Course title		Code	semester	T+U	credit	ECTS
Software Validation and Testing			7	3+0	3	6

Prerequisite Courses	None
Language of the Course	English
Course Level	Undergraduate
Type of Course	Compulsory
Course Coordinator	
Instructors	
Course Assistants	
The aim of lesson	The primary aim of this course is to teach students the basics and principles of software testing. The second aim is to inform students about basic testing methods and technologies to enable them to develop a high quality software product. The third aim of the course is to provide students with the necessary skills in software testing in the international software testing certification process. These goals will be achieved by learning software testing processes, test documentation, testing techniques, test management and testing tools.
Course Content	One of the generally accepted issues in software development is that it is not possible to develop perfect software. Therefore, before the software is used, it is necessary to reduce the effects of incorrect operation by clearing existing errors. Testing is also necessary to ensure good performance of the software. In this course, concepts and definitions of test processes, test documentation and test techniques in software engineering are taught.
Course Learning Outcomes	Students who successfully complete this course; 1. will be able to use software test terminology; 2. Will be able to explain test concepts and test types; 3. Will be able to specify test targets in software development process; 4. will be able to apply test techniques; 5. Will be able to explain the concept of software quality.

Weeks	Topics
one	Fundamentals of software testing
2	Software testing
3	Testing throughout the software lifecycle. Software testing and test documentation at YGYD
4	Static techniques. Software review and circular confusion
5	Test design techniques: Test development process. Component testing - unit testing: JUnit
6	Test design techniques: Black box testing techniques. Decision table testing and equivalence partitioning
7	Test design techniques: Transparent box testing techniques. Decision table testing and equivalence partitioning
8	Test Management.
9	Lab: Software testing framework for web applications: Selenium IDE
10	Quality management. Load test tool for web applications: Jmeter
11th	Quality management. Load test tool for web applications: Jmeter
12	Mutation testing, calculation of mutant costs
13	Mutation testing, calculation of mutant costs
14	Case Study 1
15	Case Study 2

General Competencies

To be able to research and learn about any given software engineering technical concept in the most accurate way.

resources

Sommerville I. Software Engineering. 10th ed. Addison Wesley, 2016, ISBN-13: 978-0133943030.

Black R., van Veenendaal E. and Graham D. Foundations of Software Testing. 3rd ed. Cengage Learning, 2015, ISBN-13: 978-8131526361.

Evaluation System

The dates, days and hours of the Midterm Exam, Quiz, Final Exam and Evaluations will be announced later, according to the decision of the Faculty Administrative Board.

	WITH PROGRAM LEARNING OUTCOMES COURSE LEARNING OUTCOMES RELATIONSHIP TABLE										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PC11
INCREASE 1	4	3	3	5	4	4	4	5	5	4	4
INCREASE 2	3	3	3	4	4	3	3	3	5	4	5
INCREASE 3	4	5	5	4	5	3	3	3	3	3	3
LO4	3	5	5	3	5	4	3	3	3	3	3
LO5	3	5	5	4	5	3	3	3	3	3	3
LO: Learning Outcomes OP: Program Outcomes											
Contribution Level	1 Very Low		2 Low		3 Medium		4 High		5 Very High		

Relation of Program Outcomes and Related Course

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
Software Validation and Testing	4	3	3	4	4	4	4	4	5	4	3