

ATTACHMENT 5.

Kingdom of Saudi Arabia
The National Commission for Academic Accreditation &
Assessment

T6. Course Specifications
(CS)

Course Specifications

Institution	Umm Al Qura University	Date	16/04/2016
College/Department	College of Computers and Information Systems		

A. Course Identification and General Information

1. Course title and code: 14013303-3 Software Engineering II			
2. Credit hours 3			
3. Program(s) in which the course is offered. (If general elective available in many programs indicate this rather than list programs) Computer Science			
4. Name of faculty member responsible for the course Dr Basem Al-Kazmi			
5. Level/year at which this course is offered 3rd year / level 7			
6. Pre-requisites for this course (if any) 14013303-3 Software Engineering I			
7. Co-requisites for this course (if any) N/A			
8. Location if not on main campus			
9. Mode of Instruction (mark all that apply)			
a. traditional classroom	<input checked="" type="checkbox"/>	What percentage?	<input type="text" value="100%"/>
b. blended (traditional and online)	<input type="checkbox"/>	What percentage?	<input type="text"/>
c. e-learning	<input type="checkbox"/>	What percentage?	<input type="text"/>
d. correspondence	<input type="checkbox"/>	What percentage?	<input type="text"/>
f. other	<input type="checkbox"/>	What percentage?	<input type="text"/>
Comments:			

B Objectives

<p>What is the main purpose for this course? The course presents the design and architecture part of software engineering, in particular the course focuses on object oriented modelling and UML techniques for specifying software systems. The objectives of the module are</p> <ul style="list-style-type: none"> • introduce agile method for managing software development • present standard UML software analysis and design • introduce key object-oriented design principles • introduce design patterns and demonstrate how they can facilitate the overall development process • Introduce enterprise architecture and SOA
<p>2. Briefly describe any plans for developing and improving the course that are being implemented. (e.g. increased use of IT or web based reference material, changes in content as a result of new research in the field)</p> <ul style="list-style-type: none"> • increased use of web based reference material • changes in content as a result of new research in the field

C. Course Description (Note: General description in the form used in Bulletin or handbook)

<p>Course Description:</p> <p>Overview of software lifecycle, software architectures: styles, quality attributes, design trade-offs, evaluation, specification and tools support; Object-oriented analysis and design using Unified Modelling Language (UML) and patterns.</p>

1. Topics to be Covered		
List of Topics	No. of Weeks	Contact hours

Brief Overview of Software Lifecycle Agile Software Development	1	
Engineering OO Software with Unified Modelling Language (UML)	7	
- Introduction		
- Use case modelling		
- Scenarios		
- Activity diagrams		
- Class analysis and object diagrams		
- Interaction diagrams		
- State diagrams		
- Component and deployment		
Introduction to Software Architecture and Styles	2	
Introduction to Service-oriented Architecture and Cloud SaaS	2	
Software Maintenance and Quality assurance	2	
CMMI and Process Improvement	2	
2. Course components (total contact hours and credits per semester):		
	Lecture	Tutorial
		Laboratory or Studio
		Practical
		Other:
		Total
Contact Hours	: 30	0
Credit		

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

4. Course Learning Outcomes in NQF Domains of Learning and Alignment with Assessment Methods and Teaching Strategy			
On the table below are the five NQF Learning Domains, numbered in the left column.			
<p>First, insert the suitable and measurable course learning outcomes required in the appropriate learning domains (see suggestions below the table). Second, insert supporting teaching strategies that fit and align with the assessment methods and intended learning outcomes. Third, insert appropriate assessment methods that accurately measure and evaluate the learning outcome. Each course learning outcomes, assessment method, and teaching strategy ought to reasonably fit and flow together as an integrated learning and teaching process. (Courses are not required to include learning outcomes from each domain.)</p>			
Code #	NQF Learning Domains And Course Learning Outcomes	Course Teaching Strategies	Course Assessment Methods
1.0	Knowledge		

1.1	Differentiate between different UML models and their syntax	Lectures	Examination, Continuous Assessment
1.2	explain the importance of architecture in distributed software systems, identify styles and patterns, and understand the design trade-offs involved	Lectures	Examination, Continuous Assessment
2.0	Cognitive Skills		
2.1	use UML in the analysis, architecture, design of software systems	Lectures-Project	Examination, Continuous Assessment
3.0	Interpersonal Skills & Responsibility		
3.1	Acquire skills to manage and perform the systems development process more effectively	Lectures-Project	Project
4.0	Communication, Information Technology, Numerical		
4.1	Work in teams and organize tasks	Project	
5.0	Psychomotor		

5. Map course LOs with the program LOs. (Place course LO #s in the left column and program LO #s across the top.) (I = Introduction P = Proficient A = Advanced)

Course LOs #	Program Learning Outcomes (Use Program LO Code #s provided in the Program Specifications)										
	a	b	c	d	e	f	g	h	i	j	k
1.1		P									P
1.2											P
2.1			P						P	P	
3.1	P										
4.1						P					

6. Schedule of Assessment Tasks for Students During the Semester

	Assessment task (e.g. essay, test, group project, examination, speech, oral presentation, etc.)	Week Due	Proportion of Total Assessment
1	Assignment 1	3	10
2	Assignment 2	6	10
3	Mid Term	8	20
4	Project	9	20

5	Final Exam	16	40
---	------------	----	----

D. Student Academic Counseling and Support

<p>1. 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)</p> <p>4 Office hours per week</p>

E Learning Resources

<p>1. List Required Textbooks</p> <p>UML distilled</p>
<p>2. List Essential References Materials (Journals, Reports, etc.)</p> <p>Software Engineering: A Practitioner's Approach, 8th Edition, 2014, Roger Pressman, Bruce Maxim Systems Analysis and Design, by Dennis, Wixom and Roth. ISBN-13: 978-0471722571, 6th Edition. 2014</p>
<p>3. List Recommended Textbooks and Reference Material (Journals, Reports, etc)</p> <p>Requirements Engineering: From System Goals to UML models to Software Specifications Axel van Lamsweerde John Wiley and Sons , 2010 Software Project Management (5th ed.) Bob Hughes and Mike Cotterell McGraw-Hill , 2009 Mastering the requirements process (2cnd ed.) Suzanne and James Robertson Addison-Wesley , 2006 Software Engineering (9th ed.) Ian Sommerville Addison-Wesley , 2011</p>
<p>4. List Electronic Materials, Web Sites, Facebook, Twitter, etc.</p>
<p>5. Other learning material such as computer-based programs/CD, professional standards or regulations and software.</p>

--

F. Facilities Required

Indicate requirements for the course including size of classrooms and laboratories (i.e. number of seats in classrooms and laboratories, extent of computer access etc.)
1. Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.) Lecture room (max 40 students) Computer lab (max 20 students)
2. Computing resources (AV, data show, Smart Board, software, etc.) Rational Rose Software tool
3. Other resources (specify, e.g. if specific laboratory equipment is required, list requirements or attach list) Lecture slides and notes

G Course Evaluation and Improvement Processes

1 Strategies for Obtaining Student Feedback on Effectiveness of Teaching A student-feedback form is distributed at the end of the course.
2 Other Strategies for Evaluation of Teaching by the Instructor or by the Department

Peer Review
3 Processes for Improvement of Teaching Review student feedback and marks with Program coordinator and quality department.
4. Processes for Verifying Standards of Student Achievement (e.g. check marking by an independent member teaching staff of a sample of student work, periodic exchange and remarking of tests or a sample of assignments with staff at another institution) check marking by an independent member teaching staff of a sample of student work
5 Describe the planning arrangements for periodically reviewing course effectiveness and planning for improvement. Submit course report and file at the end of each course offering.

Name of Instructor: _____

Signature: _____ Date Report Completed: _____

Name of Course Instructor _____

Program Coordinator: _____

Signature: _____ Date Received: _____