



Course Specifications

Course Title:	Software Engineering 1
Course Code:	CSI-325
Program:	Computer Science & Information
Department:	Computer Science and Information
College:	College of Science at Az Zulfi
Institution:	Majmaah university

Table of Contents

A. Course Identification	3
6. Mode of Instruction (mark all that apply)	3
B. Course Objectives and Learning Outcomes	4
1. Course Description	4
2. Course Main Objective.....	4
3. Course Learning Outcomes	4
C. Course Content	5
D. Teaching and Assessment	5
1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods	5
2. Assessment Tasks for Students	5
E. Student Academic Counseling and Support	6
F. Learning Resources and Facilities	6
1. Learning Resources	6
2. Facilities Required.....	6
G. Course Quality Evaluation	6
H. Specification Approval Data	7

A. Course Identification

1. Credit hours: 3
2. Course type
a. University <input type="checkbox"/> College <input type="checkbox"/> Department <input checked="" type="checkbox"/> Others <input type="checkbox"/>
b. Required <input type="checkbox"/> Elective <input type="checkbox"/>
3. Level/year at which this course is offered: level 6
4. Pre-requisites for this course (if any): CSI 221
5. Co-requisites for this course (if any): N/A

6. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	Traditional classroom	48	80%
2	Blended	6	10%
3	E-learning	6	10%
4	Correspondence		
5	Other		

7. Actual Learning Hours (based on academic semester)

No	Activity	Learning Hours
Contact Hours		
1	Lecture	30
2	Laboratory/Studio	30
3	Tutorial	
4	Others (specify)	
	Total	60
Other Learning Hours*		
1	Study	
2	Assignments	
3	Library	
4	Projects/Research Essays/Theses	
5	Others (specify)	
	Total	

* The length of time that a learner takes to complete learning activities that lead to achievement of course learning outcomes, such as study time, homework assignments, projects, preparing presentations, library times

B. Course Objectives and Learning Outcomes

1. Course Description

This is a course on the fundamental concepts and principles that underlie current and emerging methods, tools, and techniques for the cost-effective engineering of high-quality software systems. Software engineering (SE) is concerned with all aspects of software development, from the early stages of system specification to maintaining the system after it has gone into use. This includes technical processes of software development as well as activities such as software project management and the development of tools, methods, and theories to support software development. This is NOT a "programming" course; it focuses instead on surveying some of the critical aspects of SE that may be less familiar to students of computer science, such as identifying a development process appropriate to the circumstances, eliciting and documenting requirements, identifying appropriate design techniques, employing effective verification and validation strategies (e.g., reviews and inspections, formal methods) throughout the software lifecycle

2. Course Main Objective

- To help students to develop skills that will enable them to construct software of high quality and to function effectively on teams to accomplish a common goal.
- To make students aware of key aspects of current software engineering approaches.
- Elicit, analyze and specify software requirements through a productive working relationship with project stakeholders.
- To create models of software data and processes using structured modelling approaches.
- To demonstrate skills of software documentation, quality assurance and evaluation, and testing as part of software development.
- Communicate effectively through oral and written reports, and software documentation
- Demonstrate professionalism including continued learning and professional activities.

3. Course Learning Outcomes

CLOs		Aligned PLOs
1	Knowledge:	
1.1	Acquire knowledge of software engineering fundamentals and their practical application	ICS-a2
1.2	Understand of best practices and standards in the field of software engineering, including all the activities of the software development life cycle activities and CASE tools.	a1
1.3		
1...		
2	Skills :	
2.1	Identify and analyze user needs, design, implement, develop and evaluate computer-based systems to meet desired needs.	b1
2.2	Identify and analyze user needs, design, implement, develop and evaluate computer-based systems to meet desired needs.	b3
3	Competence:	
3.1	Present a short report in a written form and orally using appropriate scientific language, and use current techniques, skills, and tools necessary for software engineering	c1

C. Course Content

No	List of Topics	Contact Hours
1	Introduction to Software Engineering	8
2	Software processes	12
3	Software Requirements Engineering	12
4	Software Design	12
5	System Coding, Testing, and Maintenance	16
Total		60

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		
1.1	Acquire knowledge of software engineering fundamentals and their practical application	Lectures Lab demonstrations Case studies Individual presentations	Written Exam Homework assignments Lab assignments Class Activities Quizzes
1.2	Understand of best practices and standards in the field of software engineering, including all the activities of the software development life cycle activities and CASE tools.		
2.0	Skills		
2.1	Identify and analyze user needs, design, implement, develop and evaluate computer-based systems to meet desired needs.	Lectures Lab demonstrations Case studies Individual presentations Brainstorming	Written Exam Homework assignments Lab assignments Class Activities Quizzes
2.2	Identify and analyze user needs, design, implement, develop and evaluate computer-based systems to meet desired needs.		
3.0	Competence		
3.1	Present a short report in a written form and orally using appropriate scientific language, and use current techniques, skills, and tools necessary for software engineering	Small group discussion Whole group discussion Brainstorming Presentation	Written Exam Homework assignments Lab assignments Class Activities Quizzes

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	First written mid-term exam	6	15%
2	Second written mid-term exam	12	15%
3	Presentation, class activities, and group discussion	Every week	10%
4	Homework assignments	After	10%

#	Assessment task*	Week Due	Percentage of Total Assessment Score
		Every chapter	
5	Implementation of presented programs	Every two weeks	10%
6	Final written exam	16	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:

Office hours - Office call – Email - Mobile:

F. Learning Resources and Facilities

1. Learning Resources

Required Textbooks	Ian Sommerville, Software Engineering, 9 th Ed, Addison-Wesley, 2011
Essential References Materials	Roger S. Pressman, Software Engineering: A practitioner's Approach, 6 th ed, McGraw-Hill Science, 2009
Electronic Materials	Determines as the course is going on
Other Learning Materials	Video and presentation

2. Facilities Required

Item	Resources
Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)	Classroom and Lab, as those that are available at college of science at AzZulfi.
Technology Resources (AV, data show, Smart Board, software, etc.)	Smart Board - data show
Other Resources (Specify, e.g. if specific laboratory equipment is required, list requirements or attach a list)	A/N

G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
course evaluation	Student-faculty management meeting	Questionnaires
Evaluation of Teaching	Program/Department Instructor	Discussion within the staff members teaching the course Departmental internal review

Evaluation Areas/Issues	Evaluators	Evaluation Methods
		of the course.

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	
Reference No.	
Date	