

**Course Number:** CSC 640/848

**Course Title:** SOFTWARE ENGINEERING/ADVANCED SOFTWARE ENGINEERING

**Number of Credits:** 3

**Schedule:** Three hours of lecture/discussion per week.

**Prerequisite:** a grade of C or better in CSC 413; or consent of the instructor

### **Catalog Description**

Practical methods and tools for SW Engineering, including organizational teamwork. Course is paired with CSC 848. Students who have completed CSC 848 may not take CSC 640 for credit.

### **Expanded Description**

- Introduction and motivation for Software Engineering
- Overview of several basic SE methodologies with emphasis on Iterative and Incremental Development and User Centered Design
- Usability and UI design principles and practice
- Basic components of SW Engineering process: Planning; Requirements and Specifications; Iterative Design, Rapid Prototyping, Mockups; Software Design; Coding and documentation techniques (high level only);
- SW Engineering related to Web application development
- Open source SW development and management (NEW)
- Software Configuration Management, Delivery, Installation, and Documentation
- Software Metrics, Performance and Usability Measurements
- Software QA and Testing
- Software Maintenance
- Project Management issues
- Teamwork and Communication as integral part of SW Engineering
- Issues related to global SW engineering
- Basics of IP, licensing, digital rights management and copyright
- SW Engineering ethics
- Real life examples and cases from instructor and students
- Guest and student presentations
- Final Group Project including several milestones, interaction with instructor, and final demo and delivery.

### **Course Objectives and Role in Program**

Course will offer comprehensive and advanced coverage of practical methods and tools of SW Engineering, as well as its organizational, teamwork and communicational aspects. Special emphasis will be on Iterative, Incremental, Agile and User Centered Design methodologies and on global SW engineering, where teams are located in geographically and culturally dispersed areas. Students will engage in a group project in order to experience and practice key aspects of SW engineering in setting that simulates real SW company. Intended audience includes SW developers, tech leads and managers.

For more details on how we teach practical and global SW Engineering please see our research paper [http://www.cs.sfsu.edu/news/Teaching\\_practical\\_and\\_global\\_Software\\_Engineering.htm](http://www.cs.sfsu.edu/news/Teaching_practical_and_global_Software_Engineering.htm)

This course provides integrative experience in SW project organization and teamwork, including SW development in global environment, thus complementing student fundamental skills in computer science. As such, it addresses recent ACM recommendations on SW Engineering education for the areas such as SW development process, organization and teamwork.

### **Learning Outcomes**

- Knowledge of basic SE engineering methods and practices, and their appropriate application
- Knowledge of basic components and tools for full SW development lifecycle
- Knowledge of design and build practices for easy to use, maintainable SW developed using modern multi-tier architectures
- Knowledge of basic SW dependability metrics, quality metrics, and basic architectural models
- Ability to constantly iterate and re-prioritize goals based on user needs, budget, schedule and resources
- Development of significant teamwork and project based experience, as close as possible to real life
- Knowledge of basics of copyright, IP, digital rights management as it relates to SW development
- Exposure to global and open-source SE methods and practices.

### **Method of Evaluation**

Individual assignment	5% (students required to learn and install tools used in the team project)
Final Group Project	50% (Design and development of a SW application by student teams. Grade is based both on adherence to SW Eng. Process, as well as on quality and usability of final project delivery in five deliverable milestones. Team graded as a whole)
Final Exam	45% (Closed book, closed notes, individual exam)

### **Final Group Project**

*Objective:* By working on developing a SW application in a group setting, students will experience and exercise all key steps in SW Engineering methodology of choice. Grading is based on adherence to SW Engineering methodology as well as the quality of the final product.

*Organization:* Students will work in groups of 4-6. Instructor will serve as coach, VP of Engineering, Marketing and Customer representative and will attempt to create environment that simulates real life SW company. In some cases, if possible, we will simulate global SW engineering environment by including teams from other universities, currently from Fulda University in Germany. Interaction with instructor will evolve around key SW Engineering process milestones and will include e-mail feedback, group walkthrough sessions and discussion, as well as usability feedback. Instructor will not get involved nor help in details of coding and implementation; this will be the responsibility of the group.

*SW tools and environment:* Basic SW tools and environment will be provided, but students can choose to use their own. Environment for global SW engineering groups will be fixed to one set of tools. Students are expected to know basics of WWW SW Engineering.

*Deliverables:* Final documents from all the required milestones (after feedback), as well as the demo of the final product.

*Grading Policy:* Each milestone in SW Development (e.g.. high level requirements, prototype and mockups, implementation and management of iterations and usability feedback) and final delivered product will carry equal weight, for the total of 50% of the final grade. This way we will ensure adequate focus on SW Engineering process. Grading will be based only on final document versions; instructor's feedback during the process will not be graded but serve as help and guidance. The team members will receive the same grade unless there are violation of teamwork practices, in which case grades may be modified.

### **Class attendance and student responsibilities:**

Students **must**:

- Attend first four weeks to be assigned to the group and establish teamwork – if not assigned to groups in the first two weeks student will be dropped. After this attendance is strongly encouraged.
- Focus early on Milestone 0
- Participate in team activities and follow good team ethics and behaviour
- Read and respond to team and inductor e-mails and follow the requested deadlines and deliverables
- Fill in timecards and surveys used in class project management (not to be used for grading)

### **Suggested Main Textbook (class slides will be sufficient also)**

- Ian Somerville: "Software Engineering", 6th or later edition, Addison Wesley, Book's Web site <http://www.software-engin.com/>

### **Other References (optional textbooks and reading)**

- D. McCracken, R. Wolfe: "User Centered Website Development: A Human Computer Interaction Approach ", PPH, 2004
- R. J. Torres: "Practitioner's Handbook for User Interface Design and Development", Prentice Hall PTR, 2002
- W. O. Galitz: "The Essential Guide to User Interface Design", Second Edition, John Wiley & Sons, 2002
- J. Rubin: "Handbook of Usability Testing", John Wiley & Sons, 1994
- E. Dustin et al: "Quality Web Systems", Addison Wesley, 2002
- F. P. Ginac: "Creating High Performance SW Development Teams", Prentice Hall, 2000
- M. Bays: "Software Release Methodology", Prentice Hall PTR, 1999
- K. Beck: "Planning Extreme Programming", Addison Wesley, 2000
- C. Larman, V. Basili: "Iterative and Incremental Development: A Brief History", IEEE Computer, June 2003 (paper)

**Notes:**

This is paired course. One benefit of this is that we will mix graduate and undergraduate students in the programming teams.

SFSU graduate students who took CSC 640 at SFSU in 2003 or after and who need CSC 848 credit, please contact the instructor.

**Modified by:** D. Petkovic

**Last Revision Approved:** May 2010