METU Department of Computer Engineering
CENG 350 Software Engineering
Spring 2017-2018

Instructors:
Dr. Pelin Angın, section 1
Dr. Ali H. Doğru, section 2
Dr. Halit Oğuztüzün, section 3

Assistants: Alper Karamanlıoğlu, Ömer Baykal

Catalogue description: Software lifecycle processes, including specification, design, construction, testing and evolution. Software process models. Modeling of computer-based systems. Software quality assurance. Software engineering standards. Professional and ethical responsibilities of software engineers.

Course learning outcomes: The students will gain an understanding of the foundations of the software engineering discipline for developing and maintaining computer-based systems. The students will be exposed to the life-cycle processes, modeling techniques, quality concepts, testing methods, standards, and ethical and professional responsibility.

Prerequisite: CENG 213
Background: Object-oriented programming.

Textbook:
Ian Sommerville, *Software Engineering*, Pearson, 2015, 10th edition.

References: UML resources, IEEE Software Engineering Standards, Software Engineering Code of Ethics and Professional Practice (SECEPP), World Intellectual Property Organization (WIPO) brochures on intellectual property rights (IPRs).

Course Outline:

Week (Hours)	*Topic [Resource]*
1 (2 h)	Introduction [1]
2 (3 h)	Lifecycle Processes [2]
3 (3 h)	Requirements Engineering [4,IEEE Std.]
4 (4 h)	System Modeling [5, UML]
5 (2 h)	Agile Methods [3], Project (SRS part-1 delivery)
6 (4 h)	Architecture Design [6,IEEE Std.], Project (SRS final delivery)
7 (3 h)	Overview, Midterm-1
8 (2 h)	Construction [7]
9 (3 h)	Testing [8]
10 (4 h)	Dependability [10,11,12,13], Project (SDD part-1 delivery)
11 (2 h)	Quality [24.1-2], Project (SDD final delivery)
12 (3 h)	Overview, Midterm-2
13 (2 h)	Evolution [9]
14 (4 h)	Professional and Ethical Issues [SECEPP], IPR [WIPO]
__ (1 h)	Guest presentation

Grading:

Midterm-1	16%
Midterm-2	18%
Project	30%
Class participation	6%
Final Exam	30%

You must get at least a total of 60 points (out of 200) from the midterm exams and submit both SRS and SDD in final form to obtain the right to sit for the final exam.

Software Engineering Project

For the semester project, you will perform the requirements engineering and architecture design of a software product (to be announced). Your SRS, compliant with ISO-IEC-IEEE 29148-2011, will specify the functional and non-functional requirements for the software, and your SDD, compliant with IEEE 1016-2009, will document your design of the architecture for the specified software.

Deliveries (both in two installments):

- Software requirements specification (SRS) (ISO-IEC-IEEE 29148-2011) [13%]
- Software design description (SDD) (IEEE 1016-2009) [17%]

Rules:

- You may work alone or with a partner.
- Delivery medium: Paper and COW (both).
- Diagramming standard is UML 2.0. Your UML diagrams should be importable by StarUML2.
- Software Engineering Code of Ethics and Professional Practice is to be observed.

Detailed Schedule

- Project (SRS part-1: SRS document including the system context and the use-case model only): March 12, 9:30
- Project (final SRS: full document): March 19, 9:30
- Midterm-1 (emphasis on process models, requirements eng.): March 29, 17:40
- Project (SDD part-1: SDD document including the composition view only): April 16, 9:30
- Project (final SDD: full document): April 27, 9:30
- Midterm-2 (emphasis on design, testing, dependability): May 3, 17:40
- Final Exam (comprehensive exam with many questions): date to be announced

Free Resources on the Web

Web page for the textbook:
http://iansommerville.com/software-engineering-book/

Software Engineering Code of Ethics and Professional Practice:
http://www.acm.org/about/se-code
https://ethics.acm.org/

IEEE Standards:
http://ieeexplore.ieee.org/browse/standards/collection/ieee

UML 2.0 in a Nutshell, by Dan Pilone and Neil Pitman
http://proquestcombo.safaribooksonline.com/0596007957?uicode=metu

UML tutorials:
http://www.sparxsystems.com/uml-tutorial.html
http://www.tutorialspoint.com/uml/
http://edn.embarcadero.com/article/31863

StarUML:
http://staruml.sourceforge.net/en/

Design Patterns:
http://hillside.net/
http://pages.cpsc.ucalgary.ca/~kremer/patterns/
https://sourcemaking.com/design_patterns

Agile Software Development:
https://www.agilealliance.org/
http://agilemanifesto.org/
http://agilemethodology.org/

WIPO (World Intellectual Property Organization) page on basics of IP:
http://www.wipo.int/about-ip/en/index.html#ip

The Risks Digest, forum on risks to the public in computers and related systems:
http://catless.ncl.ac.uk/Risks/

Variety of Topics:
http://proquestcombo.safaribooksonline.com/
http://blog.ieeesoftware.org/
https://sourcemaking.com/
https://www.eff.org/
https://www.plagiarismtoday.com/

Software Engineering Mailing List:
http://sigsoft.org/resources/seworld.html

Dilbert, by Scott Adams:
http://www.dilbert.com/

Sommerville chapters

Although the current edition of Sommerville, edition 10, is officially our textbook, if you have the previous edition it will serve almost as well for our purposes.

Here is a rough correspondence of the covered chapters and sections:

Edition 10	Edition 9
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10.1	11.1
10.2	10.0-1
11.1	11.2
11.2	12.3
12.1-2	12.2
13.1-3	12.4
24.1-2	24.1-2