ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικόν και Καποδιστριακόν
Πανεπιστήμιον Αθηνών
ΙΔΡΥΘΕΝ ΤΟ 1837

COURSE SYLLABUS

DEPARTMENT OF
INFORMATICS &
TELECOMMUNICATIONS

<table>
<tr><td><b>SCHOOL</b></td><td colspan="6">SCHOOL OF SCIENCE</td></tr>
<tr><td><b>DEPARTMENT</b></td><td colspan="6">INFORMATICS AND TELECOMMUNICATIONS</td></tr>
<tr><td><b>COURSE LEVEL</b></td><td colspan="6">UNDERGRADUATE</td></tr>
<tr><td><b>COURSE TITLE</b></td><td colspan="6"><b>Analysis and Design of Software Systems</b></td></tr>
<tr><td><b>COURSE CODE</b></td><td colspan="2"><b>ΥΣ04</b></td><td><b>SEMESTER</b></td><td><b>6</b></td><td><b>ECTS</b></td><td><b>6</b></td></tr>
<tr><td><b>TEACHING HOURS per week</b></td><td><b>THEORY</b></td><td>3</td><td><b>SEMINAR.</b></td><td></td><td><b>LABORATORY</b></td><td>1</td></tr>
<tr><td><b>COURSE TYPE</b></td><td colspan="6"><b>Select one of the following and delete the rest</b>
Elective (ΠΜ)<table><tr><th>K</th><th>E1</th><th>E2</th><th>E3</th><th>E4</th><th>E5</th><th>E6</th></tr><tr><td>A</td><td></td><td></td><td>B</td><td></td><td></td><td></td></tr></table></td></tr>
<tr><td><b>URL</b></td><td colspan="6">https://eclass.uoa.gr/courses/D50/</td></tr>
<tr><td><b>PREREQUISITES</b></td><td colspan="6">K10 Object Oriented Programming</td></tr>
<tr><td><b>TEACHING AND EXAMINATIONS LANGUAGE</b></td><td colspan="6">GREEK and ENGLISH for ERASMUS students (ERASMUS students study English books, do project work and take written exams in English)</td></tr>
<tr><td><b>THE COURSE IS OFFERED TO ERASMUS STUDENTS</b></td><td colspan="6">YES</td></tr>
</table>

COURSE CONTENT

The course content has been structured as follows: Traditional and Modern Techniques and Methodologies for software development. Problem Analysis. Techniques for requirements capture. The IEEE standard for Requirements Specifications. Object-Oriented Analysis and Design: The Unified Modeling Language (UML); use case diagrams, class diagrams, state diagrams, activity diagrams, collaboration diagrams, sequence diagrams, deployment diagrams, packet diagrams. Function - Oriented Analysis and Design: Data Flow Diagrams, Process Specifications (Structured English, Decision Tables, Decision Tress, Data Dictionaries). Reference to Data Analysis and Entity Relationship Model. Reference to Requirements Testing. Software Architectures, Software Design, Principles of Software Design, Coupling and Cohesion. Structured Design, Structured Chart, Principles for Data Design and User Interface Design. Structure Chart. Transform and Transaction Analysis. Creation of Pseudocode. Reference to Methodologies for systems analysis and design. Computer Aided Software Engineering (CASE) tools.

STUDENT LEARNING OBJECTIVES

The course objectives are: a) to enable students to understand the concepts of Analysis and Design of Software Systems, their relation to Software Engineering and Software Life Cycle, and their importance in software development and b) to learn how to capture, analyze and create requirements specifications and design specifications according to i) Object Oriented Analysis and Design and ii) Structured Analysis and Structured Design.

Upon successful completion of the course the student will be able to:

- Describe the phases of software life cycle
- Identify the difference between user requirements and design requirements and to define their place in the software life cycle
- Identify the differences between object-oriented analysis and design and structured analysis and structured design of software systems
- Capture, Analyse and Evaluate user requirements for software development
- Develop software requirements specifications and software design specifications
- Identify the difference between software architecture design and detailed software design
- Create UML diagrams, Data Flow Diagrams, Software Architecture Diagrams and Structured Charts
- Validate user requirements specifications
- Verify software design specifications

TEACHING AND LEARNING METHODS - ASSESSMENT	
TEACHING METHOD	In Class (Face to Face)
USE OF INFORMATION AND COMMUNICATION TECHNOLOGIES	Learning process supported by the e-class platform (specifically: course description, teaching material, announcements, diary, Course Assignments and Assignment Upload, Forums for Lab Work and Course Assignments, Questionnaires, External links) Email communication Laboratory for using Computer Aided Software Engineering (CASE) tools for developing diagrams for analysis and design software specifications.
TEACHING ORGANIZATION *Describe in detail the way and methods of teaching:* *Enhanced Lectures,* *Online Lectures,* *Seminars,* *Tutorial,* *Laboratory,* *Laboratory Exercise,* *Study & analysis of literature,* *Practice (Positioning),* *Interactive teaching,* *Developing a project,* *Individual / group work* *Telework (reference to tools) etc.* *Details of the student's study hours for each learning activity and hours of non-guided study are shown to ensure that the total workload at the semester corresponds to the ECTS*	The theoretical part of the course is taking place in a typical classroom using slides, videos and teaching methods that promote students active involvement and the elicitation of their prior knowledge such as brainstorming, Q & A, discussion and activities to introduce them in the subject. The lab part is taking place in a Microsoft Windows lab where dedicated software is installed. The students are engaged in activities that give them the chance to familiarize themselves in the use of CASE tools for the creation of various diagrams for software analysis and design. The students also are assigned a project that they carry out as a group of 2-3 students; the project work is compulsory and is related to the construction of diagrams related to the software requirements and design specifications.

The project work and the labs are supported by specialized Forums in e-class.

Activity	***Student Workload*** *(hours)*
Lectures	39
Laboratory	13
Study of Analysis and Design Principles	10
Study of UML Diagrams	15
Study the theory and tools of Structured Analysis and Structured Design	15
Project Assignment (2-3 students)	38
Preparation for the exams	20
Total Course Work	***150***

ASSESSMENT OF STUDENTS

Description of the assessment process

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

Fully defined evaluation criteria are mentioned and if and where they are accessible to students.

Evaluation of theory by written examination and by a project assignment (by teams of 2-3 students).
Feedback for the written examination.
In the project a rubric is given and feedback is provided accordingly with explanations in case of missing grades.

Assessment methods	***Number***	***Percentage***
Written examination	1	70%
Project Assignment	1	30%

LITERATURE AND STUDY MATERIALS / READING LIST

Study Materials:

«UML Distilled: A Brief Guide to the Standard Object Modeling Language», Martin Fowler, 3rd Edition, Addison Wesley.

«Αντικειμενοστρεφής ανάπτυξη λογισμικού με τη UML», Συγγραφείς: Συγγραφείς: Γερογιάννης Β., Κακαρόντζας Γ., Καμέας Α., Σταμέλος Ι, Φιτσιλής Π., Εκδόσεις Κλειδάριθμος

Reading List:

1. “Systems Analysis and Design: An Object-Oriented Approach with UML”, 5th Edition by A. Dennis, B. Haley Wixom and D. Tegarden, Wiley Pubs, ISBN-13: 978-1118804674
2. “Analysis and Design of Information Systems”, by Arthur M. Langer, Springer, ISBN-13: 978-1846286544
3. “Structured Analysis and System Specification”, by T. DeMarco and P. J. Plauger, Yourdon Press, ISBN-13: 978-0138543808
4. “Structured Design: Fundamentals of a Discipline of Computer Program and Systems Design”, by E. Yourdon and L. Constantine, Yourdon Press, ISBN-13: 978-0138544713
5. Course Notes on e-class on Structured Systems Analysis and Structured Design