CSE 442  Software Engineering  Fall 2018

University at Buffalo
Department of Computer Science & Engineering
338 Davis Hall – (716) 645-4736

Syllabus

Please read this sheet carefully, and save it for future reference.

Instructor

<table>
<thead>
<tr>
<th>Name</th>
<th>Office</th>
<th>Phone</th>
<th>Email</th>
<th>Office hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matthew Hertz, Ph.D.</td>
<td>352 Davis</td>
<td>645-4736</td>
<td><a href="mailto:mhertz@buffalo.edu">mhertz@buffalo.edu</a></td>
<td>Tuesday 2:00 PM – 3:30 PM OR by appointment</td>
</tr>
</tbody>
</table>

Teaching Assistants
See course website.

Course Information

Credit hours:
CSE 442 Software Engineering – 4 credits

Course Website: [www.cse.buffalo.edu/~mhertz/courses/cse4426](http://www.cse.buffalo.edu/~mhertz/courses/cse4426)

Course Description

This course examines the software development process in detail. Using a semester-long team-project of the students’ choosing, the class provides a first-hand experience with agile software life-cycle models; architectural and design approaches; systematic software testing; coding and documentation strategies; project management; working in teams; customer relations; the social, ethical, and legal aspects of computing; and the impact of economic, environmental, safety, manufacturability, and sustainability factors on design. Students are expected to stay up-to-date with both the course and the project and do their best to put these topics into practice. While many mistakes are expected, the intent is that students appreciate the importance of their software engineering choices and can avoid duplicating these mistakes later in the term and (more important) in their future careers.

The course website contains a detailed, day-by-day schedule of topics to be covered.

Prerequisites

You must have passed CSE250 and have a declared Computer Science, Computer Engineering, or Bioinformatics major.

Given the intensity of work in this course, no exceptions to these requirements can occur.

Textbook and Materials

There is no required textbook for this course nor any required materials. There will required readings throughout the term, but these readings will be a mix of articles by practicing software engineers and descriptions of how to carry these ideas through. Readings will be available as links on the course website and must be completed for the lecture in which they are listed.

Additional reading material may be assigned during the course, and will be announced in lecture.
Student Learning Outcomes
This course serves as a capstone course for the computer science major. It covers all 6 student learning outcomes from the ABET Computer accreditation standards. A mapping of the student learning outcomes and instruments used to assess these outcomes are:

<table>
<thead>
<tr>
<th>Upon successful completion of this course a student will be able to...</th>
<th>Assessment Instruments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.</td>
<td>In-Class Activities, Final Presentation</td>
</tr>
<tr>
<td>2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program’s discipline.</td>
<td>In-Class Activities, Course Project</td>
</tr>
<tr>
<td>3. Communicate effectively in a variety of professional contexts.</td>
<td>Course Project, Final Presentation</td>
</tr>
<tr>
<td>4. Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.</td>
<td>In-Class Activities</td>
</tr>
<tr>
<td>5. Function effectively as a member or leader of a team engaged in activities appropriate to the program’s discipline.</td>
<td>Course Project (Self- and Peer-Assessments)</td>
</tr>
<tr>
<td>6. Apply computer science theory and software development fundamentals to produce computing-based solutions.</td>
<td>In-Class Activities, Course Project</td>
</tr>
</tbody>
</table>

Computing Resources
You will be provided with a CSE undergraduate computing account. You may use the undergraduate lab facilities in Baldy 19 and Baldy 21. These facilities are available for use as listed on the course website. They are on card-access - use your UB card to open the door. For your own safety, and to protect the equipment in the lab, do not open or hold the door open in order to allow other people to gain entry to the lab. All students authorized to use the lab have card access.

As this is an upper-level course, you are expected to be proficient at using the machines in the lab and the Linux operating system. While each team may design a project of their choice, your entire team will need to do whatever learning is required to gain expertise in that environment. A lack of familiarity with your chosen tools is NOT an acceptable excuse for a lack of work. Information about the CSE computing environment can be found at [https://wiki.cse.buffalo.edu/services/](https://wiki.cse.buffalo.edu/services/)

You are expected to use your UB e-mail account for all communications with course staff. Always include your full and an informative subject line for your e-mail.
Course Requirements

Lectures
For all but a few lectures, students will have assigned readings. These readings are listed on the course webpage. Lectures will expand and reinforce material from those readings with an assumption students have completed it. Lectures will present practical applications of these concepts with the expectation that students incorporate this into their project workflow. Additionally, the class will include occasional activities to provide a chance to reflect and review these concepts. It is important that students remain up-to-date with this material and seek additional help whenever they struggle.

Class attendance is mandatory; this policy discusses enforcement. **If you miss a class, you are responsible for talking to your classmates, TAs, or the instructor to find out what happened.** If you must miss class for an extended period of time, notify your instructor as soon as possible, and see your instructor immediately upon your return in order to determine how to catch up. If you have missed a significant portion of the semester, it may be recommended that you resign from the course.

TA Meetings
Each project team will need to sign up for a weekly meeting time with one of the course TAs. The meeting is a substitute for the daily standup normally used in industry. As with the daily standup, **it is critical that all members attend every TA meeting.** At these meetings, the TA serves as a combined ScrumMaster/product owner and is responsible for keeping the team cohesive and focused. Details about possible meeting times and how group’s will sign up for a meeting time will be discussed in class during the first two weeks of the term.

For most students, this class will be their first experience working using proper software engineering ideas and techniques. The TA meetings will also provide teams with a chance to get questions they have about the best ways of applying course concepts to their project. These interactions can also be used to help clarify questions about the grading criteria and get an outside opinion as to whether their documents can be clearly understood. For these reasons, the meetings also serve as an important opportunity for students to gain a deeper practical understanding of software engineering.

Time outside of class
Office hours are another chance for you to get individualized answers to your questions about the course. Both the instructor and the teaching assistants have scheduled office hours posted on the course website. Office hours are held on a first-come first-served drop-in basis. **No appointment is necessary,** but be aware that office hours become increasingly busy as deadlines or exams approach. Plan your use of office hours accordingly. Meetings outside of planned office hours are possible, but if you want to meet outside of planned office hours, you will need to talk to the instructor or teaching assistant and see if their schedule allows.

New York State Board of Regents regulations specify for every hour of time spent in lecture or recitation, students are expected to spend 2–3 hours on work outside of class. This means that **you should expect to spend at least 8 hours each week on readings, meetings, working on your project, and studying.**
Grading Policy
Each student’s grade is computed from a weighted average of the following items:

<table>
<thead>
<tr>
<th>Item</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Signup &amp; Writeup</td>
<td>2%</td>
</tr>
<tr>
<td>Sprint 1 Grade</td>
<td>18%</td>
</tr>
<tr>
<td>Sprint 2 Grade</td>
<td>25%</td>
</tr>
<tr>
<td>Sprint 3 Grade</td>
<td>30%</td>
</tr>
<tr>
<td>Final Presentation</td>
<td>15%</td>
</tr>
<tr>
<td>Activities</td>
<td>10%</td>
</tr>
</tbody>
</table>

If necessary, the instructor may revise how the final grade will be calculated. In this situation, changes will be announced during lecture and an announcement will be sent via e-mail to each student’s UB e-mail account. Each item within the course grade is described below.

The final letter grade is based upon the following cutoffs:

<table>
<thead>
<tr>
<th>Percentage</th>
<th>Letter Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>93+</td>
<td>A</td>
</tr>
<tr>
<td>90-92</td>
<td>A-</td>
</tr>
<tr>
<td>87-89</td>
<td>B+</td>
</tr>
<tr>
<td>83-86</td>
<td>B</td>
</tr>
<tr>
<td>80-82</td>
<td>B-</td>
</tr>
<tr>
<td>77-79</td>
<td>C+</td>
</tr>
<tr>
<td>73-76</td>
<td>C</td>
</tr>
<tr>
<td>70-72</td>
<td>C-</td>
</tr>
<tr>
<td>67-69</td>
<td>D+</td>
</tr>
<tr>
<td>60-66</td>
<td>D</td>
</tr>
<tr>
<td>0-59</td>
<td>F</td>
</tr>
</tbody>
</table>

If necessary, the instructor may revise these cutoffs downward.

Sprint 1 (20% of total course grade), Sprint 2 (25% of total course grade), & Sprint 3 (30% of total course grade)
Each team will complete their project over a series of 3 sprints. The deadlines for each of these sprints is:
- Sprint 1 – Sept. 23rd
- Sprint 2 – Oct. 22nd
- Sprint 3 – Dec. 3rd

A detailed rubric used to grade the student work is available on the course website. The grading for each sprint will be accomplished by having the team give an end-of-sprint presentation (including a demo of their current release) to 2 members of the course staff. During this presentation, it is the team’s responsibility to demonstrate that they met each of the criteria from the grading rubric. They will also need to answer questions the grading staff have.

Multiplying the number of team members by the team’s score on the presentation calculates the total number of individual points available for that sprint. Students will also complete a peer- and self-evaluation on how well each person contribute to the team during the sprint. Each student’s score is calculated by multiplying the total number of team points available and their normalized average percentage of points on those evaluations. This should encourage excellence in project software engineering, but also excellence in teamwork.

The TA with whom the group meets each week will NOT have any input into the team’s sprint grade. This ensures open and honest communication during those meetings AND reflects the separation of development and testing found in larger organizations.

Details about how to sign up for these end-of-sprint presentations will occur closer to each of the deadlines.

Final Presentation (15% of total course grade)
Students will present their project in a style and with content that is appropriate for a commercial trade show. These presentations will occur in the final two weeks of the course and during the final exam period. All students are required to attend all of these presentations. A detailed grading rubric for these presentations will be available on the course website several weeks before the presentations begin. Teams will also be able to sign up for the time they wish to present at that time. These sign-ups will be done on a first-come, first-served basis.
In-Class Activities (10% of final course grade)
Throughout the term, groups will work on an in-class activity related to the course material. These activities will require that students reflect on software engineering concepts before writing a few paragraphs on the way they think is best to bring those concepts into practice. The documents students write will be submitted via AutoLab and scored holistically. Since these activities will also be used to assess department-level student learning outcomes, it is important that all students participate.

Student submissions will need to list the names of all of the group members who participated in creating the response. Listing a student who did not attend or attended but did not contribute is an academic integrity policy violation by both the student whose account is being used and the student answering the questions.

Incomplete (I) grades
We will follow the UB Undergraduate Catalog Statement on Incomplete Grades, found in the Undergraduate Catalog. Generally, incomplete (“I”) grades are not given. However, very rarely, circumstances truly beyond a student’s control prevents him or her from completing work in the course. In such cases the instructor can give a grade of “I”. The student will be given instructions and a deadline for completing the work, usually no more than 30 days past the end of the semester. University and department policy dictate that “I” grades can be given only if the following conditions are met:
- An Incomplete will only be given for missing a small part of the course.
- An Incomplete will only be given when the student misses work due to circumstances beyond his/her control.
- An Incomplete is to be made up with the original course instructor within the time specified by the appropriate University regulation (see appropriate document above), and usually within the following semester.
- An Incomplete will not be given to allow the student to informally retake the entire course, and have that grade count as the grade of the original course.

Incomplete(s) cannot be given as a shelter from poor grades. It is your responsibility to make a timely resignation from the course if you are doing poorly for any reason. The last day to resign the course is Friday, November 11, 2016.

Accessibility Resources
25 Capen Hall, Tel: 645-2608, TTY: 645-2616, Fax: 645-3116 www.buffalo.edu/accessibility
If the Accessibility Resources office has determined that you are eligible for class accommodations, such as recruiting notetakers, readers, or extended time on exams or assignments, you must provide the course instructor with written documentation before any accommodation can be provided.

Counseling Center
120 Richmond Quad, Tel: 645-2720, Fax: 645-2175 ub-counseling.buffalo.edu
The Counseling Center staff is trained to help you deal with a wide range of issues, including how to study effectively and how to deal with exam-related stress. Services are free and confidential.

Distractions in the Classroom - Behavioral Expectations – UB Policy
Classroom “etiquette” expectations include:
- Attending classes and paying attention. Do not ask the instructor in class to go over material you missed by skipping a class or not concentrating;
- If you must enter a class late, do so quietly and do not disrupt the class by walking between students and the instructor. Do not leave class unless it is an absolute necessity and then leave by the least disruptive means possible;
- Not talking with other classmates while the instructor or another student is speaking;
- If you have a question or a comment, please raise your hand, rather than starting a conversation about it with your neighbor;
- Turn off your cell phones, pagers, and other noise making devices. If you need to keep the device on, place it in silent/vibrate mode so it will not disturb the class;
- Avoid audible and visible signs of restlessness. These are both rude and disruptive to the rest of the class;
- Focus on class material during class time. Sleeping, talking to others, doing work for another class, reading the newspaper, checking email, and exploring the internet are rude and unacceptable;
- Not packing bookbags or backpacks until the instructor has dismissed class.
Academic Integrity

Source: [http://www.cse.buffalo.edu/undergrad/policy_academic.php](http://www.cse.buffalo.edu/undergrad/policy_academic.php)

The academic degrees and the research findings produced by our Department are worth no more than the integrity of the process by which they are gained. If we do not maintain reliably high standards of ethics and integrity in our work and our relationships, we have nothing of value to offer one another or to offer the larger community outside this Department, whether potential employers or fellow scholars.

For this reason, the principles of Academic Integrity have priority over every other consideration in every aspect of our departmental life, and we will defend these principles vigorously. It is essential that every student be fully aware of these principles, what the procedures are by which possible violations are investigated and adjudicated, and what the punishments for these violations are. Wherever they are suspected, potential violations will be investigated and determinations of fact sought. In short, breaches of Academic Integrity will not be tolerated.

**Departmental Statement on Academic Integrity in Coding Assignments and Projects**

All academic work must be your own. Plagiarism, defined as copying or receiving materials from a source or sources and submitting this material as one's own without acknowledging the particular debts to the source (quotations, paraphrases, basic ideas), or otherwise representing the work of another as one's own, is never allowed. Collaboration, usually evidenced by unjustifiable similarity, is never permitted in individual assignments. Any submitted academic work may be subject to screening by software programs designed to detect evidence of plagiarism or collaboration.

It is your responsibility to maintain the security of your computer accounts and your written work. Do not share passwords with anyone, nor write your password down where it may be seen by others. Do not change permissions to allow others to read your course directories and files. Do not walk away from a workstation without logging out. These are your responsibilities. In groups that collaborate inappropriately, it may be impossible to determine who has offered work to others in the group, who has received work, and who may have inadvertently made their work available to the others by failure to maintain adequate personal security. In such cases, all will be held equally liable.

**Departmental Policy on Violations of Academic Integrity**

The CSE Department has a zero-tolerance policy for AI violations.

**All** AI violations will be reported to the department, school, and university, and recorded.

Even a 1st offense will receive an "F" for the course unless the instructor finds there are mitigating factors that make it appropriate to reduce the penalty. Under departmental policy, subsequent AI violations must result in an "F" grade, with no exceptions for the form or course in which the earlier violation occurred.

**Course Policy on Violations of Academic Integrity**

Aside from recitation activities and the course project, work you submit for credit in this class is individual work.

Between instructor office hours, UTA office hours, Piazza, e-mail, and reviewing the textbook and slides, students have sufficient resources to have their questions answered. There is no reason why students should violate AI policies. While it a reduced sanction from the and so it is extremely unlikely that the instructor will deem it appropriate to reduce the sanction resulting from an AI policy violation.