

<b>INSTITUTION</b>	NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS																				
<b>SCHOOL</b>	SCHOOL OF SCIENCE																				
<b>DEPARTMENT</b>	INFORMATICS AND TELECOMMUNICATIONS																				
<b>COURSE LEVEL</b>	UNDERGRADUATE																				
<b>COURSE TITLE</b>	<b>Software Development for Information Systems</b>																				
<b>COURSE CODE</b>	K23a	<b>Semester</b>	7	<b>ECTS</b>	8																
<b>TEACHING HOURS per week</b>	<b>THEORY</b>	1	<b>SEMINAR.</b>		<b>LABORATORY</b>	3															
<b>COURSE TYPE</b>	<p>Select one of the following and delete the rest Project</p> <table border="1"> <thead> <tr> <th>K</th> <th>E1</th> <th>E2</th> <th>E3</th> <th>E4</th> <th>E5</th> <th>E6</th> </tr> </thead> <tbody> <tr> <td>B</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>							K	E1	E2	E3	E4	E5	E6	B						
K	E1	E2	E3	E4	E5	E6															
B																					
<b>URL</b>	<a href="https://eclass.uoa.gr/courses/D29/">https://eclass.uoa.gr/courses/D29/</a>																				
<b>EXPECTED PRIOR KNOWLEDGE/ PREREQUISITES AND PREPARATION:</b>	K18 - Implementation of Database Management Systems, Recommended K29																				
<b>TEACHING AND EXAMINATIONS LANGUAGE:</b>	GREEK																				
<b>THE COURSE IS OFFERED TO ERASMUS STUDENTS</b>	NO																				

<b>COURSE CONTENT</b>
<p>Extensive implementation of software systems, based on the contents of other courses, in three consecutive phases: a) Implementation of a simplified form of various internal layers of a Database Management System: block and record organization on disk, a static data structure on a file on disk (e.g., static hash table), a dynamic data structure on a file on disk (e.g., B+ tree), system catalogs, query processing for a simple database query language. b) Implementation of an application on top of some commercial Database Management System or the Internet or software based on some other technology, c) Adding multithreading support and other optimization features to the application of the previous phases. Emphasis is given in software development techniques, i.e. use of source control systems (git), unit testing, etc. The project concept is usually inspired from the previous SIGMOD programming contest.</p> <p>A team of selected students take part as a NKUA team to the current year SIGMOD programming contest, and are evaluated based on their participation and effectiveness in the competition.</p>

--

STUDENT LEARNING OBJECTIVES
<p>Teaching-Learning Goals-Expected Learning Outcomes</p> <p>Upon successful completion of the course the student will be able to:</p> <ul style="list-style-type: none"> <li>• Design and implement applications and software systems using C or C++.</li> <li>• Design and implement data structures for database systems.</li> <li>• Use source control system for maintaining their code.</li> <li>• Create and maintain unit tests for their code</li> <li>• Optimize their code performance using software tools</li> <li>• Work as a team member developing software</li> <li>• Record and analyze the technical and functional requirements of information systems software</li> </ul>

TEACHING AND LEARNING METHODS - ASSESSMENT											
<b>TEACHING METHOD</b>	In Class (Face to Face)										
<b>USE OF INFORMATION AND COMMUNICATION TECHNOLOGIES</b>	<p>Learning process supported by the e-class platform (specify which specific tools you use, eg Hardware delivery, Discussions, Announcements, Task assignments, Student groups)</p> <p>Email communication</p> <p>Live transmission of lectures</p> <p>Ability to track recorded lectures</p> <p>Utilization of educational environments (please specify name and http)</p> <p>Utilization of Specialized Software</p>										
<b>TEACHING ORGANIZATION</b>	<p><i>Describe in detail the way and methods of teaching:</i> Enhanced Lectures, Online Lectures, Seminars, Tutorial, Laboratory, Laboratory Exercise, Study &amp; analysis of literature, Practice (Positioning), Interactive teaching, Developing a project, Individual / group work Telework (reference to tools) etc.</p> <p><i>Details of the student's study hours for each learning activity and hours of non-guided study are shown to ensure</i></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2">A series of lectures and tutorials introduce the students to concepts, technologies and software development tools that will be used for the project. The project goal is to develop a system optimizing the operation of an information system. The project concept is inspired by the previous SIGMOD programming contest. Student teams of 2 or 3 persons are created, depending on the project's requirements.</td> </tr> <tr> <th style="width: 60%;">Activity</th> <th style="width: 40%;">Student Workload (hours)</th> </tr> <tr> <td>Lectures</td> <td>12</td> </tr> <tr> <td>Tutorial</td> <td>9</td> </tr> <tr> <td>Laboratory</td> <td>3</td> </tr> </table>	A series of lectures and tutorials introduce the students to concepts, technologies and software development tools that will be used for the project. The project goal is to develop a system optimizing the operation of an information system. The project concept is inspired by the previous SIGMOD programming contest. Student teams of 2 or 3 persons are created, depending on the project's requirements.		Activity	Student Workload (hours)	Lectures	12	Tutorial	9	Laboratory	3
A series of lectures and tutorials introduce the students to concepts, technologies and software development tools that will be used for the project. The project goal is to develop a system optimizing the operation of an information system. The project concept is inspired by the previous SIGMOD programming contest. Student teams of 2 or 3 persons are created, depending on the project's requirements.											
Activity	Student Workload (hours)										
Lectures	12										
Tutorial	9										
Laboratory	3										

that the total workload at the semester corresponds to the ECTS	Teamwork in a case study	200	
	Independent Study	26	
	<b>Total Course (25 hours of workload per unit of credit)</b>	<b>250</b>	
<p><b>ASSESSMENT OF STUDENTS</b> <i>Description of the assessment process</i></p> <p><i>Assessment Methods, Formative or Concluding, Multiple Choice Test, Quick Response Questions, Test Development Questions, Problem Solving, Written Work, Report / Report, Oral Examination, Public Presentation, Laboratory Work, Other / Other</i></p> <p><i>Fully defined evaluation criteria are mentioned and if and where they are accessible to students.</i></p>	<p>Describe explicitly methods, evaluation tools and provided feedback. The table below is supplemented accordingly. <i>The evaluation is performed (a) with the delivery of a software system in 3 phases, which is presented and orally examined in the lab, and (b) with an online programming evaluation in the lab</i> <i>The final grade is the evaluation of the software system. The oral examination in the first two phases is used to provide feedback on the course of the development and to determine the progress of the team. Teams that have not met the minimum set requirements cannot continue. After delivering the final software system, the online programming evaluation is used to determine a pass/fail review of each student. Students who fail the programming evaluation, fail the class. Students who pass the programming evaluation, receive the software system grade.</i></p>		
	<b>Assessment methods</b>	<b>Number</b>	<b>Percentage</b>
	Oral examination/delivery of 1 <sup>st</sup> phase software system	1	-
	Oral examination/delivery of 2nd phase software system	1	-
	Oral examination/delivery of 2nd phase software system	1	100%
	Programming Evaluations	1	-
	Final		As described above

**LITERATURE AND STUDY MATERIALS / READING LIST**

No specific textbook. The instructor suggests educational resources online as well as manuals of the technologies, software systems and software development tools used.