Faculty: Information Technology		
Department: Software Engineering	**Program: Master degree**	
Academic year:	**Semester:**	

Course Plan

First: Course Information

Course Number: (1503712)	Course Name: Software Verification and Validation	Credit Hours: 3
Prerequisite Number:	Section Number:	Lecture Time:
Course Type:	☐ *University Obligatory Requirement* ☐ *Faculties Obligatory Requirement* ☐ *Department Elective Requirement*	☐ *University Elective Requirement* ◼ *Department Obligatory Requirement* ☐ *Supporting Requirement*
Type of Education:	☐ **Fully Direct (Fully Face-to-Face Education)** ☐ **Blended Learning (2 Face-to-Face + 1Asynchronous)** ◼ **Fully Electronic Education (2Synchronous + 1 Asynchronous)**	

Second: Instructor's Information

Name:		Academic Rank: Associate Professor		
Office Number:		Ext. Number:		
Office Hours:	Sunday	Monday	Tuesday	Wednesday	Thursday

Third: Course Description

The goal of this module is to provide a broad systematic study of quality assurance aspects of the software development process with an emphasis software quality, software testing, and software quality certification and standards. This module will expose students to the principles of software quality assurance and identify the tasks that are essential for successful quality projects and discuss how tasks interact with each other. It will also present current methods, techniques and certification standards involved in software quality assurance from a practical industry implementation perspective. The specific objectives of the module are: - Understand and define the scope of the software development process from a quality perspective - Understand, design and implement procedures for developing software quality - Understand the issues and approaches involved in software quality assurance at the company practice level - Understand the main approaches to software testing - Understand and be able

to implement testing solutions at code level. - Benchmark organizations against industry standards for software quality

Fourth: Learning Source

Main Reference:	Software Quality Engineering: testing quality assurance and quantifiable improvement	
Author: Jeff Tian	Issue No.: 3rd Edition	Publication Year: 2020
Additional Sources&Websites:	• Moodle	
Teaching Type:	☐Classroom ☐ Laboratory ☐ Workshop ◼ MS Teams ◼ Moodle	

Fifth: Learning Outcomes

Course code	Course Intended learning outcome (CILOs)	Connection to program ILOs Code
Knowledge		
K1	Understand the concepts of software testing and quality assurance	PK1, PK4
K2	Identify important historical and current literature addressing software quality assurance	PK4
K3	Understand matrix decompositions and how it can be applied to Software System	PK4
K4	Explain and construct quality-oriented software development processes	PK4
K5	Describe the concepts behind software testing and appraise the most appropriate testing approaches for a given situation `	PK4
K6	Identify and contrast the basic principles behind software process, process improvement and process standards	PK4
Skills		
S1	The ability to use Software testing method and tools	PS3
S2	The ability to analytic and Evaluate the concepts embodied in the most prevalent software quality assurance	PS3

S3	The ability to analytic and Evaluate the concepts software testing in the most prevalent software systems	PS3
S4	The ability use the software testing techniques and methods, including knowledge of their advantages and disadvantages, and when it may be appropriate to use each approach	PS3
S5	The ability use the quality assurance techniques and methods, including knowledge of their advantages and disadvantages, and when it may be appropriate to use each approach	PS3
Competencies		
C1	Analytic skills: Assess	PC2
C2	Teamwork	PC2
C3	Creativity	PC3
C4	Leadership	PC2
C5	Strategic Thinking	PC1

* P: Program, **K: knowledge, ***S: skills, ****C: competencies.

Sixth: Course Structure

Lecture Date	Intended Teaching Outcomes(ILOs)	Topic(s)	Teaching *Procedures	Teaching ***Methods	References***
Week 01	K1, K3, S1 K1, K3, S1 S2, S3, C3 K1, K3 K1, K3	Introduction to Software Testing	Synchronous	Lecture	Outline File
		Research Methods	Synchronous	Lecture	Textbook
		Papers Summary	Synchronous	Lecture	Textbook
		Systematic Papers	Synchronous	Lecture	Textbook
		Summary Paper Summary A systematic snapshot of small packaged software vendors' enterprises	Asynchronous	Reading material and Executing Tasks	Course Slides / book / External Sources
Week 02	K1 K1 K1, K3	What is software testing	Synchronous	Lecture	Textbook
		Software Development Methodologies	Synchronous	Lecture	Textbook
		Quiz	Asynchronous	Quiz	Course Slides / book / External Sources
Week 03	K1, K3, K4 K1, K3, K4, 3 C1, C2, C3 S1, S2	Why Testing is Important	Synchronous	Lecture	Textbook
		Program testing	Synchronous	Lecture	Textbook
		Validation and defect testing	Synchronous	Lecture	Textbook

		Questions and discussion	Asynchronous	Executing Tasks	Course Slides / book / External Sources
Week 04	K1, K3, K4, S1, S2, S3	Inspections and testing	Synchronous	Lecture	Textbook
		Advantages of inspections	Synchronous	Lecture	Textbook
		Video Summary	Asynchronous	Video	Course Slides / book / External Sources
Week 05	K1, K3, S1, C1, C2, C3	Levels of Test: The V-model	Synchronous	Lecture	Textbook
		Testing New Product Versions	Synchronous	Lecture	Textbook
		Types of Testing	Synchronous	Lecture	Textbook
		Paper Analysis and comparison	Asynchronous	Reading material and Executing Tasks	Course Slides / book / External Sources
Week 06	K1, K3, K5, S1, S2, S3, C1, C2	Black Box Introduction	Synchronous	Lecture	Textbook
		Black Box Equivalence partitioning	Synchronous	Lecture	Textbook
		Case Study on Equivalence partitioning	Asynchronous	Video, Reading material and Executing Tasks	Course Slides / book / External Sources
Week 07	K1, K3, K5, S1	Black Box testing Decision Table	Synchronous	Lecture	Textbook
		Black Box Boundary Testing	Synchronous	Lecture	Textbook
		Case Study on Decision Table and Boundary Testing	Asynchronous	Video, Reading material and Executing Tasks	Course Slides / book / External Sources
Week 08	K3, S2, C3	Introduction White box testing	Synchronous	Lecture	Textbook
		Video Summary	Asynchronous	Video	Course Slides / book / External Sources
Week 09	K3, S2, C3	Statement Coverage Criterion	Synchronous	Lecture	Textbook
		Path Coverage Criterion	Synchronous	Lecture	Textbook
		Case Study about Statement Coverage Criterion and Path Coverage Criterion	Asynchronous	Video, Reading material and Executing Tasks	Course Slides / book / External Sources
Week 10	K1, K3, S1, S2, S3, C3, C4, C5	LCSAJ coverage	Synchronous	Lecture	Textbook
		Data Flow Criterion	Synchronous	Lecture	Textbook
		Video Summary	Asynchronous	Video	Course Slides / book / External Sources
Week 11	**Apply the knowledge gained throughout the course to create a software model for a real-world scenario.** **Presentation about real-world scenario**				

* Learning procedures: (Face-to-Face, synchronous, asynchronous). * * Teaching methods: (Lecture, video…..). ** * Reference: (Pages of the book, recorded lecture, video….).

Seventh: Assessment methods

Methods	Online Learning	Blended Learning	Face-To-Face Learning	Measurable Course (ILOs)
First Exam	0			
Second Exam	0			
Mid-term Exam	0			
Participation				K1, K2, K3, S1, S2
Asynchronous Activities	60			K4, K5, K6, K7, S1, S2, C1, C2
Final Exam	40			K1, K2, K3, S1, S2

Eighth: Course Polices

- All course policies are applied on all teaching patterns (online, blended, and face-to-face Learning) as follows:
  a. Punctuality.
  b. Participation and interaction.
  c. Attendance and exams.
- Academic integrity: (cheating and plagiarism are prohibited).

Approval	Name	Date	Signature
Head of Department			
Faculty Dean			