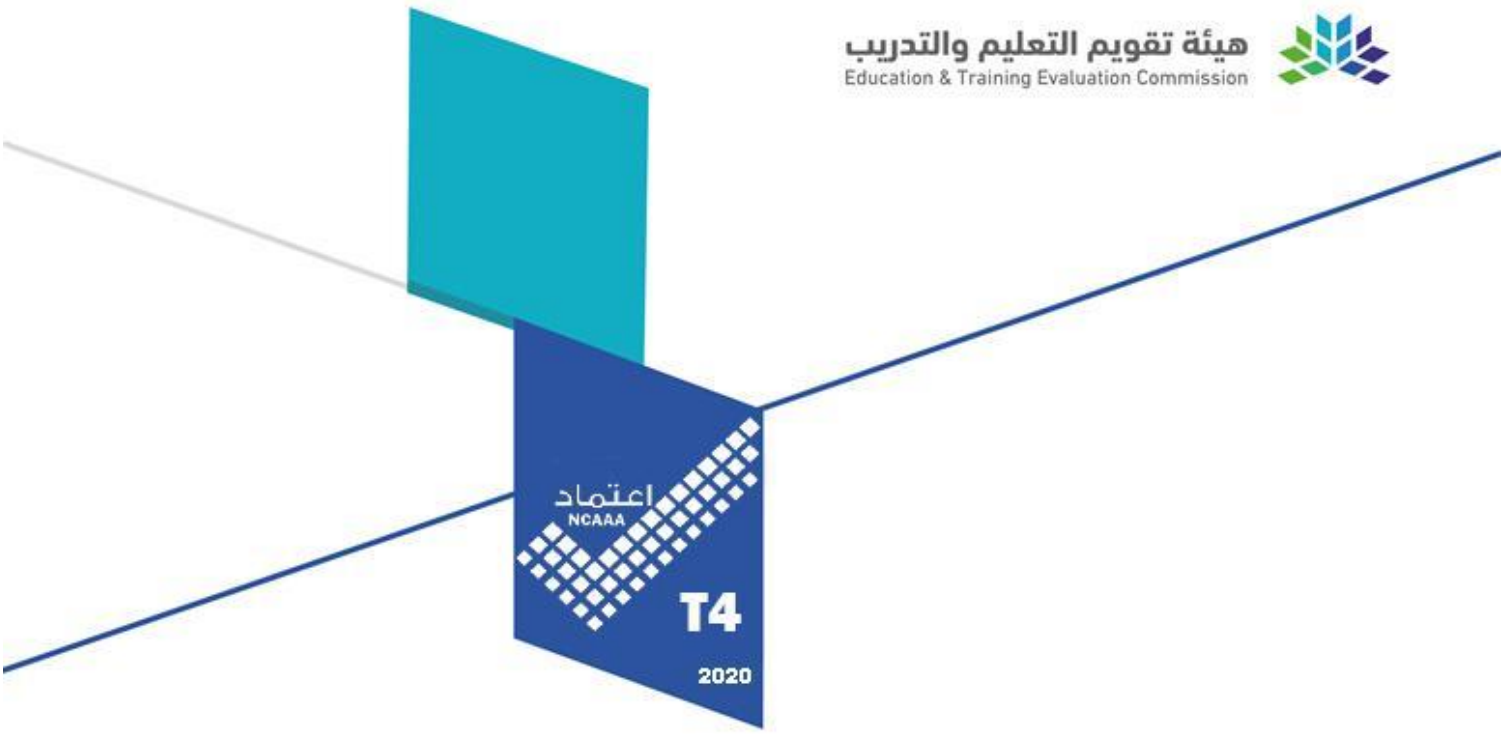
هيئة تقويم التعليم والتدريب
Education & Training Evaluation Commission



Course Specifications

Course Title:	Advanced Software Engineering	
Course Code:	**501446-3**	
Program:	**Bachelor in Computer Science**	
Department:	**Department of Computer Science**	
College:	**College of Computers and Information Technology**	
Institution:	**Taif University**	

Table of Contents

A. Course Identification

1. Credit hours:3
2. Course type **a.** University ☐ College ☐ Department ☑ Others ☐ **b.** Required ☑ Elective ☐
3. Level/year at which this course is offered: 10/4
4. Pre-requisites for this course (if any)**:** Software Engineering 501343-3
5. Co-requisites for this course (if any)**: None**

6. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	**Traditional classroom**	5	100%
2	**Blended**	0	0
3	**E-learning**	0	0
4	**Distance learning**	0	0
5	**Other**	0	0

7. Contact Hours (based on academic semester)

No	Activity	Contact Hours
1	**Lecture**	50
2	**Laboratory/Studio**	0
3	**Tutorial**	0
4	**Others** (specify)	0
	Total	50

B. Course Objectives and Learning Outcomes

1. Course Description

This course aims to equip students with appropriate software tools and environments for successful software implementation, testing and verification, maintenance and evolution, quality assurance, software metrics, and requirements engineering.

2. Course Main Objective

The main objective is to provide students with advanced knowledge in software engineering and improve their skills in software engineering practices using appropriate tools.

3. Course Learning Outcomes

CLOs		Aligned PLOs
1	**Knowledge and Understanding**	
1.1	Specify key techniques and tools in software testing.	K1
1.2	Understand various quality assurance techniques, including unit testing, functional testing, and automated analysis tool	K1
1.3	Understand the reuse-driven software engineering	K1
1.4	Define software maintenance and re-engineering process	K1
1.5	Understand the use of metrics in software engineering	K1
2	**Skills :**	
2.1	Apply an appropriate and effective software process for a given project.	S1
2.2	Evaluate software quality.	S2
3	**Values:**	
3.1	Work as a team leader/member of a software testing team.	V2

C. Course Content

No	List of Topics	Contact Hours
1	Software tools and environment	5
2	Software implementation (from design to implementation)	5
3	Software testing and verification	10
4	Software metrics	5
5	Software quality assurance	5
6	Software evolution	5
7	Software Maintenance and re-engineering.	5
8	Emerging Trends in software Engineering.	10
Total		50

D. Teaching and Assessment

1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	**Knowledge and Understanding**		
1.1	Specify key techniques and tools in software testing.	Lecture Discussion	**Direct Assessment Tool** Assignments Projects Quizzes **Indirect Assessment Tool** Course Exit Survey
1.2	Understand various quality assurance techniques, including unit testing, functional testing, and automated analysis tool	Lecture Discussion	**Direct Assessment Tool** Written Exams Projects Assignments Quizzes **Indirect Assessment Tool** Course Exit Survey
1.3	Understand the reuse-driven software engineering	Lecture Discussion	**Direct Assessment Tool** Written Exams Assignments Projects Quizzes **Indirect Assessment Tool** Course Exit Survey
1.4	Define software maintenance and re-engineering process	Lecture Discussion	**Direct Assessment Tool** Written Exams Assignments Projects Quizzes **Indirect Assessment Tool** Course Exit Survey
1.5	Understand the use of metrics in software engineering	Lecture Discussion	**Direct Assessment Tool** Written Exams Assignments Projects Quizzes

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
			Indirect Assessment Tool Course Exit Survey
2.0	**Skills**		
2.1	Apply an appropriate and effective software process of a given project.	Lecture Discussion	**Direct Assessment Tool** Written Exams Assignments Quizzes Project **Indirect Assessment Tool** Course Exit Survey
2.2	Evaluate software quality.	Lecture Discussion	**Direct Assessment Tool** Written Exams Assignments Quizzes Project **Indirect Assessment Tool** Course Exit Survey
3.0	**Values**		
3.1	Work as a team leader/member of a software testing team.	Lecture Discussion	**Direct Assessment Tool** Project **Indirect Assessment Tool** Course Exit Survey

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	Assignments	10	10%
2	Quizzes	6	10%
3	Midterm Exam	6	25%
5	Project	10	15%
6	Final Exam	12	40%

***Assessment task** (i.e., written test, oral test, oral presentation, group project, essay, etc.)

E. Student Academic Counseling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice:

Academic advising and counseling of students is an important component of teaching; student academic advising is a mandatory requirement of College of Computers and Information Technology (CCIT). Appropriate student advising provides support needed for the student during times of difficulty. In addition, it helps the student to build a close relationship with his/her advisor and to provide student motivation and involvement with the institution.

In addition, since faculty are usually the first to recognize that a student is having difficulty, faculty members play a key role in developing solutions for the students or referring them to appropriate services. Faculty members also participate in the formal student-mentoring program.

Additional counseling is provided by course directors, who provide students with academic reinforcement and assistance and refer “at risk” students to the Vice Dean for Academic Affairs and the Vice Dean for female section

F. Learning Resources and Facilities

1.Learning Resources

Required Textbooks	Sommerville, I., Software Engineering, latest Edition.
Essential References Materials	Pressman, & Roger S. & Ice, Darrel, Software Engineering a Practitioner's Approach: European Adaptation latest Edition
Electronic Materials	Presentations and recorded lectures
Other Learning Materials	NON

2. Facilities Required

Item	Resources
Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)	A Lecture room appropriate for maximum 25 students with a personal computer, a data show, and a smart board.
Technology Resources (AV, data show, Smart Board, software, etc.)	data show, UML editor software, Prototyping software tool and Project management software.
Other Resources (Specify, e.g. if specific laboratory equipment is required, list requirements or attach a list)	

G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
Effectiveness of Teaching	Students	Students surveys and Students course evaluation ●
Improvement of Teaching	Course Coordinator	Deficiencies based on the student Evaluation, faculty input, course file, and program assessment
Verifying Standards of Student Achievement	● Curriculum Committee	● Review CAF (Course assessment file) ● Alumni surveys. Periodic exchange and remarking of tests or a sample of assignments with staff at another

Evaluation areas (e.g., Effectiveness of teaching and assessment, Extent of achievement of course learning outcomes, Quality of learning resources, etc.)
Evaluators (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)
Assessment Methods (Direct, Indirect)

H. Specification Approval Data

Council / Committee	CS council
Reference No.	Meeting #12
Date	23-10-1443