



Course Specifications

Institution:	College of Science at Az Zulfi
Academic Department :	Department of Computer Science and Information
Programme :	Computer Science and Information Program
Course :	Software Engineering 2
Course Coordinator :	Dr. Zeiad El-Saghir
Programme Coordinator :	Associate Prof. Yosry Azzam
Course Specification Approved Date :	23/ 12 / 1435 H



A. Course Identification and General Information

1 - Course title : Software Engineering 2 Course Code: CSI 422	
2. Credit hours : (3) (2 Lecture + 2 Lab)	
3 - Program(s) in which the course is offered: Computer Science & Information	
4 – Course Language : English	
5 - Name of faculty member responsible for the course:	
6 - Level/year at which this course is offered : 8th level	
7 - Pre-requisites for this course (if any) :	
<ul style="list-style-type: none"> • CSI 325 	
8 - Co-requisites for this course (if any) :	
<ul style="list-style-type: none"> • N/A 	
9 - Location if not on main campus :	
(College of Science at Az Zulfi)	
10 - Mode of Instruction (mark all that apply)	
A - Traditional classroom	<input checked="" type="checkbox"/> What percentage?
B - Blended (traditional and online)	<input checked="" type="checkbox"/> What percentage?
D - e-learning	<input type="checkbox"/> What percentage?
E - Correspondence	<input type="checkbox"/> What percentage?
F - Other	<input checked="" type="checkbox"/> What percentage?

80 %
10 %
.... %
.... %
10 %

Comments :

One-tenth of the course is presented mainly inside video lectures of other instructors worldwide. They illustrate the same topics that I introduced in my lectures with a different presentation.

B. Objectives

What is the main purpose for this course?

This course introduces students a complex view of object-oriented software development process, aiming in the ability to solve real problems in the given domain. In this course, the student will learn and apply a unified methodology to the analysis, design, implementation, testing and demonstration of a software system of a significant size and complexity. Topics include: Object-oriented software processes,





requirements engineering, system models, Unified Modeling Language (UML) concepts (class diagram, object diagram, use case diagram, collaboration diagram, sequence diagram, component diagram, and deployment diagram), rapid application development and CASE tools for object-oriented systems, object-oriented system testing, operation, maintenance and management.

Here are several course goals:

- To help students to be aware of key aspects of current software engineering approaches.
- Develop skills that will enable them to construct object oriented software of high quality and to function effectively on teams to accomplish a common goal.
- Create models of software data and processes using object oriented modeling approaches.
- Use a common method (UML diagrams) to specify the requirements and design of a moderately sized software product.
- Discuss key principles and common methods for software project management such as scheduling, size estimation, cost estimation and risk analysis.
- Build solutions using different technologies, architectures and life-cycle approaches in the context of different organizational structures.
- Successfully assume a variety of roles in teams of diverse membership.
- Demonstrate professionalism including continued learning and professional activities.

Briefly describe any plans for developing and improving the course that are being implemented :

1. Increasing the ability of the students to implement the methods and practices that are presented in the course.
2. Formative exams during the term with a feedback to the students, so these examinations can be used as a method of learning..
3. Using group discussion through the internet with course attending students.
4. Updating the materials of the course to cover the new topics of the field.
5. Help students to develop their knowledge about the topics that are presented in the course.





C. Course Description

1. Topics to be Covered

List of Topics	No. of Weeks	Contact Hours
1. Introduction to Object-Oriented Software Engineering: Object-Oriented Software Engineering principles - Professional software development - Software engineering ethics - Case studies.	1	4
2. Object-oriented Software Processes: Agile process models, process activities, the Rational Unified Process (RUP), Computer-Aided Software Engineering, and object-oriented task management using suitable CASE tools.	2	8
3. System Models & Requirements Using UML: Context models, behavioral models, data models, object models, CASE workbenches, UML modeling, and using UML diagrams to specify the requirements of a moderately sized software product.	2	8
4. Object Oriented Concepts: Unified Modeling Language (UML): Class diagram, object diagram, use case diagram, collaboration diagram, sequence diagram, component diagram, and deployment diagram, and writing a software requirements specification document for object-oriented systems using appropriate CASE tools (e. g. power designer).	3	12
5. Rapid application development: Rapid application development and CASE tools for object-oriented systems, writing SDS (Software Design Specification) document for Object -oriented systems using CASE tools (e. g. power designer).	3	12
6. System Testing and Maintenance: Object-oriented systems testing, operation and maintenance, using CASE tools in the auto-generation of object-oriented code, and object oriented development using reuse approaches.	2	8
7. Project Management: Software project management, project scheduling, project staffing, software configuration management, quality assurance, and project monitoring.	2	8



2. Course components (total contact hours and credits per semester):

	Lecture	Tutorial	Laboratory	Practical	Other:	Total
Contact Hours	30	30	60
Credit	30	15	45

3. Additional private study/learning hours expected for students per week.

5

The private self-study of my student is crucial for this course. It includes:

- implementing methods and practices using suitable CASE tools,
- reading carefully the topics in the textbook or reference book,
- searching the websites that concerned with the course,
- solving the exercises that are assigned,
- discussing the course topics with the instructor in his office hours,
- watching video lectures of other instructors who presented related topics worldwide.

The total workload of the student in this course is then: $60 + 5 * 15 = 135$ work hours.





4. Course Learning Outcomes in NQF Domains of Learning and Alignment with Assessment Methods and Teaching Strategy

	NQF Learning Domains And Course Learning Outcomes	Course Teaching Strategies	Course Assessment Methods
1.0	Knowledge		
1.1	Acquire knowledge of computing and mathematics appropriate to software engineering including simulation and modeling, and Understand of best practices and standards and their application related to software engineering.	Lectures demonstrations Case studies Individual presentations	Written Exam Homework assignments Lab assignments Class Activities Quizzes
2.0	Cognitive Skills		
2.1	Identify and analyze user needs and take them into account in the selection, creation, evaluation and administration of computer-based systems.	Lectures Lab demonstrations	Written Exam Homework assignments
2.2	Apply software engineering principles and practices to the planning and development of actual software projects, and expert proficiency in the UML2 superstructure to design software architectures.	Case studies Individual presentations Brainstorming	Lab assignments Class Activities Quizzes
3.0	Interpersonal Skills & Responsibility		
3.1	Use current techniques, skills, and tools necessary for software engineering practice.	Small group discussion Lab Whole group discussion Brainstorming Presentation	Written Exam Homework assignments Lab assignments Class Activities Quizzes
4.0	Communication, Information Technology, Numerical		
4.1	Function effectively on teams to accomplish a common goal, and Communicate effectively with a range of audiences.	Small group discussion Lab Whole group discussion Brainstorming Presentation	Written Exam Homework assignments Lab assignments Class Activities Quizzes
5.0	Psychomotor		
5.1
5.2



5. Schedule of Assessment Tasks for Students During the Semester:

	Assessment task	Week Due	Proportion of Total Assessment
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	5%
4	Homework assignments	After Every chapter	5%
5	Final written exam	16	40%
6	Final Lab Exam	15	20%
	Total		100%

D. Student Academic Counseling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice. (include amount of time teaching staff are expected to be available each week)

Office hours: Sun – Wed: 12 PM - 2:00 PM

Office call: Sun – Wed: 12 PM - 2:00 PM

Email: z.abdoun@mu.edu.sa





E. Learning Resources

1. List Required Textbooks : <ul style="list-style-type: none">• Ian Sommerville, Software Engineering, 9th Ed, Addison-Wesley, 2011.
2. List Essential References Materials : <ul style="list-style-type: none">• Roger S. Pressman, Software Engineering: A practitioner's Approach, 6th ed, McGraw-Hill Science, 2009.
3. List Recommended Textbooks and Reference Material : <ul style="list-style-type: none">• IEEE TRANSACTIONS ON SOFTWARE ENGINEERING JOURNAL.• ANNALS OF SOFTWARE ENGINEERING JOURNAL.• SOFTWARE ENGINEERING JOURNAL.
4. List Electronic Materials : <ul style="list-style-type: none">• http://nptel.ac.in/courses.php?branch=Comp• https://www.coursera.org/
5. Other learning material : <ul style="list-style-type: none">• Video and presentations that available with the instructor.

F. Facilities Required

1. Accommodation <ul style="list-style-type: none">• Classrooms,• Library, and• Laboratories, as those that are available at the college of science at AzZulfi
2. Computing resources <ul style="list-style-type: none">• Smart Board
3. Other resources <ul style="list-style-type: none">• None





G. Course Evaluation and Improvement Processes

<p>1. Strategies for Obtaining Student Feedback on Effectiveness of Teaching:</p> <ul style="list-style-type: none">• Questionnaires (course evaluation) achieved by the students and it is electronically organized by the university.• Student-faculty management meetings.
<p>2. Other Strategies for Evaluation of Teaching by the Program/Department Instructor :</p> <ul style="list-style-type: none">• Discussion within the staff members teaching the course.• Departmental internal review of the course.
<p>3. Processes for Improvement of Teaching :</p> <ul style="list-style-type: none">• Periodical departmental revision of methods of teaching.• Monitoring of teaching activates by senior faculty members.• Training course.
<p>4. Processes for Verifying Standards of Student Achievement</p> <ul style="list-style-type: none">• Instructors of the course are checking together and put a unique process of the evaluation.
<p>5. Describe the planning arrangements for periodically reviewing course effectiveness and planning for improvement :</p> <ul style="list-style-type: none">• Course evaluation• Exam evaluation• Improvement plan

Course Specification Approved

Department Official Meeting No (6) Date **23 / 12 / 1435 H**

Course's Coordinator

Name : Dr. Zeiad El-Saghir

Signature :

Date : 23 / 12 / 1435 H

Department Head

Name : Associate Prof. Yosry Azzam

Signature : 

Date : / ... / H

