

1.	Course Title	DevOps
2.	Code	F18L3S118
3.	Study program	Software engineering and information systems
4.	Study Program Organizer	Faculty of Computer Science and Engineering
5.	Degree (first, second, third cycle)	first cycle
6.	Academic year / semester 3 / summer / optional	7. ECTS credits 6
8.	Teacher	full professor Panche Ribarski, assistant professor Milosh Jovanovikj
9.	Course enrollment prerequisites	Оперативни системи и (Веб програмирање или Интернет технологии или Имплементација на системи со слободен и отворен код)
10.	<p>Course program goals (competencies): Getting familiar with the basic concept of development and operations (DevOps) through continuous integration, testing, delivery and deployment of software solutions and systems. Learning about the different perspectives of the development (Dev) and operations (Ops) part of a system's life-cycle. Enabling students to use tools for automating the processes of integration, testing, delivery and deployment. Learning about the challenges and best practices in managing software releases and configuration.</p>	
11.	<p>Course program content: - Introduction to DevOps. Familiarizing with the CAMS principles (Culture, Automation, Measurement, Sharing). - Development and operations. Different perspectives of development (Dev) and operational (Ops) activities, potential conflicts between them and ways to resolve them. - Using continuous integration in the software development process. - Tools for continuous testing through pipeline automation. - Implementing continuous delivery in software processes. - Continuous management of technical debt. - Software release management. - Software configuration management. - Pipeline execution monitoring. - Using and automating software repositories. - Using software orchestration systems for executing a pipeline.</p>	
12.	<p>Learning methods: Lectures using presentations, interactive lectures, exercises (using equipment and software packages), teamwork, case studies, invited guest lecturers, independent preparation and defense of a project assignment and seminar work.</p>	
13.	Total available time	6 ECTS x 30 hours = 180 hours
14.	Distribution of the available time	30 + 45 + 15 + 15 + 75 = 180 hours

15.	Teaching activity forms	15.1.	Lectures – theoretical teaching	30 hours
		15.2.	Exercises (laboratory, auditory), seminar papers, teamwork	45 hours
16.	Other activity forms	16.1.	Project Tasks	15 hours
		16.2.	Independent Learning Tasks	15 hours
		16.3.	Home learning	75 hours
17.	Assessment methodology			
	17.1.	Tests		10 points
	17.2.	Seminar paper/project (presentation: written and oral)		10 points
	17.3.	Activity and learning		10 points
	17.4.	Final exam		70 points
18.	Assessment criteria (points/grade)	up to 50 points		5 (five) (F)
		51 to 60 points		6 (six) (E)
		61 to 70 points		7 (seven) (D)
		71 to 80 points		8 (eight) (C)
		81 to 90 points		9 (nine) (B)
		91 to 100 points		10 (ten) (A)
19.	Course completion and final exam requirements	Realized activities 15.1 and 15.2		
20.	Teaching Language	Macedonian and English		
21.	Teaching quality evaluation method	Internal evaluation mechanisms and questionnaires		
22.	Course Material			
	22.1.	Mandatory course material		

No	Author	Title	Publisher	Year
1	Nicole Forsgren, Jez Humble, Gene Kim	Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations	IT Revolution Press	2018
2	Gene Kim, Jez Humble, Patrick Debois, John Willis, John Allspaw	The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations	IT Revolution Press	2016
3	Len Bass, Ingo Weber, Liming Zhu	DevOps: A Software Architect's Perspective	Addison-Wesley	2015
4	Jez Humble, David Farley	Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation	Addison-Wesley	2010
5	Paul Duvall, Stephen Matyas, Andrew Glover	Continuous Integration: Improving Software Quality and Reducing Risk	Addison-Wesley	2007
22.2.	Additional course material			
No.	Author	Title	Publisher	Year