Department of Computer Engineering and Computer Science

CECS 543 Course Outline

Advanced Software Engineering

DESCRIPTION

This course aims to equip students to develop techniques of software-intensive systems through successful requirements engineering, design, testing, maintenance and evolution, and project and quality management. Students build on their basic software engineering knowledge by extending it with specific techniques for maintenance, evolution, dependability, reliability, safety, security, and resilience.
Lecture 2 hours. Semester long team project plus final exam. Letter grade only (A-F).

I. PREREQUISITE TOPICS

CECS 343 or other basic knowledge about the principles of software engineering and the software lifecycle. Sufficient programming skills for the team development project.

II. COURSE TOPICS

This course exposes students to the advanced problems of software engineering. After mastering the basics of requirements engineering, design, and testing, we explore maintenance and evolution, project and quality management, as well as the engineering for distinct quality characteristics.
In detail, the course covers:

1. A recapitulation of software engineering process models
2. A recapitulation of the basic techniques for requirements engineering and design
3. Project management
4. Process and project metrics
5. Estimation for software projects
6. Project scheduling
7. Risk management
8. Maintenance and reengineering
9. Dependability of systems
10. Reliability engineering
11. Safety engineering
12. Security engineering
13. Resilience engineering

III. COURSE OBJECTIVES

- Overall: Advanced knowledge in software engineering.
- A knowledge of and an ability to apply:
    - Quality assurance techniques
    - Requirements management techniques
    - Software project planning
    - Quality engineering techniques

Sample assignments:

- Performing a review of a requirements specification.
- Developing a safety analysis for a system under development.
- Developing a project plan for a software system to be developed.
- Developing a quality assurance plan for a software project

V. METHODS FOR ASSESSING STUDENT LEARNING

- A semester-long software engineering project, composed of individual, written assignments (to practice and demonstrate the skills from the course objectives above) and the implementation of a software system.
- A final examination.

VI. FURTHER READING

- Software Engineering: A Practitioner's Approach. 7th Ed. Roger Pressman.
  Specifically chapters 24-29
- Software Engineering by Ian Sommerville. Publisher: Pearson.
  Specifically chapters 10-14

Prepared by Birgit Penzenstadler and Michael Hoffman