Course Information

- All information is placed on:
  Component-Based Software Systems

- Lectures:
  - Thursday 2:00-5:00pm, EIT 3141

Course Description

- Building large-scale and complex software systems from available parts with the goal of increasing return on investment, decreasing time to market, and assuring quality and reliability.

- The course covers:
  - The basic software component concepts
  - The overview of advanced topics on software components and component-based software engineering from research and practice.
Course Objectives

- There is growing interest in the notion of software development through integrating of pre-existing software components.
- This course will expose you to the **concepts, methods, techniques, processes, and tools** for engineering of component-based software systems.
- After taking the course, you will be able to:
  - Understand the issues of component-based software systems
  - Understand the current outstanding problems
  - Be familiar with the cutting edge solutions.

Course Content

- Component Concepts and Definitions
- Specification of Software Components
- Software Architecture and Components
- Component Models and Frameworks
- Developing Software Components
- Software Component Quality Models
- Component Composition and Integration
- Testing Component-Based Systems
- The Evolution and Maintenance of CBSS
- COTS-Based Development
- Cost Estimation for COTS Development

Course Grading

- Final Exam (50%)
- Course Project (35%)
- Presentation (15%)

Course Project

- The course will include a project to be done by **each student**.
- The emphasis of the project should be towards further learning of a particular topic in component-based software engineering through
  - substantial implementation
  - evaluation of a system
  - a survey
  - some combination of these things
Project Proposal

- You will prepare a one- to two-page description of your proposed project, to be submitted on June 1.

- This is simply a description of what problem you intend to address. It should be detailed enough to show that you have a reasonable and achievable project (no too trivial and not too large).

- The proposal should describe the problem, the steps you will take to work on the project, the results you expect (or hope) to obtain, what resources you will need, and a brief schedule of how you expect the work to proceed.

Project Report

- The final project should be a paper, about 5000-7000 words, double-spaced and single-sided.

- The results need not be of publishable magnitude, but the paper should be written as if it were being submitted for publication in a technical conference.

References

- Selection of papers from technical literature

- C. Szyperski, Component Software - Beyond Object-Oriented Programming, Addison-Wesley, 2003


About You

- Now it is your turn to present yourself 😊

- Name

- Main interest

- Experience with component-based development

- Expectation from the courses