



**COLLEGE OF INFORMATION TECHNOLOGY**  
**DEPARTMENT OF IT**  
**COURSE SYLLABUS/ SPECIFICATION**

<b>Course Code &amp; Title:</b>	<b>ITCS 313 – Software Engineering I</b>
<b>Weight:</b>	<b>(2-2-3)</b>
<b>Prerequisite:</b>	<b>ITCS 201</b>
<b>NQF Level Allocated:</b>	<b>7</b>

**NQF Notional Hours / Credits: 120 notional hours/ 12 NQF credit**

**Description:** This course is to give a clear understanding of the concepts of software engineering. It imparts knowledge of developing a software system from scratch, different software process models, software requirement engineering, and software design with object oriented technology using UML.

**Objective:**

1. To understand software process models and how to select the suitable model to be used in software development.
2. To develop the skills to gather different kinds of user requirements, various requirements modeling techniques, and requirements verification and validation.
3. To explain the process of object oriented design, and modeling techniques using UML.
4. To translate a requirements specification into an implementable design, following a structured and organized process.
5. To help students gain experience of working as a member of a software engineering project team.

Semester:

Instructor (s):

Office Telephone: EXT:

Email (s):

**Intended Learning Outcomes (ILOs):**

<b>A. Knowledge and Understanding</b>		<b>NQF Descriptor/ Level</b>
<b>A1</b>	<b>Concepts and Theories:</b> Demonstrate advanced knowledge and understanding of the software engineering concepts such as software development process and life cycle, object-oriented design methodology.	Knowledge: theoretical understanding [Level 7]
<b>A2</b>	<b>Contemporary Trends, Problems and Research:</b> N/A	
<b>A3</b>	<b>Professional Responsibility:</b> N/A	
<b>B. Subject-specific Skills</b>		<b>NQF Descriptor/ Level</b>
<b>B1</b>	<b>Problem Solving:</b> Identify real world problems, elicit the requirements of stakeholders, design solutions, and test them against user requirements.	Knowledge: Practical Application [Level 7] Skills: Communication, ICT & Numeracy [Level 7]
<b>B2</b>	<b>Modeling and Design:</b> Model using UML and design a computer-based system architecture, process and component needed to solve real world problem.	Knowledge: Practical Application [Level 7]
<b>B3</b>	<b>Application of Methods and Tools:</b> Apply the appropriate Computer Aided Software Engineering (CASE) tools to model and design computer software effectively.	Knowledge: Practical Application [Level 7] Skills: Communication, ICT & Numeracy [Level 7]
<b>C. Critical-Thinking Skills</b>		<b>NQF Descriptor/ Level</b>
<b>C1</b>	<b>Analytic skills:</b> Critically analyze a problem, identify needs and define the computing requirements appropriate to its solution.	Generic Problem Solving & Analytical skills [Level 7]

<b>C2</b>	<b>Synthetic:</b> Review, and extend knowledge and skills in software development to construct and reuse the multiple components of software.	Generic Problem Solving & Analytical skills [Level 7]
<b>C3</b>	<b>Creative:</b> N/A	

<b>D. General and Transferable Skills (other skills relevant to employability and personal development)</b>		<b>NQF Descriptor/ Level</b>
<b>D1</b>	<b>Communication:</b> Organize and relate ideas effectively, in written, oral, and graphical form.	Communication, ICT and Numeracy Skills [Level 7]
<b>D2</b>	<b>Teamwork and Leadership:</b> Function and work effectively as member/leader of a software development project team.	Competence: Autonomy, Responsibility and Context [Level 7]
<b>D3</b>	<b>Organizational and Developmental Skills:</b> Demonstrate the ability to effectively manage and allocate appropriate time to develop computer software.	Competence: Autonomy, Responsibility and Context [Level 7]
<b>D4</b>	<b>Ethical and Social Responsibility:</b> N/A	

## Course Structure (Outline)

Week	Hours		ILOs	Unit/Module or Topic Title	Teaching Method	Assessment Method
	Lec.	Lab				
1	2	2	A1	<b>Introduction:</b> -Software Engineering -Software Engineering Problems.	Lecture	
2	2	2	A1	<b>Software Development Life Cycle</b>	Lecture/ Class Discussion	
3	2	2	A1, C1	<b>Software Process Models</b>	Lecture/ In- Class supervised Work	Quiz 1/ Case Study
4	2	2	A1, B1, C1	<b>Software Analysis Phase:</b> -Requirements Engineering process. -Requirements Elicitation. -Types of Requirements. -Fishbone Diagram	Lecture/ In- Class Supervised Work/ Lab Demonstration	In-Class Exercises
5	2	2	B2, B3, C1	<b>Software Analysis Phase:</b> -Business Process. -Functional Modeling. -DFD. -Use Case Diagrams.	Lecture/ In- Class Supervised Work/ In- Lab Supervised Work	In-Lab/ In-Class Exercises/ Quiz 2
6	2	2	B2, B3, C1	<b>Software Analysis Phase:</b> Business Process Modeling With Activity Diagrams.	Lecture/ In- Class Supervised Work/ In- Lab Supervised Work	In-Class/ In-Lab Exercises

7	2	2	B2, B3, C1	<b>Software Analysis Phase:</b> -Structured Modeling. -CRC Cards.	Lecture/ In- Class Supervised Work/ In- Lab Supervised Work	In-Class/ In-Lab Exercises
8	2	2	A1, B2, B3, C1	<b>Software Analysis Phase:</b> Class Diagram.	Lecture/ In- Class Supervised Work/ In- Lab Supervised Work	In-Lab Exercises/ Quiz 3
9	2	2	A1, B2, B3, C1	<b>Software Analysis Phase:</b> Behavioral Modeling.	Lecture/ In- Class Supervised Work/ In-Lab Supervised Work	Major Test
10	2	2	A1, B2, B3, C1	<b>Software Analysis Phase:</b> Interaction Diagram.	Lecture/ In- Class Supervised Work/ In- Lab Supervised Work	In-Class/ In-Lab Exercises
11	2	2	B2, C2	<b>Software Design Phase:</b> Evolving The Analysis Models into Design Models.	Lecture/ In- Class Supervised Work	In-Class Exercises
12	2	2	B1, B2, C2	<b>Software Design Phase:</b> Class and Method Design.	Lecture/ In- Class Supervised Work	In-Class Exercises

13	2	2	B1, B2, C1, C2	<b>Software Design Phase:</b> User Interface Design and Dialog Diagram.	Lecture/ In-Class Supervised Work	In-Class Exercises
14	2	2	B1, B2, C1, C2	<b>Software Design Phase:</b> Architecture Design.	Lecture/ In-Class Supervised Work	Case Study
15	2	2	B1, B2, B3, C1, C2, D1, D2, D3	Student Projects	Projects Supervision	Evaluation of Project Presentation and Report
16	2	-	A1, B1, B2,	All Topics		Final Exam

\* Formative assessment

### Teaching Materials:

<b>Textbook(s):</b>	<ol style="list-style-type: none"> <li>1. Tagarden D. and Dennis A. (2015) <i>Systems Analysis and design with UML</i>, 5<sup>th</sup> Edition, Wiley.</li> <li>2. Sommerville I. (2015) <i>Software engineering</i>, 10<sup>th</sup> Edition, Pearson.</li> </ol>
<b>Handout(s):</b>	PowerPoint slides available on Moodle i.e. <a href="http://www.ahlia.edu.bh/moodle">http://www.ahlia.edu.bh/moodle</a>
<b>Reference(s):</b>	<ol style="list-style-type: none"> <li>1. Bennett S., McRobb S. and Frammer R. (2010) <i>Object-Oriented System Analysis and Design using UML</i>, 4th Edition, McGraw Hill.</li> <li>2. Pressman R. and Maxim B. (2014) <i>Software Engineering: A Practitioner's Approach</i>, 8th Edition, McGraw-Hill.</li> <li>3. Booch G., Rumbaugh J. and Jacobson I. (2005) <i>The Unified Modeling Language User Guide</i>, 2nd Edition, Addison Wesley.</li> <li>4. Bruegge B. and Dutoit A. H. (2013) <i>Object-Oriented Software Engineering: Using UML, Patterns and Java</i>, 3rd Edition, Pearson.</li> <li>5. More references are available in the course website on Moodle.</li> </ol>

## Assessment

Method of Assessment	Description	Learning Outcomes	Weighting
In-Lab Exercises	Each of the In-Lab exercises consist of a set of practical tasks to be implemented by students individually in class, as shown in the above weekly structure. Students work will be observed and graded directly during the lab sessions.	B3	10%
In-Class Exercises	In-class exercises consist of problem modeling and solving.	B2, C1	Formative
Case Studies	Different software development project cases are analyzed and studied.	C1	Formative
Quizzes	The purpose of the quiz is to assess the students' knowledge and understanding of concepts, principles and specialist theories of the topics. Students will be given three quizzes, best two will be considered.	A1	10%
Major Test	The test will be an in-class 90 minutes exam that will consist of short-answer, essay, problem solving, and analysis questions and cover the topics studied in the first 8 weeks.	A1, B1, B2, C1	20%
Project	Starting from week 4, the class will be divided into teams of 2-3 students where each team will be asked to develop software of good characteristics. In this project the students will be able to analyze problems and specify requirements, model and design software using UML.	B1, B2, B3, C1, C2, D1, D2, D3	20%
Final Exam	The final exam is comprehensive and will be of two hours duration. It will consist of short-answer, essay and problem-solving questions.	A1, B1,B2, C1, C2	40%
<b>Overall:</b>			<b>100 %</b>

<b>Admissions</b>	
<b>Minimum number of students</b>	<b>5</b>
<b>Maximum number of students</b>	<b>20</b>

**Ahlia University values academic integrity. Therefore, all students must understand the meaning and consequences of cheating, plagiarism and other academic offences under the Code of Student Conduct and Disciplinary Procedures (see [www.ahlia.edu.bh/integrity](http://www.ahlia.edu.bh/integrity) for more information).**