



University: *Fayoum University*
Faculty: *Computers and Information*
Department: *Information Systems*

Course Specification

1- Basic Information								
Code: CSC 354	Course Title: Software Engineering	Year/Level: Third year – Second term						
Programme: B.Sc degree in Information Systems	Number of units:	<table border="1"> <tr> <td>Lecture:</td> <td>3 hrs/ week</td> </tr> <tr> <td>Tutorial:</td> <td>0 hrs/ week</td> </tr> <tr> <td>Practical:</td> <td>2 hrs/ week</td> </tr> </table>	Lecture:	3 hrs/ week	Tutorial:	0 hrs/ week	Practical:	2 hrs/ week
Lecture:	3 hrs/ week							
Tutorial:	0 hrs/ week							
Practical:	2 hrs/ week							

2- Aims of Course:	<ol style="list-style-type: none"> 1. To understand Software Engineering. 2. To gain basic knowledge software process and software life cycle. 3. To be able to achieve different software artifacts in different software development phases.
---------------------------	---

3- Intended Learning Outcomes	
A- Knowledge and Understanding:	<p>A1) Identify quality criteria that enable future development of computer-based systems.</p> <p>A2) List the Fundamental topics in Computer Science related to software engineering principles, computer organization and architecture</p> <p>A6) Explain essential concepts, principles, and theories related to computer-application development such as databases, information systems development</p> <p>A7) Demonstrate essential facts, concepts, principles and theories relating to computing and information and computer applications as appropriate to the program of study</p> <p>A10) Identify and explain the fundamental concepts, principles, and techniques needed for the analysis, development, validation, verification, deployment, and operations of computer-based systems</p> <p>A11) Describe main concept of operating systems, information system and databases</p> <p>A14) List the professional, moral and ethical issues involved in the exploitation of computer technology and be guided by the appropriate professional, ethical and legal practices relevant to the computing and information industry</p> <p>A15) Demonstrate the extent to which a computer-based</p>

	<p>system meets the criteria defined for its current use and future development.</p> <p>a1) Understanding software process and software evolution. a2) Recognize the different software models.</p>
B- Intellectual Skills:	<p>B2) Determine different computer- system application attributes, components, relationships, patterns, architecture, and source of errors</p> <p>B7) Determine goals for problem solving and test the result of the solution of the problems</p> <p>B8) Identify criteria to measure and interpret the appropriateness of a computer system for its current deployment and future evolution</p> <p>B9) Compare between the classifications of (data, results, methods, techniques, algorithms... etc.).</p> <p>B12) Define the standard methodologies for solving information systems problems</p> <p>b1) Understand difference between Software Engineering and Computer Engineering. b2) Understand costs of Software Engineering. b3) Evaluating attributes of good software</p>
C- Professional and Practical Skills:	<p>C6) Plan, schedule, control, and lead ICT projects</p> <p>c1) Practicing software artifacts/documents development. c2) Practicing system design and system decomposition. c3) Practicing requirements engineering. c4) Project management, Working with CASE/UML tools.</p>
D- General and transferable Skills	<p>D3) Work as a member of a development team, recognizing the different roles within a team and different ways of organizing teams</p> <p>D6) Demonstrate skills in team work, team management, time management and organizational skills</p> <p>d1) Practicing working in teams d2) Attending Competitions with colleagues</p>

4-Course Content:	<ol style="list-style-type: none"> 1. Introduction: Well-engineered software, the software process, software evolution, and software reliability. 2. Human factors in software engineering: Human diversity, knowledge processing, group working.
--------------------------	---

	<ol style="list-style-type: none"> 3. Software specification and system modeling: The software requirements document, requirements evolution, system contexts, 4. Requirements definition and specification: Requirements specification, nonfunctional requirements definition. 5. Requirements validation and prototyping: The prototyping process, prototyping techniques. 6. Formal specifications, algebraic specification. Model based specification. 7. Software design: Top-down design, systems design, design decomposition, 8. software design quality, 9. Design description languages.
--	---

5- Teaching and Learning Methods:	<ol style="list-style-type: none"> 1. Lectures 2. Computer-lab Sessions 3. Practical lab work 4. Class discussions 5. Internet searches 6. Group projects
--	---

6- Teaching and Learning Methods for handicapped students :	-
--	---

7- Student Assessment	
A- Assessment Methods:	<ol style="list-style-type: none"> 1. Assignments and Quizzes 2. Midterm written exam 3. Oral exam 4. Practical exam 5. Final written exam
B- Assessment schedule:	Midterm Examination: Week 7 Practical examination: Week 13 Oral Examination: Week 14 Final Examination: Week 15
C- Weighting of assessments:	Assignments and Quizzes: 0% Mid-Term Examination: 10% Oral Examination: 10% Practical Examination: 15% Final-term Examination: 65%

8- Books and References	
A- Notes:	Carefully selected, well reviewed in core presentations
B- Essential Books (Text Books):	<ul style="list-style-type: none"> ▪ Ian Sommerville, "Software Engineering", Addison-Wesley publisher. 10th edition (2016)
C- Recommended Books:	<ul style="list-style-type: none"> ▪ Roger S. Pressman, "Software Engineering, A Practitioner's Approach", Higher Education (2008)

	<ul style="list-style-type: none">▪ Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides: “Design Patterns”, Addison-Wesley. 14th edition (2016)▪ Grady Booch, James Rumbaugh, Ivar Jacobson, “The Unified Modeling Language User Guide”, Addison Wesley. (2012)
D- Periodicals, Web sites, ... etc	-

Course Professor: Dr. Amira Edress Department Head: Dr. Amira Edress

Course Content Intended Learning Outcomes Matrix**Course Title:** Software Engineering**Course Code:** CSC 354

Course Content	Week	Knowledge & Understanding		Intellectual Skills			Professional & Practical Skills				General & Transferable Skills	
		a1	a2	b1	b2	b3	c1	c2	c3	c4	d1	d2
1. Introduction: Well-engineered software, the software process, software evolution, and software reliability.	1	x		x		x	x				x	x
2. Human factors in software engineering: Human diversity, knowledge processing, group working.	2	x				x					x	x
3. Software specification and system modeling: The software requirements document, requirements evolution, system contexts,	3	x	x				x		x		x	x
4. Project Management	4	x	x						x	x	x	x
5. Requirements definition and specification: Requirements specification, nonfunctional requirements definition.	5	x			x				x		x	x
6. Requirements validation and prototyping: The prototyping process, prototyping techniques.	6	x							x		x	x
7. Model based specification.	7		x				x				x	x
8. Midterm Exam	8											
9. Software design: Top-down design, systems design, design decomposition,	9	x	x					x			x	x
10. software design quality,	10				x	x					x	x
11. Introducing Agile Methodology	11	x		x		x	x				x	x

Course coordinator: Dr. Amira Edress

Head of Department: .Dr. Amira Edress