

## COURSE OUTLINE

### (1) GENERAL

<b>SCHOOL</b>	Science and Technology		
<b>ACADEMIC UNIT</b>	Science and Technology		
<b>PROGRAMME OF STUDIES</b>	MSc in Mobile and Web Computing, MSc in ICT Systems		
<b>LEVEL OF STUDIES</b>	Postgraduate		
<b>COURSE CODE</b>	ME01, CC06	<b>SEMESTER</b>	2
<b>COURSE TITLE</b>	Software Development Methodologies		
<b>COURSE TYPE</b> <i>Elective, compulsory</i>	Compulsory		
<b>INSTRUCTOR(S)</b>	Assist. Prof. Christos Tjortjis		
<b>INDEPENDENT TEACHING ACTIVITIES</b> <i>if credits are awarded for separate components of the course, e.g. lectures, laboratory exercises, etc. If the credits are awarded for the whole of the course, give the weekly teaching hours and the total credits</i>		<b>WEEKLY TEACHING HOURS</b>	<b>CREDITS</b>
		4	6
<i>Add rows if necessary. The organisation of teaching and the teaching methods used are described in detail at (d).</i>			
<b>TEACHING ACTIVITIES BREAKDOWN</b>		<b>WEEKLY HOURS</b>	
Theory		3,3	
Recitation		0,7	
<b>COURSE TYPE</b> <i>general background, special background, specialised general knowledge, skills development</i>	Special background		
<b>PREREQUISITE COURSES:</b>	-		
<b>LANGUAGE OF INSTRUCTION and EXAMINATIONS:</b>	English		
<b>IS THE COURSE OFFERED TO ERASMUS STUDENTS</b>	Yes		
<b>COURSE WEBSITE (URL)</b>	<a href="https://elearn-ucips.ihu.gr/">https://elearn-ucips.ihu.gr/</a>		

### (2) LEARNING OUTCOMES

<p><b>Learning outcomes</b></p> <p><i>The course learning outcomes, specific knowledge, skills and competences of an appropriate level, which the students will acquire with the successful completion of the course are described.</i></p> <p><i>Consult Appendix A</i></p> <ul style="list-style-type: none"> <li>• <i>Description of the level of learning outcomes for each qualifications cycle, according to the Qualifications Framework of the European Higher Education Area</i></li> <li>• <i>Descriptors for Levels 6, 7 &amp; 8 of the European Qualifications Framework for Lifelong Learning and Appendix B</i></li> <li>• <i>Guidelines for writing Learning Outcomes</i></li> </ul> <p>On completing the course, the student will be able to:</p> <ul style="list-style-type: none"> <li>• Appreciate the principles, concepts and techniques for effective software development</li> <li>• Apply effective software engineering methods, tools and techniques</li> <li>• Plan, manage and collaborate on a software development team project</li> <li>• Expose the knowledge and skills required to effectively manage the software maintenance process</li> <li>• Effectively apply software engineering, management and communication skills</li> </ul>
--

<p><b>General Competences</b></p> <p><i>Taking into consideration the general competences that the degree-holder must acquire (as these appear in the Diploma Supplement and appear below), at which of the following does the course aim?</i></p> <p><i>Search for, analysis and synthesis of data and information, with the use of the necessary technology</i>      <i>Project planning and management</i>  <i>Adapting to new situations</i>      <i>Respect for difference and multiculturalism</i>  <i>Decision-making</i>      <i>Respect for the natural environment</i>  <i>Working independently</i>      <i>Showing social, professional and ethical responsibility and sensitivity to gender issues</i>  <i>Team work</i>      <i>Criticism and self-criticism</i>  <i>Working in an international environment</i>      <i>Production of free, creative and inductive thinking</i>  <i>Working in an interdisciplinary environment</i>      .....  <i>Production of new research ideas</i>      <i>Others...</i>  .....</p>	
<ul style="list-style-type: none"> <li>● Search for, analysis and synthesis of data and information, with the use of the necessary technology</li> <li>● Decision Making</li> <li>● Teamwork</li> <li>● Production of free, creative, and inductive thinking</li> </ul>	

### (3) SYLLABUS

<p>The course introduces fundamental concepts and tools related to Software Development Methodologies:</p> <ul style="list-style-type: none"> <li>● Principles, techniques, methods and tools of software development</li> <li>● Requirements analysis</li> <li>● Unified Modeling Language (UML)</li> <li>● Object-oriented analysis, design and programming</li> <li>● Application Programming Interface (API)</li> <li>● Software maintenance and development</li> <li>● Project and version management</li> </ul>
---

### (4) TEACHING and LEARNING METHODS - EVALUATION

<p><b>DELIVERY</b></p> <p><i>Face-to-face, Distance learning, etc.</i></p>	<p>Hybrid: Face to face and synchronous distance learning</p>	
<p><b>USE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY</b></p> <p><i>Use of ICT in teaching, laboratory education, communication with students</i></p>	<p><b>Use of ICT in Teaching</b></p> <p>During the educational process, various programming tools are used, along with the material available at the e-learning platform. The hybrid teaching method involves synchronous learning with the support of the videoconferencing tool Zoom. Students are taught a variety of tools related to the course content and material.</p> <p><b>Use of ICT in Communication with students</b></p> <ul style="list-style-type: none"> <li>● The course material (slides, scientific articles, exercises, etc.) is posted on the course page at the e-learn platform (Moodle).</li> <li>● Use of Moodle Forums announcements.</li> <li>● Live video meetings via Zoom/Teams.</li> <li>● Contact via email.</li> </ul>	
<p><b>TEACHING METHODS</b></p> <p><i>The manner and methods of teaching are described in detail.</i></p>	<p><i>Activity</i></p>	<p><i>Semester workload</i></p>
	Lectures	30 hrs.
	Recitation	10 hrs.

<p>Lectures, recitation, seminars, laboratory practice, fieldwork, study and analysis of bibliography, tutorials, placements, clinical practice, art workshop, interactive teaching, educational visits, project, essay writing, artistic creativity, etc.</p> <p>The student's study hours for each learning activity are given as well as the hours of non-directed study according to the principles of the ECTS</p>	Non-Directed Study	70 hrs.
	Project	30 hrs.
	Exams and Corrections	10 hrs.
	<b>Course total</b>	<b>150 hrs.</b>
<p><b>STUDENT PERFORMANCE EVALUATION</b></p> <p>Description of the evaluation procedure</p> <p>Language of evaluation, methods of evaluation, summative or conclusive, multiple choice questionnaires, short-answer questions, open-ended questions, problem solving, written work, essay/report, oral examination, public presentation, laboratory work, clinical examination of patient, art interpretation, other</p> <p>Specifically-defined evaluation criteria are given, and if and where they are accessible to students</p>	<p>Language of Evaluation: English</p> <p>Evaluation Procedure:</p> <ul style="list-style-type: none"> <li>● Written Exams (50%). Methods of evaluation: <ul style="list-style-type: none"> <li>○ Open-ended questions</li> <li>○ Problem solving</li> <li>○ Multiple choice questions (on lab material)</li> </ul> </li> <li>● Projects (50%): <ul style="list-style-type: none"> <li>○ The students should achieve a passing grade to participate in the written exams.</li> </ul> </li> </ul> <p>The evaluation procedure is announced to the students during the first lecture and is also accessible at the e-learn platform throughout the entire semester.</p>	
<p><b>STUDENT OBLIGATIONS</b></p> <p>Compulsory attendance of lectures, labs, recitations, compulsory participation in midterms, exams, compulsory delivery of homework, projects, etc.</p>	<ul style="list-style-type: none"> <li>● Compulsory attendance of lectures</li> <li>● Compulsory attendance of recitation</li> <li>● Compulsory participation in the exams</li> <li>● Compulsory delivery of project</li> </ul>	

## (5) ATTACHED BIBLIOGRAPHY

<ol style="list-style-type: none"> <li>1. D. Avison, G. Fitzgerald, Information Systems Development methodologies, techniques and tools, 4e, McGraw Hill, 2006.</li> <li>2. A. Dennis, B.H. Wixom, D. Teagarden, Systems Analysis and Design: An Object-Oriented Approach with UML, Wiley, 5th ed., 2015.</li> <li>3. S. Bennett, S. McRobb, R. Farmer, Object-Oriented Systems Analysis and Design, 4/e ed., McGraw Hill, 2011.</li> <li>4. B. Oestereich, Developing software with UML: object-oriented analysis and design in practice, 2nd ed. Addison Wesley, 2002.</li> <li>5. M. O' Docherty, Object-Oriented Analysis &amp; Design. Understanding System Development with UML 2.0, Wiley, 2005.</li> <li>6. R.S. Pressman, Software Engineering- A Practitioner's Approach, 8/e ed. McGraw Hill, 2015.</li> <li>7. I. Sommerville, Software Engineering, 10<sup>th</sup> ed. Addison-Wesley, 2016.</li> <li>8. Class handouts</li> </ol>
--