

QF01/0408-4.0E	Course Plan for Master program - Study Plan Development and Updating Procedures/ Department
----------------	--

Study plan No.	2021/2022	University Specialization	Software Engineering
Course No.	0104714	Course name	Advanced Software Testing
Credit Hours	3	Prerequisite Co-requisite	
Course type	<input type="checkbox"/> MANDATORY UNIVERSITY REQUIREMENT	<input type="checkbox"/> UNIVERSITY ELECTIVE REQUIREMENTS	<input type="checkbox"/> FACULTY MANDATORY REQUIREMENT
			<input type="checkbox"/> Support course family requirements
			<input checked="" type="checkbox"/> Mandatory requirements
			<input type="checkbox"/> Elective requirements
Teaching style	<input type="checkbox"/> Full online learning	<input checked="" type="checkbox"/> Blended learning	<input type="checkbox"/> Traditional learning
Teaching model	2Synchronous: 1asynchronous	<input checked="" type="checkbox"/> 2 face to face : 1synchronous	<input type="checkbox"/> 3 Traditional

Faculty member and study divisions information (to be filled in each semester by the subject instructor)

Name	Academic rank	Office No.	Phone No.	E-mail	
Mohammed Lafi	Assistant professor	322	0795325333	lafi@zuj.edu.jo	
Division number	Time	Place	Number of students	Teaching style	Approved model

Brief description

<p>This course will examine advanced software testing techniques. In particular, the important phases of testing will be reviewed, emphasizing on the significance of each phase when testing different types of software.</p> <p>Students will learn the state of the art in testing technology for object-oriented, component-based, concurrent, distributed, graphical-user interface, and web software. In addition, closely related concepts such as model checking and program analysis will also be studied. Emerging concepts such as test-case prioritization and their impact on testing will be examined. Students will gain hands-on testing/analysis experience by proposing new solutions to open research problems in the field of software testing and experimentally demonstrating the strengths/weaknesses of their solutions.</p>
--

Learning resources

Course book information (Title, author, date of issue, publisher ... etc)	Paul Ammann and Jeff Offutt. 2017. Introduction to Software Testing (2nd ed.). Cambridge University Press, New York, NY, USA.
Supportive learning resources (Books, databases, periodicals, software, applications, others)	<ol style="list-style-type: none"> Aditya P. Mathur, Foundations of Software Testing, 2/e , Publisher: Pearson Education India (September 4, 2013) Markus Gärtner, Markus Gärtner , ATDD by Example: A Practical Guide to Acceptance Test-Driven Development (Addison-Wesley Signature Series (Beck)) 2012 Paul C. Jorgensen, Software Testing: A Craftsman's Approach, Fourth Edition, Published: October 18, 2013 by Auerbach Publications

QF01/0408-4.0E	Course Plan for Master program - Study Plan Development and Updating Procedures/ Department
----------------	--

Supporting websites				
The physical environment for teaching	<input checked="" type="checkbox"/> Class room	<input checked="" type="checkbox"/> labs	<input type="checkbox"/> Virtual educational platform	<input type="checkbox"/> Others
Necessary equipment and software	Java NetBeans, JUnit			
Supporting people with special needs				
For technical support				

Course learning outcomes (S= Skills, C= Competences K= Knowledge,)

No.	Course learning outcomes	The associated program learning output code
Knowledge		
K1	understand the concepts and theory related to software testing.	MK1
K2	understand different testing techniques used in designing test plans, developing test suites, and evaluating test suite coverage	MK1
Skills		
S1	Apply knowledge of software testing techniques standards and processes to enhance software quality.	MS1
S2	Synthesize knowledge of software systems for effectively and efficiently conducting software testing.	MS1
S3	Compose, analyze and evaluate test cases for software systems and make justified recommendations to improve testing processes and software quality.	MS1
S4	Apply software testing tools for automated tests and unit testing.	MS1
Competences		
C1	Work in groups to design tests in different domains	MC1
C2	Apply software testing techniques in research	MC2

Mechanisms for direct evaluation of learning outcomes

Type of assessment / learning style	Fully electronic learning	Blended learning	Traditional Learning (Theory Learning)	Traditional Learning (Practical Learning)
First exam	0	0	%20	0
Second / midterm exam	%30	%30	%20	30%
Participation / practical applications	0	0	10	30%
Asynchronous interactive activities	%30	%30	0	0
final exam	%40	%40	%50	40%

QF01/0408-4.0E	Course Plan for Master program - Study Plan Development and Updating Procedures/ Department
----------------	--

Note: Asynchronous interactive activities are activities, tasks, projects, assignments, research, studies, projects, work within student groups ... etc, which the student carries out on his own, through the virtual platform without a direct encounter with the subject teacher.

Schedule of simultaneous / face-to-face encounters and their topics

Week	Subject	learning style*	Reference **
1	Why Do We Test Software?	Lecture	1-17
2	Model-Driven Test Design	Lecture	19-34
3	Test Automation	Lecture	35-53
4	Putting Testing First	Lecture	54-62
5	Criteria-Based Test Design	Lecture	64-70
6	Input Space Partitioning	Lecture	75-105
7	Graph Coverage	Lecture	106-173
8	Logic Testing	Lecture	177-231
9	Midterm exam	Lecture	
10	Syntax-Based Testing	Lecture	234-281
11	Managing the Test Process	Lecture	285-291
12	writing Test Plans	Lecture	292-295
13	test Implementation	Lecture	296-303
14	Regression Testing for Evolving Software	Lecture	
15	Final discussion and presentation of students' research work	Lecture	234-281
16	Final Exam	Final Exam	

* Learning styles: Lecture, flipped learning, learning through projects, learning through problem solving, participatory learning ... etc.

** Reference: Pages in a book, database, recorded lecture, content on the e-learning platform, video, website ... etc.

Schedule of asynchronous interactive activities (in the case of e-learning and blended learning)

Week	Task / activity	Reference	Expected results
1	Choose a research topic related to software testing	Week 1 lectures	Ability to select a research topic in software testing
2	Write literature review about the selected topic in software testing	Week 2 lectures	Ability to write literature review in a software testing topic
3	Discussion forum	Week 3 lectures	
4	Write a background about a select topic in software testing	Week 4 lectures	Ability to write background in a software testing topic

QF01/0408-4.0E	Course Plan for Master program - Study Plan Development and Updating Procedures/ Department
-----------------------	--

5	Write a summary and comparison between different software testing research approaches in a certain topic	Week 5 lectures	Ability to write a summary and comparisons of different software testing research approaches in a certain topic
6	Discussion forum	Week 6 lectures	
7	One or more of the following interactive contents: fill in blanks, drag the words, Drag and drop, dialog cards, flashcards, multiple choice, fill in blanks, True/False questions	Week 7 lectures	Practice testing approaches
8	One or more of the following interactive contents: fill in blanks, drag the words, Drag and drop, dialog cards, flashcards, multiple choice, fill in blanks, True/False questions	Week 8 lectures	Practice testing approaches
9	Discussion forum	Week 9 lectures	
10	One or more of the following interactive contents: fill in blanks, drag the words, Drag and drop, dialog cards, flashcards, multiple choice, fill in blanks, True/False questions	Week 10 lectures	Practice testing approaches
11	One or more of the following interactive contents: fill in blanks, drag the words, Drag and drop, dialog cards, flashcards, multiple choice, fill in blanks, True/False questions	Week 11 lectures	Practice testing approaches
12	Discussion forum	Week 12 lectures	
13	Practicing Junit testing	Week 13 lectures	Get familiar with automated testing using JUnit
14	Practicing Junit testing	Week 14 lectures	Get familiar with automated testing using JUnit
15	Discussion forum	Week 15 lectures	
16	Write a term paper	Weeks 1-15 lectures	Review final exam material