

CS 506 Software Engineering
FALL 2015
The real syllabus: <http://go.wisc.edu/91st27>

Instructor: Peter Ohmann
Office: Computer Sciences 6352
Office Hours (Tentative): Tues/Thurs 10:00 – 11:00 AM
Email: ohmann@cs.wisc.edu

TA: Honghui (Alan) Zhang
Office: TBD
Office Hours (Tentative): Mon 4:00 – 5:00 PM
Email: honghui@cs.wisc.edu

Lecture: Mon/Wed 2:30–3:45PM, Noland Zoology 168
Moodle Course Page: <http://go.wisc.edu/91st27>

Course Description

Building large software systems is hard, but experience shows that building large software systems that actually work is even harder. Trying to do all this before your competitors has proved fatal to many software projects. This course covers techniques for dealing with the complexity of software systems. We will focus on the technology of software engineering for the individual and small team, rather than business or management issues. Topics will include, among others: specifications, principles of design and software architecture, testing, debugging, static analysis, and version control. You are expected to actively participate in the classes and the online discussion forum.

This course is distinctive in the CS curriculum in that students are involved in a large-team project. You will see what it takes to collaborate with people with different skills and approach to software development. Students select the topics of the projects and almost all aspects of development (programming language, libraries, build environment, etc.).

Prerequisites: CS 367 and at least one other project-oriented course (e.g., CS 536, 537, 545, 559, 564, 570)
Textbook: None. All readings will be posted on Moodle.

Course Objectives

Students who successfully complete this course will gain a broad understanding of software engineering concepts and tools, as well as their application to the design and development of software systems. This includes, but is not limited to: basic software engineering models (e.g., waterfall, agile), software requirements and design modeling (e.g., UML), source configuration management (e.g., git, svn), and testing frameworks and methodologies (e.g., TDD). These students will gain practical hands-on experience designing and building a real software system in a large (6–8 person) team.

Grade Distribution

3% constructive participation in class and on the class discussion forum
10% (≈ 2) homework assignments
25% midterm exam
62% project
There will be no final exam.

Tentative Course Outline

Week 1	08/31 – 09/04	Introduction, What is Software Engineering?
Week 2	09/07 – 09/11	Software Engineering Processes
Week 3	09/14 – 09/18	Agile Processes
Week 4	09/21 – 09/25	Requirements and Specification, UML
Week 5	09/28 – 10/02	Software Architectures
Week 6	10/05 – 10/09	Design Patterns, Tools
Week 7	10/12 – 10/16	Project Presentations (10/14), Source Configuration Management
Week 8	10/19 – 10/23	Unit Testing
Week 9	10/26 – 10/30	Test-Driven Development
Week 10	11/02 – 11/06	Advanced Testing Topics, Code Coverage
Week 11	11/09 – 11/13	Midterm (11/09 in class), Debugging
Week 12	11/16 – 11/20	Research in Software Engineering
Week 13	11/23 – 11/27	Presentation of Project Outcomes (11/23 and 11/25)
Week 14	11/30 – 12/04	Guest Speakers
Week 15	12/07 – 12/11	Final Project Demos (12/07 or 12/09)
Week 16	12/14 – 12/18	Secrets of Software Engineering Grad Student Life

Conflicts

If you have a conflict with any deadlines (see the full list on the Moodle page), you *must* notify me within the first two weeks of class. Notify me ASAP if you have a conflict with a presentation date; you are not required to present with your team, but it will impact your presentation attendance grade. Note that travel for holidays is *not* a valid excuse. For the midterm, if you wait until after this deadline, it is very unlikely that I will grant you a make-up exam.

Academic Misconduct

Plagiarism is a very serious offense, and cheating on assignments or exams will not be tolerated. All cases will be dealt with through the University's Office of Student Assistance and Judicial Affairs. You may collaborate with your team on the project, but must cite outside sources. You should work alone on homework assignments. If you receive *any* assistance on a homework assignment from anyone other than course staff, you *must* cite them in the assignment write-up; your grade will be based on *your* contribution to the assignment. No help will or may be given on the midterm exam. Failure to acknowledge sources (in homeworks or the project) is plagiarism and will be treated as a serious breach of academic honesty.

General Policies

- Regular attendance in lectures is expected, and you will be partially graded on your participation both in-class and on Moodle. *Resistance is futile.*
- I don't care what you bring to class, so long as it isn't a distraction. If your cell phone rings during class, that is a distraction. *Set phasers to stun* (and phones to silent).
- Use Moodle for most class-related discussion. Moodle participation partially determines your grade. *It would be highly illogical* to wait for my email response if a classmate could answer your question.
- That said, don't use Moodle for questions specific to you (personal absences, etc.) or specific homework questions. Email all members of the course staff with homework questions.
- This course should be loads of fun. If it isn't fun, either you or I must be doing something wrong. If you think it's me, send me email! I'm always open to feedback about how the course is progressing.
- Go to the real syllabus / course page: <http://go.wisc.edu/91st27> *Make it so.*