

Course Syllabus

Read this carefully!

28 August 2008

Meeting Time and Place: Tue & Thu 8:00–9:15 in RGAN 203

Instructor: Professor Ken Calvert

Instructor Email: calvert@netlab.uky.edu

Course Web Page: <http://protocols.netlab.uky.edu/~calvert/classes/499-f08/>

Instructor Office Hours: Tuesday and Thursday 9:15-10:30 in FPAT 773C. Wednesday 8:30-10 in Hardymon 230.

Required Textbooks:

1. *Design Patterns: Elements of Reusable Object-Oriented Software*, by E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Addison-Wesley Professional Computing Series, 1995. ISBN 0-201-63361-2.
2. *Making Teams Work: 24 Lessons for Working Together Successfully*, by Michael Maginn. McGraw-Hill Professional Education Series, 2003. ISBN 0-071-43530-1.

Both of these will be used for assignments.

Recommended Textbooks:

- *The Practice of Programming*, by Brian Kernighan and Rob Pike. Addison-Wesley Professional Computing Series, 1999. ISBN 0-201-61586-X. I really like this book, and recommend it for anybody who does any programming.
- *The Mythical Man-Month* (second edition), by Frederick P. Brooks. Addison-Wesley Professional, 1995. ISBN 0-201-83595-9. Unquestionably a classic. Though some of the specifics are now astonishingly dated (original copyright 1975—check out the reference to “ferrite” as a component of hardware in the first chapter), the principles given here are timeless.

1 Course Description

This course provides a “capstone” experience for Computer Science and Computer Engineering students. Students will work together in small teams to specify, design, and implement a nontrivial software system for a “customer”. To accomplish this, they will need to learn about software project management approaches, design, teamwork and management skills, and professional and ethical responsibilities.

2 Expectations (Preconditions)

Students are assumed to be Computer Science majors in their senior year. A knowledge of program development techniques in an object-oriented language and a knowledge of data structures and algorithms at the level of CS 315 is assumed.

3 Learning Outcomes (Postconditions)

Students will gain experience in the design and implementation process using material from throughout their undergraduate career. They will gain experience working in groups. At the conclusion of the course, the successful students will be able to:

1. use accepted project development processes in the project implementation;
2. implement a large project;
3. work as part of a team;
4. present their work results orally;
5. document their work in written form.

In addition, students will be able to discuss the ethical responsibilities of a computing professional and be aware of the contents of the IEEE code of ethics for such professionals.

4 Policies

This section describes the policies that will be followed in the course. Please read it carefully. If you have questions, ask them **now** rather than later.

4.1 Cell Phones

Unless your spouse is pregnant and delivery is imminent, there's no reason for you to be answering a phone call or page during class. Please remember to turn off or silence your phone/pager. (So will I.)

4.2 Responsibility and Attendance

One of the purposes of this course is to give you exposure to the kinds of **responsibility** you will experience in a job after graduation. One of your responsibilities is to **show up**. A portion of your grade will be determined by your participation in class and in your team's work (as evaluated by your teammates). Just to be completely explicit: if you regularly (or even occasionally) miss class or are chronically late, your grade will be affected (negatively).

4.3 Teams

Much of the work you do for this course will be done in the context of a team. I will assign you to teams. (In the real world, teams generally are not self-organizing.) I will endeavor to balance teams' composition in various dimensions. I will also appoint one member of each team to be the leader; you can change the leadership of your team if you like and if everybody agrees. In case of a dispute about leadership on your team, I will arbitrate and my decision will be final.

All team members will receive the same grade on the project components. You will be asked periodically to assess your teammates' contribution to the project.

4.4 Grading

Your grade in the course will be determined by your performance with respect to the learning outcomes of the course, as measured by:

- Your performance on homework assignments and the occasional quiz (graded by me);
- Your team's grade on the project deliverables and final presentation (determined by me, with input from the project's stakeholder);
- Your contribution to your team's results (as assessed by your teammates); and
- Your assessment of your teammates' contributions to your team's results.

The high-level breakdown of the contribution of these components is as follows:

Assignments	25%
Team Project	50%
Your team contribution	10%
Your assessment of teammates' contributions	15%

Your team's project grade will be made up of several sub-components, including: weekly team progress reports; a project requirements/specification document; a design document; a test plan; and the final project presentation. Again, **all team members will receive the same grade on each team deliverable**. In addition, your team will present a prototype demonstration partway through the semester.

4.5 Academic Conduct

I expect each student to act honestly and to do his or her own work. It is my responsibility and my intention to protect the interests of the honest students. Therefore **CHEATING IN ANY FORM WILL NOT BE TOLERATED**. Now, I don't mind if you help each other with understanding the material; in fact, I encourage it. The key point is this:

Anything that you turn in—homework, examinations, whatever—with your name on it **must be your own work**, composed and written by you without looking at others' work.

If you need to refer to someone else's work, or if you collaborate with someone, **cite them explicitly**. **Plagiarism, including misrepresenting others' work as your own, will not be tolerated in any form.**

Academic offenses (including plagiarism, cheating, and falsification or misuse of academic records) are defined in Section 6.3.0 of Part II of the University Senate Rules. You are expected to have read and abide by those rules. In particular, note well the following:

6.3.1 PLAGIARISM. All academic work, written or otherwise, submitted by students to their instructors or other academic supervisors, is expected to be the result of their own thought, research, or self-expression. In cases where students feel unsure about a question of plagiarism involving their work, they are obliged to consult their instructors on the matter before submission.

When students submit work purporting to be their own, but which in any way borrows ideas, organization, wording or anything else from another source without appropriate acknowledgment of the fact, the students are guilty of plagiarism.

Plagiarism includes reproducing someone else's work, whether it be published article, chapter of a book, a paper from a friend or some file, or whatever. Plagiarism also includes the practice of employing or

allowing another person to alter or revise the work which a student submits as his/her own, whoever that other person may be. Students may discuss assignments among themselves or with an instructor or tutor, but when the actual work is done, it must be done by the student, and the student alone.

When a student's assignment involves research in outside sources or information, the student must carefully acknowledge exactly what, where and how he/she has employed them. If the words of someone else are used, the student must put quotation marks around the passage in question and add an appropriate indication of its origin. Making simple changes while leaving the organization, content and phraseology intact is plagiaristic. However, nothing in these Rules shall apply to those ideas which are so generally and freely circulated as to be a part of the public domain.

When academic dishonesty is suspected, the procedures specified in Section 6.4.0 of the University Senate Rules will be followed.

5 Schedule

The following schedule is a draft and is subject to change.

Week	Date	Topic	Milestone/Deliverable
0	28 Aug	Course/syllabus intro	diagnostic survey.
1	2 Sep	Software project management	Project overviews.
2	9 Sep	Software development processes	Team assignments; Refine project list
3	16 Sep	Requirements and specifications.	Begin weekly team reports . Project selections
4	23 Sep	Design patterns	Project contracts signed.
5	30 Sep	Design patterns	Meet with stakeholders; finalize requirements
6	7 Oct	Software validation/testing	Exam . Design and test plan
7	14 Oct	Formal Methods	
8	21 Oct		Project meetings
9	28 Oct	Ethics and professional responsibility	prototype demos
10	4 Nov		prototype demos
11	11 Nov		Team reports
12	18 Nov		Team reports
13	25 Nov		Team reports
14	2 Dec		Final presentations
15	9 Dec		Final presentations