



Course Syllabus

Course Code	Course Title	ECTS Credits
COMP-512DL	Software Engineering	10
Prerequisites	Department	Semester
None	Computer Science	Fall
Type of Course	Field	Language of Instruction
Elective	Computer Science	English
Level of Course	Lecturer(s)	Year of Study
2 nd Cycle	Prof. A. Kokkinaki	2 nd
Mode of Delivery	Work Placement	Corequisites
Distance Learning	N/A	None

Course Objectives:

This course systematically addresses the process of developing high-quality software products by employing principles, methodologies and best practices. This course is concerned with all aspects of software development, from the early stages of requirements specification to maintaining the system after it has gone into use. This includes technical processes of software development, activities supporting software project management and the development of tools, methods, and theories in software engineering with emphasis on object-oriented software engineering. More specifically, the course aims to build concepts and competencies related to:

- elicitation and documentation of requirements,
- identification of appropriate design techniques,
- construction of high-quality software products,
- employment of effective testing methods,
- software maintenance,
- configuration management,
- software engineering management,
- software engineering processes,
- software engineering methods and models
- software quality aspects
- software engineering economics and
- software professional practices

Learning Outcomes:

After completion of the course students are expected to be able to:

1. Select, with justification the software development models and process elements most appropriate for the development and maintenance of a diverse range of software products.
2. Apply key elements and common methods for elicitation and analysis to produce a set of software requirements for a large-scale software system.
3. Use the users' requirements by the users for the object oriented design and development a large-scale software system.
4. Apply the object-oriented concepts software for developing design models.
5. Understand the potentials of object-oriented methodology, analysis and design of software (i.e. the ICONIX methodology) and apply it in practice.
6. Construct high quality products.
7. Reason and select appropriate processes, methods and models for each software engineering project assigned.
8. Evaluate the quality of multiple software designs based on key design principles and concepts.
9. Systematically assess the economics of software development projects.
10. Adhere to high professional standards and abide by code of ethics in software engineering.

Course Content:

1. Project Management
2. Software Processes
3. Agile Software Development
4. Requirements Engineering
5. Architectural Design
6. Design & Implementation
7. Software Testing
8. Software Evolution
9. Formal Specifications
10. Introduction to Proofs of Correctness
11. Distributed Software Engineering
12. Aspect-Oriented Software Engineering
13. Process Improvement

Learning Activities and Teaching Methods:

1. Faculty Lectures and Guest-Lectures Seminars
2. Directed and Background Reading
3. Collaborative Learning through a Group Project
4. In-class Exercises
5. Student-led Presentations

Assessment Methods:

Project, Homework Assignments, On-line Quizzes, Final Exam

Required Textbooks / Readings:

Title	Author(s)	Publisher	Year	ISBN
Guide to the Software Engineering Body of Knowledge (SWEBOK(R)): Version 3.0	IEEE Computer Society, Pierre Bourque	IEEE Computer Society Press; 3rd edition	2014	0769551661

Recommended Textbooks / Readings:

Title	Author(s)	Publisher	Year	ISBN
Object-Oriented Software Engineering Using UML, Patterns and Java	Bernd Bruegge and Allen H. Dutoit	Prentice Hall, 3 rd Edition	2013	9332518688
Software Engineering, Volume 1, The Development Process, 3rd Edition	Thayer R. H. and Christensen, M. J.	Wiley-IEEE Computer Society Press	2012	978-0-471-68417-6

Software Engineering, Volume 2, The Supporting Processes	Thayer, R. H. and Dorfman, M.	Wiley-IEEE Computer Society Press	2005	978-0-471-68418-3
A series of articles and best practices approaches (electronically distributed)				