, software architectures (logical and physical) and architecture and system documentation; get to know the styles of creating software architectures, create software architecture for the needs of modern business, with appropriate software components and interfaces.

Content of the course

Theoretical teaching

Models, development processes (methodological approaches) and software architectures. Two-level, three-level and multi-level architectures. Logical and physical software architectures. Software components: Interfaces and classes. Frameworks. Multilevel architectures of software components in a heterogeneous environment. Middleware and general services: Life cycle of objects. Asynchronous and synchronous communication of objects. Object state and persistence. Security requirements. Using Middleware Services and Components. Overview of classic software architectures: OMG CORBA, SUN J2EE, EJB. Problems with classical architectures. OMG MDA architecture. Platform-independent and platform-dependent architectures. Continuous delivery, software delivery problem, configuration management, continuous integration, test strategy implementation, non-functional requirements testing, infrastructure and environment management.

Practical teaching

Exercises, seminar work and homework. During the exercises, the practical part of the mentioned content from the lectures is implemented.

Literature

[1] Vladimir Mladenović, Danijela Milošević, Software architecture - theory and application in practice, Textbook, Faculty of Technical Sciences in Čačak, University of Kragujevac © 2016, ISBN: 978-86-7776-184-4

[2] John Reekie, Rohan McAdam, A Software Architecture, Angophora Press, 2006.

[3] Len Bass, Paul Clements, Rick Kazman, Software Architecture in Practice, 2nd edition, Addison-Wesley Professional, 2006

[4] J. Humble and D. Farley, Continuous Delivery, 2011, Pearson Education

Number of active teaching classes: 4	**Theoretical teaching: 2**	**Practical teaching: 2**

Teaching methods

Realization of lectures and exercises according to the model of interactive teaching (popular lecture, discussion, methods of demonstration, practical work, research, workshops); activated forms of learning: verbal sense receptive learning, cooperative, practical and discovery learning.

Evaluation of knowledge (maximum number of points 100)

Pre-exam obligations	**Points**	**Final exam**	**Points**
Activities during teaching process	10	Final exam (written):	10
Practice teaching	20	Final exam (oral):	20
Colloquium	30		
Project	20		