

CSC 509 Software Engineering II Spring 2009

Instructor David Janzen
Website <http://www.csc.calpoly.edu/~djanzen/courses/509S09/>
Email djanzen@calpoly.edu
Office 14-212
Phone (805)756-2929
Office Hours Monday: 3:10-5pm; Tuesday: 11:10am-noon; Wednesday 2:10-4:30pm;
Friday: 2:10-2:30pm

Catalog Description

In-depth study of software modeling and design. Formal design methodologies. Design patterns. Detailed case studies of existing projects. Tools and methods for designing large software systems. 4 seminars.

Prerequisites

CSC 508 and graduate standing, or consent of instructor.

Required Texts

There are no required texts for this course. Many scholarly papers will be assigned for reading and discussion. Most papers will be available through the internet and digital libraries.

Schedule

This course will meet Monday and Friday from 12:10 to 2pm in 14-232B.

Course Requirements

1. Attendance and Participation

Students are expected to attend and actively participate in all course meetings and discussions, as well as the course project. Readings will be assigned and must be completed prior to the class in which they are discussed. Readings will be posted on the course website schedule. Occasional quizzes may be given which will count towards the attendance/participation grade. More than three absences for any reason will result in a zero for the attendance/participation grade.

2. Technical Paper Presentations

Each week we will select papers to be discussed in class from software engineering conferences and publications such as IEEE Software, IEEE Transactions on Software Engineering, the International Conference on Software Engineering (ICSE), the IEEE Conference on Requirements Engineering, and the conference on Object Oriented Programming, Systems, Languages, and Applications (OOPSLA). Although some papers will be selected by the instructor, each student will also select a paper from an assigned issue/conference. These selections must be announced to the class one week prior to the date that the paper is to be discussed. All students are to read the selected papers, and the student who selected each one will lead a discussion on the paper. Students will be graded on their summary of the salient points of the paper, and effectiveness leading the class in a meaningful discussion of the paper.

3. *Current Events/Video Reviews*

Each student is expected to present two current events or video reviews in class. These can be presented on any day when we are discussing technical papers (not on days with mini-tutorials or guest speakers). Students should turn in a brief printed summary of the review at the time it is presented.

4. *Mini-Tutorial*

This course will focus on software design and methods/tools/technologies for developing large systems. Individually or in pairs, students will prepare a two-hour mini-tutorial on a particular technology, framework or tool. Example technologies include: Spring, Struts, Flex, Adobe Data Services, J2EE and EJB, .NET, Software Product Lines tools, xUnit, Aspect-Oriented Programming, Service-Oriented Architectures, Model Driven Architecture tools, Alloy, Architecture Description Languages, Object Constraint Language, iPhone, Android, etc. Particular emphasis should be placed on how the technologies exploit design patterns and architectural styles. The tutorials should have a practical hands-on component for all participants. The tutorial will be graded on the presentation quality, the student's mastery of the subject, accurate characterization of design patterns and architectural styles, and the ability to effectively transfer knowledge to the class.

5. *Research Paper*

Students will write a high-quality, publishable research paper. The topic should be consistent with the topics of this course, but may incorporate the student's other interests or research. The paper should target a particular software engineering conference, journal, or scholarly magazine, and should match the respective formatting guidelines. Failure to identify the target venue and match the formatting guidelines will result in a 15% grade reduction on the paper. You are not required to submit the paper for publication, but you are encouraged to do so. The goal is that your paper is of publishable quality. Keep track of all contributors so you can properly acknowledge them if you do submit for publication. At a minimum, relatively short papers (around 5,000 words) could target high-quality magazines such as IEEE Software, IEEE Computer, or Communications of the ACM (around 2,500 words). Medium-length papers could target high-quality conferences such as OOPSLA (max 10,000 words) and ICSE (10 pages in ICSE format). Longer papers could target high-quality journals such as "Transactions on Software Engineering" or "Empirical Software Engineering, An International Journal." Such papers may be spread over the 508 and 509 courses, but a distinct deliverable must be provided in this course.

The following deadlines pertain to the research paper:

4/13/09: Initial Paper Proposal and Presentation

4/24/09: Structured Abstract & Bibliography

5/11/09: Complete First Draft

6/1/09: Final Paper

Failure to meet an intermediate deadline will result in a 10% reduction in the final paper grade.

6. Review Peer Work

Students will review each others' papers at three points: verbal feedback will be given collectively at the initial paper proposal presentations, written feedback of the first complete draft will be given for two peers, and written feedback of the final draft will be given for two peers. The last stage of this process is meant to imitate a scholarly peer-review process.

Communication

The best place to discuss the course is during class meetings and office hours. All assignments and due dates will be announced in lecture and most, if not all, will be placed on the course web site. Most class materials are available on the course web site; be sure to check regularly.

Email will only be used for special circumstances, such as communicating time sensitive information. You may email the instructor regarding personal issues only. If you use email, put CSC 509 on the subject line to get the best response time. All students are expected to have their calpoly.edu and csc.calpoly.edu email accounts forward to wherever they will read email at least daily. Leaving phone voicemails should be a last resort.

Classroom Etiquette

To ensure a professional learning environment, the following rules will be enforced in the classroom:

- Do not eat except when food is provided for the entire class
- Do not use electronic devices that make sounds (e.g. cell phones, ipods)
- Do not use computers for anything besides presenting or taking notes when anyone is presenting (exceptions will be clearly announced, e.g. during project meeting times)

Grading

The course grade will be determined on the following factors:

1. Class Attendance and Participation (10%)
2. Technical Paper Presentations (10%)
3. Current Events/Video Reviews (5%)
4. Research Paper (35%)
5. Mini-Tutorial (35%)
6. Peer Paper Reviews (Draft, Final) (2.5% each)

Letter grades will be assigned based on the expectation that an 'A' is earned with excellent work on all aspects of the course, sustained throughout the course. A 'B' is earned with very good work, perhaps excellent at times. A 'C' is earned with average work, perhaps very good at times but poor at other times. A 'D' is earned with consistently poor work.

Classroom Participation

Students are expected to take an active role in their own learning and the learning of their peers. Students are expected to attend and be prepared for all lecture and lab sessions including reading all assigned sections prior to class. Lecture times will reflect this.

Late Work Policies

A software engineer has a responsibility to manage time effectively and turn in work on time. Deadlines are rarely absolute; if you are having a problem, discuss it as far in advance as possible. Failure to do so will result in a grade reduction.

Integrity

All work submitted is to be your own. Cooperative study and mutual aid are healthy learning methods and are strongly encouraged. You are especially encouraged to build on the work of others. Just cite sources of anything you have copied, summarized or discussed directly with another. It is cheating to copy someone's work or allow someone to copy your work. It is cheating to copy material from a publication without giving credit. Plagiarism will result in a course grade of F. When you find good ideas by other people, the best policy is to summarize other work in your own words and cite their work as the source for the principle you state. Citing resources is not a sign of weakness of your own ideas, it is a sign that you can do research and build on others' work.