

CEN 5035 Software Engineering

Credits: 3 credits

Text book, title, author, and year: Ian Somerville, Software Engineering (9th Ed.), Addison-Wesley 2010

Reference materials: Course slides placed in Blackboard. Selected papers

Specific course information:

Catalog description: An introduction to the basic principles and practices of software engineering. Emphasis is placed on modeling to describe complex systems and in programming language support for software engineering principles, especially techniques for data abstraction, code reusability, and programming-in-the-large. Other topics include software life cycle models, general design, implementation and testing issues, specification and design methodologies using patterns. Emphasis on the importance of non-functional requirements

Prerequisites: Intro. to Programming, Data Structures . A senior-level course that integrates and applies the knowledge of lower-level courses

Specific goals for the course: Methods for the development of complex software systems. Lifecycle stages. Approaches to building software. Modeling for handling complexity and nonfunctional requirements. Be able to evaluate and select software lifecycle approaches and their stages

1. Analyze tradeoffs in cost, quality, and other factors in the development of software.
2. Analyze and apply different software design methods, including UML and patterns.
3. Be able to perform the basic tasks in each software development stage: define requirements, build conceptual models, apply design artifacts, code, and test
4. Be able to incorporate nonfunctional aspects into software systems, including security, reliability, and safety
5. .Apply principles of software architecture to build high quality software

Brief list of topics to be covered:

1. **Motivation and objectives**, Introduction to Software Engineering, goals and terminology, principles
2. **Software Quality** Defects in software. Verification and Validation.
3. **Software Process Models**
4. **Agile Software Processes** . Principles, agile methods
5. **Process Improvement and Process Metrics** Organization-Level Quality
- 6.. **Software Configuration Management** Goals

7. **Project Management Principles** Risk Management, Software Project Organization, Team Size
8. **Requirements** Principles of Requirements. Analysis. Nonfunctional Requirements, Security Requirements
9. **Software Design** Principles, goals. Modeling. Integrating Design Models, Frameworks