Study programme(s): Information Technologies

Level: Bachelor

Course title: System Integration

Lecturer: Miloš M. Radovanović

Status: elective

ECTS: 7

Requirements: none

Learning objectives

This course is intended to provide a comprehensive understanding of technical details that participate in design and implementation of modern distributed systems through integration. It also introduces various concepts and approaches for system integration.

Learning outcomes

Minimum: At the end of the course, it is expected that a successful student can demonstrate the ability to assess the needs and benefits of different models and approaches to system integration and implement a simple distributed system which can effectively communicate with existing and new architecture.

Desirable: At the end of the course, it is expected that a successful student can demonstrate the ability to critically evaluate different integration capabilities, design and use of object-based, service-oriented and resource-oriented distributed systems.

Syllabus

Theoretical instruction

Theoretical basis of integration, middleware-oriented integration, Java RMI and CORBA (architecture, interfaces, IDL, clients, services, CORBA vs. RMI and RPC). Introduction to service-oriented architecture (SOAP, WSDL, UDDI). Introduction to resource-oriented architecture. JAX-RS specification. Filters and interpreters. Caching. Asynchronous REST services. REST service security. Web service integration. Semantic-oriented integration (architecture, agent-oriented approach, intelligent systems and agent-service communication).

Practical instruction

Case study analysis.

Literature

Recomended

1 Gregor Hohpe, Bobby Woolf, Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions, Addison-Wesley, 2012

2. George F. Coulouris, Jean Dollimore, Tim Kindberg, Distributed Systems: Concepts And Design, Addison-Wesley, 2005

3. Thomas Erl, Service-oriented architecture: a field guide to integrating XML and Web services, Prentice Hall, 2004

4. Doreen L. Galli, Distributed operating systems: concepts and practice, Prentice Hall, 2000

5. Burke Bill, Restful Java with Jax-RS 2.0: Designing and Developing Distributed Web Services, O'Reilly Media, Inc, 2013

Weekly teaching load

Lectures:	Exercises:	Practical Exercises:	Student research:	Other:
2	1	2	0	0

Teaching methodology

At the lectures, conventional teaching methods are used, with the help of the projector. At the theoretical exercises, conventional teaching methods with the projector are used, case studies are analyzed, practical skills exercised and work with the recommended tools and envioronments introduced. Students improve their knowledge in each of the research topics and check them through practical assignments presented during and at the end of the course.

Grading method (maximal number of points 100)

Pre-exam oblications	**points**	**Final exam**	**points**
Practical instruction	50	Oral exam	50