

Software Engineering Capstone Project Course
SEG4910/SEG4911 Course Sequence
(previous course codes: SEG 4000, SEG4912/SEG4913)
Rules for Projects

This document contains the rules for the Software Engineering Capstone Project. Changes to the rules may be added to the document. Please consult new versions of the document.

Changes as of May 2010: Clarified that the 20% evaluation by the course coordinator can combine an individual grade for each team member with the overall group grade based on their participation, effort, performance and overall professional attitude.

Changes as of May 2009: Minor typos corrected. Length of presentations clarified to be 20 minutes, including time for questions. Clarifications of expectations around individual group meetings.

Changes as of May 2005: The course codes for the Capstone Projects have been changes to SEG4912 (from SEG4910) and SEG4913 (from SEG4911). The courses remain the same except that the number of credits have been changed. Revised schedule to better reflect the division of the work between the two courses for each student stream (non coop and coop streams).

Changes as of November 2004: Minor edits to reflect that the project is now completed over two single semester courses (SEG4910/SEG4911) instead of a single two semester course (SEG4000). Other than the course codes, and the fact that students will get a grade after the first semester, there are no changes to the rules. Students are required to take SEG4910/SEG4911 in sequence and work on the same project with the same group members for both semesters just as they did with SEG4000.

Change as of December 2003: Minor edits to correct spelling mistakes and clean up wording. Clearer definition of the marking scheme, lectures, and deliverables for six milestones in anticipation of breaking the course into two one semester courses.

Change as of January 2003: Emphasis on impact analysis and risk assessment to address concerns of CEAB. Also emphasize on iterative development with 6 milestones. Group size is now larger. More examples of special roles in team structure section. Introduction of 2 quality assurance presentations, one each semester.

1. Group size

Students will work in groups of between 4 and 6. Exceptions will only be considered in exceptional situations.

2. Kinds of projects that are acceptable

All the projects must involve serious software engineering work. You must do requirements analysis, design, implementation, testing and deployment. However the details will depend heavily on the project. A project could be a prototype of a more complex system, or a final version of a simpler system. It could also be an enhancement to an existing system. A process based on the Unified Process will be presented in the class. The waterfall model is not recommended; an iterative, spiral or star approach is preferred. All aspects of quality will be important, including maintainability, usability and reliability. You will also have to use project management skills to estimate costs, plan schedules, make sure you don't try to do too much etc.

3. Customers

Each project has to have at least one defined customer -- the person who has the problem you are solving. This could be a professor, somebody from a company, or the 'open market'. In the latter case, you have to do a market analysis and actually find some people who will review requirements and prototypes and beta-test your product (these people should not be students in SITE). You will spend a lot of time interacting with your customer(s); they will be asked to read your final report, act as beta testers and complete two questionnaires that will be used to help determine your final mark.

4. Type of software

Anything goes as long as it involves real software development. Some possible types of software include data processing, MIS, personal productivity, e-commerce, telecom, real-time, embedded, games etc. However, for some kinds of software you will clearly have a harder time delivering something, so you will have to justify the feasibility of the project. Web sites, per se, are *not* considered valid projects, but the web can be used as the user interface provided there is useful programmed functionality in the site (CGIs or other Server capabilities, or Java applets). Projects that purely involve research (e.g. analyzing an algorithm) are not acceptable, although developing software to *support* researchers would be acceptable if it meets other criteria.

5. Team structure

The SEG project coordinator will act as if he were the CEO of a small company, where each team is a unit of that company. The co-coordinator will veto things of which he or she disapproves, and also give students general advice etc. One member of each team will normally be elected 'project manager'. This does not mean that that person can order the others around, just that he has the prime responsibility for project management. Others on the team may take other specialized responsibilities such as 'chief programmer', 'user interface expert', 'documentation manager/technical writer/configuration manager', 'quality assurance manager', 'requirements manager'. Part of your final report will be a section describing the work each person performed.

6. How will projects be found?

This can occur in several ways. Firstly, if any group of students has a project in mind, they can suggest it to the co-ordinator. Secondly, the coordinator will approach industry and faculty members to see if they have any ideas. The co-ordinator will accept projects that relate to a student's existing employer, provided: a) All team members work for the same employer, b) the SEG project coordinator retains the ability to direct your work, with your employer being considered the customer, c) you can stick to the same schedule as other students, and d) the employer declares in writing that the students will be able to manage their own project according to these rules, and that the project can continue even if the employer changes plans regarding for what it is willing to pay financial compensation.

7. Financial compensation and intellectual property

Unless the students and customer have an agreement (explicit or implicit) to the contrary, SEG project students retain the intellectual property rights to their project (but only those aspects of the project they actually worked on). They may therefore sell it or derivatives of it for profit. Students have a responsibility to let their customers know this, and give the customers the opportunity to negotiate some alternative arrangement. Students may, for example, receive financial compensation for their project work; if they receive such compensation then the customer will normally retain the intellectual property rights. Any intellectual property arrangements must be disclosed to the SEG project supervisor. In all cases, the SEG project supervisor must be granted rights to read and run results of the project.

8. Schedule

For regular students (non-coop) (Starting in September of any given year)

SEG4912 (Fall term)

- Compulsory initial group meeting: 3-4 days after classes start.
- Determining the subject matter: Between April and Sept 15.
- Work on project: Sept to December
 - 3 milestones with deliverables, one every 4-5 weeks.
 - Project definition , Analysis report, Quality Assurance Presentation (Strategy and Framework)
 - Each milestone is meant to be iterative, show progress in terms of a “running” system
 - Regular status reports for each milestone include screenshots (“Mockup”, “Hello World”, “Demo”)
- Intermediate evaluation by customers: end of December.

SEG4913 (Winter term)

- Work on project: January to April
 - 3 milestones with deliverables, one every 4-5 weeks.
 - Winter: Design report, Quality Assurance Presentation (Measuring Progress), Final Report
 - Each milestone is meant to be iterative, show progress in terms of a “running” system
 - Regular status reports for each milestone include screenshots (“Alpha Release”, “Beta Release”, Final Release)

- Final report due: April, of the following year.
- Final evaluations by customers: Same date as final report.

For coop students (starting in January of any given year)

SEG4912 (Winter term)

- Compulsory initial group meeting: 3-4 days after classes start.
- Determining the subject matter: Prior to January 15.
- Work on project: January to April
 - 3 milestones with deliverables, one every 4-5 weeks.
 - Project definition , Analysis report, Quality Assurance Presentation (Strategy and Framework)
 - Each milestone is meant to be iterative, show progress in terms of a “running” system
 - Regular status reports for each milestone include screenshots (“Mockup”, “Hello World”, “Demo”)
- Intermediate evaluation by customers: end of April.

SEG4913 (Fall term)

- Work on project: September to December
 - 3 milestones with deliverables, one every 4-5 weeks.
 - Winter: Design report, Quality Assurance Presentation (Measuring Progress), Final Report
 - Each milestone is meant to be iterative, show progress in terms of a “running” system
 - Regular status reports for each milestone include screenshots (“Alpha Release”, “Beta Release”, Final Release)
- Final report due: December, of the following term
- Final evaluations by customers: Same date as final report.

9. On peut faire le projet en Français.

10. Workload

Each person to put in about 3-4 person-weeks of work (full-time equivalent) for each of the two semesters of the project. This is the normal workload for a 1 semester 3-credit course.

11. Grading

Your grade will be determined as follows.

- 20%: Satisfaction of customers. Computed from two questionnaires completed by your customers worth 10 marks each. One at the end of each of the courses (i.e each semester). This questionnaire will ask about all aspects of the work including whether the customer's problem was adequately solved, the customer's perception of the quality of the software, whether the students communicated well with the customer etc. If you do work without a defined customer, then you get 0 for this component. If you have multiple customers, then they will all complete the questionnaire, and the average value will be used.

- 20%: Evaluation of course coordinator based on the effectiveness of project management and professionalism of the team. There will be a mark out of 10 assigned at the end of each course (i.e. each semester). You will be asked to give regular reports to the co-coordinator as the project progresses. At the end of the project, the coordinator will assess the project management aspects of these. Marks will be given for such aspects as regularity in delivery of reports, quality and accuracy of schedules and cost estimates, effectiveness of distribution of work to team members, and overall professional attitude. At the discretion of the course coordinator, this part of the evaluation can combine an individual grade for each team member with the overall group grade based on their participation, effort, performance and overall professional attitude.
- Team member adjustment: If team members agree that one member of a team contributed more than others, then that member's mark may be increased by up to 5%, with the other team members' marks being reduced to compensate. Such an agreement must be arranged and approved no later than the design deliverable due date (e.g. end of the first month of the second semester).

The following items will be judged by the coordinator, based on the six deliverables (three each semester).

- 10%: Quality of presentations: Each team will present their work. The presentations will be rated by the co-coordinator and other faculty members or customers who attend.
- 20%: Quality of the software's design, including maintainability, usability and reliability. In some projects other aspects of quality such as safety and efficiency will be of great importance. You should provide analyses that prove you have reached satisfactory levels of quality (e.g. the results of usability evaluation, algorithm analysis, proofs of program correctness and/or performance evaluation). The marker will also study your design, looking for weaknesses.
- 10%: Quality of the writing and graphics. This includes writing style, clarity and effectiveness of communication.
- 10%: Completeness of the written material. Good marks if you have requirements, design, test cases, user manual, report on usability evaluation etc. The exact components of the written material will vary depending on the needs of each project. You have to justify the documentation you produce. Also, don't produce unnecessary documentation. Material delivered late will not count.
- 10%: Complexity/difficulty/innovation of project: If you get a lower mark on satisfaction of customers or quality of the software (e.g. the project is not working properly) because you ran into technical difficulties, then you may be able to compensate in this category. The default mark in this category is 5 marks. A project judged with lower than average complexity will get less than 5; a project with higher than average complexity will get more than 5.

Please note that the above grading scheme provides 100% for the project. A mark will be assigned to each of the two courses based on half of the project grading, that is, the grade for SEG4912 will be based on the first half of project grade, while the grade for SEG4913 will be based on the second half of the project grade.

12. Proposals

In order to officially start work on your project, the SEG project coordinator must accept a written proposal. This should be reasonably detailed and must be accepted by the project start date. You should submit it as soon as possible, preferably by Sept 5 for non-co-op and Jan 5 for co-op students. This gives time for feedback and resubmission before the final Sept 15 (or Jan 15) deadline. The proposal must include:

- Names, student numbers and co-op/non-co-op status of the group members. Also describe each member's role on the team.
- Title, and a few paragraphs describing the project, including a clear problem statement.
- Names and backgrounds of customers, and plan for interacting with the customers. If your customers are from off campus, include their addresses and telephone numbers. If you have solicited customers (e.g. in an open-market project), include a letter from them indicating they agree to evaluate the resulting software, as beta-testers.
- If you are interfacing with non-standard hardware or software, describe this.
- If you are working on a part of a larger project (e.g. with electrical engineering students doing the hardware component), describe the larger project.
- Engineering challenges and risks you expect to face, and how you anticipate addressing them.
- Impact Analysis: what are the potential liabilities and benefits of the project from a business, environmental, financial, legal, and end-user perspective. What standards and/or legislation are relevant to your project?
- Development process you plan to follow. E.g. RUP, SCRUM, spiral model etc. Give some details.
- How you plan to evaluate your work. It is suggested that you set measurable quality objectives, and describe procedures you will follow to verify you have met these objectives. Your evaluation procedures can include formal reviews or inspections, analyses of various kinds, and usability evaluation (videotaped, heuristic and/or cognitive).
- Schedule for the work, including documents you plan to deliver, and on what dates. Gantt and Pert charts can help convey this information.

You will be permitted to make changes to the above as you develop the project, although the original proposal, and any changes, is subject to approval by the coordinator.

13. Delivery dates

You will be expected to deliver to the coordinator six reports, spaced about a month apart. Co-op students will have a 5-month gap between the third and fourth report. You will not be reminded to deliver the reports, you will simply be penalized on the project management component of your mark if you do not deliver reports on a regular monthly basis. Each report should, as a minimum contain:

- Summary of the work completed since the last report.
- Summary of issues you have had to consider and resolve since the last report. Include key requirements and technical challenges that have required discussion.
- Time spent by each team member.

- Summary of interactions with customers, users and other people.
- Update to any of the information given in the original proposal. In particular, you must refine the schedule -- giving greater detail about what you plan to do in the next month.
- Any documentation you are scheduled to deliver (e.g. requirements, design, test plan, user manual, source code, usability evaluation, etc.)

14. Meetings with customers:

Groups should meet with their customers on a weekly or bi-weekly basis, and more often when requirements are being gathered and prototypes are being evaluated. If the customer is at a distant location, then email communication is acceptable, as long as there are monthly telephone communications and at least two physical visits.

15. Scheduled meetings with the SEG project coordinator(s):

These will be held on a regular basis and will be compulsory for all students as indicated in the tables below. The following will be the general schedule for the meetings. Students from both streams are encouraged to attend all meetings. Each meeting will fit into a 1.5 hour pre-assigned time slot.

FALL SESSION	
Regular Stream Students (SEG4912)	Coop Stream Students (SEG4913)
Early September	
Regular stream students meet other regular stream members to discuss project proposals and find partners if they have not yet done so.	Review of Design Deliverable. Lecture on Managing Deployment and iterative releases. Project Status update.
Overview of course. Lecture on iterative development and managing expectations with customer. Project Definition Deliverable.	
Late September/Early October	
Lecture on prototyping and analysis of requirements. Analysis Deliverable.	Lecture on testing, quality assurance reporting and stabilizing a release. QA Presentation II Deliverable.
Late October	
Attend QA Presentation II by Coop students.	QA Presentation II for Coop stream (both streams attend).
Late October/Early November	
Lecture on quality assurance frameworks and strategies, software process, build and test environments. QA Presentation I.	Lecture on post mortems, lessons learned, documentation, deployment issues. Final Report Deliverable.
Late November/Early December	
QA Presentation I for regular stream (both streams attend).	Attend QA Presentations I by regular stream students.

	COOP Stream wrap up. Final demos. Course Post mortem).
--	--

WINTER SESSION	
Regular Stream Students (SEG4913)	Coop Stream Students (SEG4912)
Early January	
Review of Design Deliverable. Lecture on Managing Deployment and iterative releases. Project Status update.	Coop stream students meet other regular stream members to discuss project proposals and find partners if they have not yet done so.
	Overview of course. Lecture on iterative development and managing expectations with customer. Project Definition Deliverable.
Late January/Early February	
Lecture on testing, quality assurance reporting and stabilizing a release. QA Presentation II Deliverable.	Lecture on prototyping and analysis of requirements. Analysis Deliverable.
Late February	
QA Presentation II for Regular stream (both streams attend).	Attend QA Presentation II by regular stream students.
Late February/Early March	
Lecture on post mortems, lessons learned, documentation, deployment issues. Final Report Deliverable.	Lecture on quality assurance frameworks and strategies, software process, build and test environments. QA Presentation I.
Late March/Early April	
Attend QA Presentation I by Coop students.	QA Presentation I for coop stream (both streams attend).
Regular Stream wrap up. Final demos. Course Post mortem).	

16. Quality Assurance Presentation I:

You will have 20 minutes to present your plan and approach for addressing quality assurance in your project to the class. You will demonstrate the traceability of your requirements to a) the design and architecture of the system and b) the test plan, testing strategies and process, and test tools that will be used to verify the requirements and validate the design. Allow for up to 5 minutes of questions during your presentation

17. Quality Assurance Presentation II:

You will have 20 minutes in which you should highlight the main difficulties you encountered, and how you overcame them. The objective will be to help the other class members learn something from your experiences. You will be expected to define the progress and quality of what you have developed in terms of a Quality Assurance

Report (that is done for each milestone in second semester). Allow for up to 5 minutes of questions during your presentation

18. Presentations in general:

Consider the following to help improve your presentations:

- Use Powerpoint or a similar tool.
- Understand the intent of the presentation and get to the point. Do not take up a lot of time with uninteresting background. You may have done a lot of work, but focus on a few points about which your audience will be most interested (leave full details for a written report).
- Make sure font size is at least 32 point for slide headings and 20 point for other text.
- Make sure no point takes more than 2 lines. Don't expect readers to read a long sentence - use point form.
- Use diagrams (e.g. UML or pictures of screens) as much as possible. You can point to these and explain details of them.
- Plan on using one slide for each 1-3 minutes.
- Make sure you proofread your slides so they do not have spelling or grammar errors.
- Your slides should only highlight key points of what you will say orally. Make sure you don't 'read' your slides.
- Practice your presentation in advance in front of other members of your group, plus other friends. Make sure you get the timing right.
- Practice speaking up and projecting your voice. Try to put energy into your presentation. Convince your audience that you are enthusiastic about what you are trying to say.
- Each group member should be involved in the presentations during the course, however it is acceptable for a single group member to do one presentation, while the next presentation is done by another group member.

19. Individual Group Meetings

Groups may also meet with the coordinator individually on an ad-hoc basis. Generally speaking, it is expected that groups will schedule such a meeting at least once a month in addition to regularly scheduled meetings when good progress is being made, and that they will schedule even more meetings, as needed, if problems arise. Please try to give 5 days notice of a desired meeting, but if problems occur the coordinator should be contacted immediately by email, if not in person. If the coordinator is not satisfied with any report, then he or she may request a meeting.