



Department of Computer Science and Software Engineering

Senior Project 1

CSSE 497 Fall Term 2019

Instructor(s): Joseph Hollingsworth

Email: hollings@rose-hulman.edu

Phone(s): 812-877-8482 (office)

Office: Moench Hall, Room F203

Office Hours: Check with Senior Project Advisor at first meeting.

Or by Appointment.

Introduction

Software projects are challenging efforts and as CSSE students prepare to enter the workforce, experience is needed. In the junior year sequence of courses (CSSE 371/372/374/375) students gained experience with key aspects of software engineering, often engaged in term-long projects which are largely directed by the respective instructors. Topics covered included:

- | | |
|---|--|
| 1. Software requirements analysis & specification | 4. Software construction & evolution |
| 2. Software project planning & management | 5. Forming and managing project teams |
| 3. Software architecture & design | 6. Software risk planning and management |
| | 7. Software configuration management |
| | 8. Software validation & verification |

Senior capstone projects cover these topics and more, but the students play a much more directive role in their projects by working with the clients to define the projects (typically 1400-1600 hours of effort for 4 team members). Rather than meeting in class, there are weekly meetings with the faculty advisors and the clients. The focus for the students is largely on the full development of a software system for a real-world client where the students define, plan, develop and deliver a usable product. Students demonstrate what they have learned in the classroom now on an active project. CSSE 497 is the first of the three terms for this year-long project. Students are graded each term for their performance and participation on the teams, as well as using input from the clients.

Course Description: Group software engineering project requiring completion of a software system for an approved client. Tasks include project planning, risk analysis, use of standards, prototyping, configuration management, quality assurance, project reviews and reports, team management and organization, copyright, liability, and handling project failure.

Learning Outcomes

Upon successfully completing this series of senior project courses, a student should be able to.

1. Demonstrate effective Communication Skills
 - **Reading:** read technical documents and offer constructive criticism of their content and style

- **Writing:** write several different types of technical documents
 - **Oral presentation:** prepare and deliver technical material at the appropriate level of detail
2. Demonstrate effective Management Skills
 - **Leadership:** lead a small software team (if team leader) or ability to support the leadership of the team (if not)
 - **Time management:** estimate and monitor personal time across multiple tasks
 - **Meeting facilitation:** lead and participate in small groups in constructive meetings
 - **Estimating:** estimate effort required to complete technical tasks
 - **Risk:** assess project risks and plan mitigation strategies
 - **Planning:** prepare a feasible plan for the accomplishment of several technical tasks
 - **Monitoring:** track the progress of several tasks according to a plan
 3. Apply Technical Skills on a real-world problem.
 - **Analysis:** analyze technical requirements and proposals for feasibility and to model the consequences of proposed solutions
 - **Design:** construct appropriate abstractions of problems and solutions
 - **Coding:** produce and inspect implementations of software according to project standards
 - **Testing:** prepare test plans and to participate in both unit-level and system-level testing activities
 4. Demonstrate key areas of Professionalism
 - **Ethics:** identify and prevent unethical professional behavior
 - **Intellectual property issues:** make appropriate professional judgments regarding choice of methods for protecting intellectual property
 - **Social issues:** evaluate and avoid possible negative social aspects of a software product
 - **Relationships with clients:** interact with clients in a professional manner
 - **Teamwork:** working collaboratively with the team to deliver the project and accomplish other project goals.

Prerequisites

CSSE 371 and CSSE 374. (Software development experience, and an ability write and communicate effectively will make this course more meaningful.)

Textbooks

Readings may be assigned from relevant papers (e.g., case studies).

Course Evaluation and Feedback

Please feel free to provide feedback about the course at any time. If you feel uncomfortable talking with me directly, there is an anonymous feedback box under the Moodle account for this course where you can provide feedback throughout the term; I check it regularly and will try respond to feedback in a reasonable time. There may also be anonymous plus-delta evaluations of the course where you can offer suggestions on how to improve the course and its delivery.

Grading:

- | | |
|----------------------------------|-----|
| • Meetings | 20% |
| • Project reviews & deliverables | 50% |
| • Presentation(s) | 15% |
| • Client Feedback | 15% |

Expectations

Students will be expected to attend and participate in senior project related meetings. Students will be required to use the CSSE497 course website on Moodle to obtain relevant information, and interact with instructor and other students. Announcements and assignments will be conveyed via Rose-Hulman email addresses and/or posted on the website. Students will be expected to work on most assignments with other team members.

Assignments

Project assignments are somewhat different than our usual classes. Please provide access to your software project material/artifacts in your configuration managed repository to the instructor. Some assignments may be requested separately (e.g., a response to a case study). Unless otherwise instructed, please post these on Moodle in the associated drop boxes. While this course is demanding, it is also rewarding for those that want strong understanding of software engineering as a discipline.

Late Submissions

Please note that project assignments and deliverables will be due at the specified time on the specified day, in the format agreed upon with advisor. Late assignments and project deliverables will also not be accepted (receiving a zero for grade).

Academic Integrity

CSSE Honesty Policy (see <https://www.rose-hulman.edu/class/cs/csse304/honesty.html>) governs class and performance. Joint study is allowed (even encouraged) on some items as expressed by the instructor; however, each student must produce his or her solutions individually. Students must not collaborate on tests or homework that is passed in unless directed by the instructor.

Attendance Policy

Attendance of meetings and events are mandatory (unless with a legitimate excused absence such as illness). If you cannot make it to meeting or planned event, you are still responsible for all information covered in the meeting or event. Students who have more than 4 unexcused absences will receive a final course grade reduced by up to one full grade level; a student whose total absences (excused and unexcused) exceed 8 will fail this course.

Valid Excuses

A valid excuse consists of a memorandum on Institutional letterhead from the Dean of Students. Job and graduate/professional school interviews, attending scientific conferences and Institute-sponsored activities are also valid excuses provided that every attempt has been made to avoid missing major assignments and examinations, and the student notifies the course instructor in writing at least one week in advance of the event. Verified illness and exceptional circumstances are, of course, valid excuses if a confirming memorandum from the Dean of Students is provided within one week of the illness/circumstance.

Writing

Written communication is important in CSSE 497, as it is in the software profession in general. Remember that a software document has several unique and important characteristics:

1. Technical documents are often the result of group authorship; thus, it requires planning and final editing.
2. Specificity and organization are more important than flow; hence technical documents are often ordered around lists and tables rather than paragraphs.
3. Documents are often the reader's only source of information on the particular subject or product; hence they must be thorough and complete.
4. Documents are often used to answer specific questions; hence, they should facilitate finding specific pieces of information (navigation).
5. Documentation must bridge from general specifications to particulars of implementation and operation, hence it must make abstract concepts concrete and make concrete facts fit generalized concepts.
6. Documentation can be presented in many forms: online via HTML, MS help files, just plain text, and on paper as reference manuals, tutorial, quick reference guides, etc. It is important to choose the correct medium and even more important to write to fit the medium.

You can always drop by your project advisor's office if you have any questions regarding your documents. I would be happy to look at it and suggest improvements. You should also be aware of the service provided by the Learning Center.

Professionalism

As would be expected in the workplace, you are asked to behave in a professional manner. This includes your appearance, such as your apparel, your hygiene, as well as potentially offensive computer desktops. We furthermore ask that you turn off your cell phones during meetings, let people finish their turn talking and do not be disruptive in other ways.

Violations of this policy will have a negative impact on your "Meetings" score.

Caveat

The instructor reserves the right to modify the course content, schedule, policies, etc. outlined in this syllabus.

Tentative Schedule of Course Deliverable: Please check online for latest updates daily. Also, other meetings, such as client meetings, team meetings, and junior project mentorship meetings will be established separately.

Week Date	Project Assignments & Deliverables Due	Topic Content & <i>Notes</i>
0-Th-Fri 9/5-6	Establish Time(s) for Weekly Meetings (Client, & Team meetings)	Introductions & expectations By 9:00am Wednesday, 1 st week
1-Wed 9/11	Kick-off Presentation for Senior Project Teams	-Presentation <i>during 6-7th periods</i>
1-Wed 9/11	Introductory Weekly Advisor Meeting	Outline schedule & expectations. Status of Project, Future tasks, Outstanding Issues, resources.
2-Wed 9/18	First Weekly Regular Weekly Meeting	(1) Project Management URL (i.e., Trello) (2) Repository URL provided (3) Project Communication URL provided (e.g., Slack) (4) Status of Project (5) Project's upcoming tasks (6) Issues facing project (7) Needed Resources (8) Discussions on Problem Statement & Client
3-Wed 9/25	-Configuration Management approach -Preliminary Project Metrics	Discussions on Metrics
3-Fri 9/27	-Preliminary Problem Statement -Baseline Software Project Plan	
4-Wed 10/2	-Problem Statement, Requirements, Features	Discussions on Problem statement, requirements, and features.
4-Fri 10/4	-Baseline Requirements with release planning approach.	
5-Wed 10/9	Early demo of features so far...	Mid-term Peer-Evaluations
10/10 - 10/11	Fall BREAK	
6-Wed 10/16	Requirements Document Review (Bring document to Weekly meeting)	
7-Wed 10/23	-Problem Statement, Requirements, Features Panel Review Presentation	-Presentation <i>during 2nd-9th periods</i>
7-Fri 10/25	Requirements Document Signoff (by client)	
8-Wed 10/30	-Initial Architecture Document	
9-Wed 11/6	-Demo Working Prototype	-Presentation <i>during 7th-9th periods with group or in weekly advisor meeting (TBD)</i>
10-Wed 11/13	-Working prototype shown to client	-Final Peer Evaluations -All code and documents submitted and ready for grading.
Finals Week	TBD	