

Study Programme: Information technology – software engineering
Course Unit Title: Internet programming
Course Unit Code: OAS311
Name of Lecturer(s): Assistant professor Ljubica Kazi, PhD
Type and Level of Studies: Bachelor Academic Degree
Course Status (compulsory/elective): Compulsory
Semester (winter/summer): Winter
Language of instruction: English
Mode of course unit delivery (face-to-face/distance learning): Face-to-face
Number of ECTS Allocated: 6
Prerequisites: None
<p>Course Aims:</p> <p>The main objective of the subject is to provide students with the necessary knowledge to develop Internet applications using a modern web programming languages. The subject includes the display of different methodologies developing web applications with a focus on the multi-layered software architecture, communication with the database and exchange data via web services. Particular aim is to address issues in business software development and business rules enforcement within the multi-layered software architecture.</p>
<p>Learning Outcomes:</p> <p>Students will be able to create web application with use of modern programming languages and technologies (PHP, Java, ASP.NET), organizing the multi-layered software architecture based on working with databases, creating web service (SOAP, REST) and interoperability support file formats such as XML and JSON.</p>
<p>Syllabus:</p> <p><i>Theory</i></p> <p>Information technologies used in the implementation of Internet services and applications. The exchange of data and information over the Internet. Basic features of HTTP transmission protocol and HTML standard for describing Web pages. Understanding the basic structure of the Internet applications, exchange parameters between web page using POST and GET methods. Comparative overview of the technologies for the development of information systems software. Programming languages in the web environment and the basic structure and principles of development of Internet applications. Web internet services and service-oriented architecture. Requirements specification and implementation of solutions. Understanding and implementing MVC pattern, communication with the database and forming a web service. Exchanging data via web services through JSON and XML formats.</p> <p><i>Practice</i></p> <p>Implementation of n-tier web application with using web services.</p>
<p>Required Reading:</p> <ol style="list-style-type: none"> 1. David Garlan: History of software architecture, Carnegie Mellon University, NASA Fault Management Workshop, New Orleans, April 2012. 2. WWW consortium standards, https://www.w3.org/standards 3. Shklar L., Rosen R., Web Application Architecture, John Wiley & Sons Ltd, 2003 4. Paterno F., The Encyclopedia of Human-Computer Interaction, The Encyclopedia of Human-Computer

Interaction, 2nd Ed., Interaction Design Foundation, 2013.

5. B.JALENDER, A.GOVARDHAN, P.PREMCHAND: DESIGNING CODE LEVEL REUSABLE SOFTWARE COMPONENTS, International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012
6. Carl Erickson: OBJECT-ORIENTED PROGRAMMING, Atomic Object LLC, 2009.
7. Marek Potociar: When To Use SOAP And When REST, jAZOON, International conference on the modern art of Software, 2011.
8. Liang Chen, Bruno Wassermann and Wolfgang Emmerich: Web Service Orchestration with BPEL, ICSE'06, May 20–28, 2006, Shanghai, China.
9. OASIS standard, https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss
10. F.B. Vernadat, Enterprise modelling and integration: principles and applications, London: Chapman & Hall, 1996.
11. Adam Freeman: Pro ASP.NET MVC 5, APress
12. Luke Weiling, Laura Thomson: PHP and MySQL Web development, 5th edition, Addison Wesley, 2017.
13. Herbert Schildt, Java 2: Complete reference, Micro book, 2001.

Weekly Contact Hours: 6

Lectures: 3

Practical work: 3

Teaching Methods:

Lectures and students group work

Knowledge Assessment (maximum of 100 points): 100

Pre-exam obligations	points	Final exam	points
Active class participation	10	written exam	20
Seminar work	20	oral exam	30
Practical lab work	20		