هيئة تقويم التعليم والتدريب
Education & Training Evaluation Commission
اعتماد
NCAAA
2024
TP-153

Course Specification

(Bachelor)

Course Title:	Software Testing and Quality Assurance
Course Code:	SE4012
Program:	BSc in Computer Science
Department:	Software Engineering
College:	College of Computing
Institution:	Umm Al Qura University
Version:	1.0
Last Revision Date:	22/04/2025

Table of Contents

A. General information about the course:

1. Course Identification

1. Credit hours: (3)					
2. Course type					
A.	☐University	☐College	☒ Department	☐Track	☐Others
B.	☐ Required		☒ Elective		
3. Level/year at which this course is offered: (4th year/ 7th or 8th level)					
4. Course General Description:					
This course provides a comprehensive understanding of the principles, techniques, and tools used in software testing and quality assurance. Students will learn to design, implement, and manage testing processes to ensure software reliability, functionality, and performance. The course emphasizes practical skills in creating test cases, using automated testing tools, and managing defects, while exploring topics such as security testing, regression testing, and testing in Agile environments.					
5. Pre-requirements for this course (if any)**:**					
SE1201 - Foundations of Software Engineering					
6. Co-requisites for this course (if any)**:**					
N/A					
7. Course Main Objective(s):					
Upon successful completion of this course, you will be able to: 1. Equip students with essential knowledge and techniques in software testing and quality assurance, focusing on ensuring software reliability, functionality, and performance. 2. Enable students to design, develop, and execute test plans using industry-standard tools and methodologies, with an emphasis on automated and performance testing. 3. Develop students' abilities to identify and manage defects, ensuring thorough software validation and verification in various development environments, including Agile and DevOps. 4. Instill a professional approach to software quality management by promoting collaboration, ethical responsibility, and adherence to industry standards.					

2. Teaching mode (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	Traditional classroom	٦٠	100%
2	E-learning	0	0
3	Hybrid • Traditional classroom • E-learning	0	0
4	Distance learning	0	0

3. Contact Hours (based on the academic semester)

No	Activity	Contact Hours
1.	**Lectures**	٣٠
2.	**Laboratory/Studio**	٣٠
3.	**Field**	0
4.	**Tutorial**	0
5.	**Others (specify)**	0
Total		٦٠

B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
1.0	**Knowledge and understanding**			
1.1	Understand key concepts in software quality assurance, including verification, validation, and defect management.	K1	Lecture, Exercise	Quiz, Exams, Assignments
1.2	Identify and differentiate between various types of testing (e.g., unit, integration, system, acceptance).	K2	Lecture, Exercise	Quiz, Exams, Assignments
2.0	**Skills**			

Code	Course Learning Outcomes	Code of PLOs aligned with the program	Teaching Strategies	Assessment Methods
2.1	Design and execute test cases based on software requirements and specifications.	S1	Lecture, Exercise	Quiz, Exams, Assignments
2.2	Apply testing techniques (e.g., boundary value analysis, equivalence partitioning) to real-world scenarios.	S3	Lecture, Exercise	Quiz, Exams, Assignments
2.3	Use testing tools (e.g., JUnit, Selenium, Postman, JMeter) to automate testing processes.	S2	Lecture, Exercise	Quiz, Exams, Assignments
3.0	**Values, autonomy, and responsibility**			
3.1	Demonstrate ethical and professional behavior in maintaining software quality standards.	V1	Discussion, role-playing scenarios	reflective essays

C. Course Content

No	List of Topics	Contact Hours
1.	Introduction to Software Testing and Quality Assurance	4
2.	Test Planning, Coverage and Types of Software Testing	4
3.	Unit testing frameworks (Junit, Mokito) and techniques (stubs, mocks, fake)	8
4.	Integration, System, and Acceptance Testing (Database, API)	8
5.	Functional Testing Automation (Selenium)	4
6.	Performance Testing (JMeter)	4
7.	Static Code Testing (code review, static analysis)	4
8.	Security Testing (SQL Injection, XSS)	4
9.	Regression Testing and maintenance	4
10.	Test in Agile and DevOps(CI/CD pipeline)	4
11.	Defects Management and Reporting (Jira)	4

12.	Verification & Validation	4
13.	AI-Driven Test Case Generation and Defect Prediction • AI-Powered Test Case Generation • Defect Prediction and Analysis • AI for Test Optimization	4
Total		60

D. Students Assessment Activities

No	Assessment Activities *	Assessment timing (in week no)	Percentage of Total Assessment Score
1.	Quizzes and Assignments	2-14	20
2.	Projects	2-14	10
4.	Lab	2-14	10
5.	Mid Term	7	20
6.	Final Exam	16-17	40

*Assessment Activities (i.e., Written test, oral test, oral presentation, group project, essay, etc.).

E. Learning Resources and Facilities

1. References and Learning Resources

Essential References	• Bierig R, Brown S, Galván E, Timoney J. *Essentials of software testing*. Cambridge University Press; 2021. • Shen, J. J. (2019). *Software testing: Techniques, principles, and practices*. Independently Published. ISBN 978-1693054907.
Supportive References	• Aniche, M. (2022). *Effective software testing: A developer's guide*. Manning Publications. ISBN 978-1633439931.
Electronic Materials	
Other Learning Materials	

2. Required Facilities and equipment

Items	Resources
facilities (Classrooms, laboratories, exhibition rooms, simulation rooms, etc.)	Classroom
Technology equipment (projector, smart board, software)	Projector

Items	Resources
Other equipment (depending on the nature of the specialty)	N/A

F. Assessment of Course Quality

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of teaching	Students	Direct, Indirect
Effectiveness of Students' assessment	Faculty, Peer reviewer	Direct, Indirect
Quality of learning resources	Faculty, Course coordinator	Direct, Indirect
The extent to which CLOs have been achieved	Course coordinator, Program management committee	Direct
Other		

Assessors (Students, Faculty, Program Leaders, Peer Reviewers, Others (specify)
Assessment Methods (Direct, Indirect)

G. Specification Approval

COUNCIL /COMMITTEE	**SOFTWARE ENGINEERING DEPARTMENT COUNCIL**
REFERENCE NO.	**THE 17TH MEETING FOR THE ACADEMIC YEAR 1446H**
DATE	**22/04/2025**

COUNCIL /COMMITTEE	**Computer Science and Artificial Intelligence Department Council**
REFERENCE NO.	**THE 16TH MEETING FOR THE ACADEMIC YEAR 1446H**
DATE	**22/04/2025**