SFWRENG 4G06 (Software Design IV — Capstone Design Project)

Spencer Smith

September 3, 2024

This course outline contains important information that will affect your grade. You should retain and refer to this outline throughout the term. It is your responsibility to be familiar its contents.

1 Instructor

Dr. Spencer Smith
Office: ITB/167
E-mail: smiths@mcmaster.ca
Web: http://www.cas.mcmaster.ca/~smiths
Office Hours: By appointment (see schedule)

2 Teaching Assistants (TAs)

Christopher Schankula (schankuc@mcmaster.ca)
Lucas Dutton (duttonl@mcmaster.ca)
Yiding Li (liy443@mcmaster.ca)
Hunter Ceranic (ceranich@mcmaster.ca)

3 Calendar Description

Student teams prepare the requirements, design, documentation, and implementation of a software system taking economic, health, safety, legal, marketing factors into account. Students must demonstrate a working system and convincing test results. Software project management.

4 Mission

The SE capstone provides students with an opportunity to integrate what they have learned in earlier courses, deepen their understanding of that material, extend their area of knowledge and to apply their knowledge and skills in a realistic simulation of professional experience. The emphasis of this course is on design and the design process.

4.1 Learning Objectives: Postcondition

You can find the learning objectives in the course repo.

4.2 Learning Objectives: Precondition

The *precondition* of the course is the set of university-level learning objectives that each student is expected to have achieved before the start of the course. The capstone builds on all previous courses, so the precondition is the set of postconditions from all previous courses. Specifically, each student should have experience with requirements documentation, design, implementation, and testing. Depending on the project specialized engineering knowledge may also be necessary.

4.3 Avenue, Teams, GitLab and GitHub

This course will be administered via Avenue to Learn.

Students should be aware that, when they access the electronic components of this course, private information such as first and last names, usernames for the McMaster e-mail accounts, and program affiliation may become apparent to all other students in the same course. The available information is dependent on the technology used. Continuation in this course will be deemed consent to this disclosure. If you have any questions or concerns about such disclosure please discuss this with the Instructor.

It is the student's responsibility to be aware of the information on the course's Avenue to Learn page and to check regularly for announcements.

The primary purpose of Avenue will be for maintaining grades. Most of the course content will be maintained in a public git (CAS GitLab) repository:

https://gitlab.cas.mcmaster.ca/courses/capstone

In addition, we'll make use of a GitHub project template located here:

https://github.com/smiths/capTemplate

A google calendar is also maintained with the schedule of lectures and deliverables for the capstone course. We will use Teams for some tutorial meetings and for discussion.

Our communication policy is that questions that are relevant to other students in the class should be posted on Teams. That way everyone can benefit from the discussion. Questions that are relevant to only one team, or one individual, should be sent to the instructor or TA (as appropriate) via e-mail. Please send only normal McMaster e-mail; do not send mail via Avenue. Do not use Teams for direct messages to the instructor.

5 Required Materials

1. No required textbook.
2. Some projects may require students purchase some equipment/supplies, but for fairness the total expenses should not to exceed $750 per team. (You should keep any receipts in case you are asked to produce them.)

6 Other Resources

- Carlo Ghezzi, Mehdi Jazayeri, and Dino Mandrioli. *Fundamentals of Software Engineering*. Prentice Hall, Upper Saddle River, NJ, USA, 2nd edition, 2003 (https://www.amazon.ca/Fundamentals-Software-Engineering-Carlo-Ghezzi/dp/0133056996)
- Daniel M. Hoffman and Paul A. Strooper. *Software Design, Automated Testing, and Maintenance: A Practical Approach*. International Thomson Computer Press, New York, NY, USA, 1995 (https://gitlab.cas.mcmaster.ca/smiths/pub/-/blob/master/HoffmanAndStrooper1995.pdf?ref_type=heads)
- Hans van Vliet. *Software Engineering (2nd ed.): Principles and Practice*. John Wiley & Sons, Inc., New York, NY, USA, 2000. ISBN 0-471-97508-7 (https://www.amazon.ca/Software-Engineering-Hans-van-Vliet/dp/0471975087)
- SE 2AA4 / CS 2ME3 Course Notes

7 Forming a Team

The first step is to form a team of four to five students with broadly similar interests. In some cases it may be possible to include team members from computer science or from other engineering departments. If you are interested in working with a teammate from CS, or from outside of CAS, you will need to discuss your specific proposal with the course instructor.

As you will be working with your partners for two terms, select people you can get along with. Some questions you should ask of yourself and your future teammates are as follows:

- Will your teammates have adequate time to meet and fully participate in the project, including classroom activities and meetings with stakeholders? Are there enough gaps in everyone's schedules to hold meetings? Students on co-op terms may not be available for in-person meetings, or have restrictions on the hours they can work on the project.
- Do your teammates have the same, or at least compatible, work habits? When will you work together? All-nighters at the last minute? Regular afternoons? Make sure you pick people with a similar work style, or at least an approach that you can live with.
- Do your teammates have compatible conflict management styles? You should assess your conflict management style and that of your teammates using the assessment tool available in the course repo.
- Do you want to produce a world-class deliverable, or just a good school project? There is no shame in producing a simple yet solid work, learning the principles of good project planning and execution at the same time. However, if your course load permits, this might be your opportunity to show the world — and future employers — what you can do. Either approach is acceptable, but be sure you and your colleagues share the same goals and aspirations, or you will all be frustrated.
- Does your potential teammates provide a good combination of talents and interests, so that collectively you can handle the different aspects of the project. Our projects will combine technical, interpersonal and English writing skills. You should make sure all of these skills are available within your team.
- Do your teammates share your interest in the theme/topic of your project?

One note of caution: good friends do not always make good teammates! Look for people who complement your strengths and weaknesses, not just the people you know.

8 Selecting a Project

You can potentially find a project from the list of project proposals, or you can propose your own project. Student proposed projects can come from interaction with CAS faculty members, outside-CAS faculty members, from industry, or from your own ideas. If relevant for your project, you should enlist the assistance of faculty, staff, or people from industry. The instructor can often suggest useful contacts. A project may be an entirely original concept, continue an older topic, or repeat an older project, but with significant improvements.

Projects can be carried out and supervised off-campus. Projects of this sort often arise naturally as the result of summer employment. To avoid complications, please be sure to discuss any industry collaborations with the instructor and the McMaster Industry Liaison Office (MILO) as soon as possible.

One final note about project selection: while there is a broad range of acceptable topics and approaches, each project must demonstrate some exceptional quality that will form the basis for its evaluation. If the project is academic in nature, finding the solution and implementing it will likely be sufficient. Commercial projects, however, may have technically simple solutions; in such a situation, the project group must do an exceptional job of running the project, collecting requirements, human computer interaction, testing and communicating with the supervisors. All projects have to be approved by the instructor to ensure that they will be sufficiently sophisticated, especially with respect to design, for the capstone course.

9 Onboarding Procedure (Team Formation And Project Approval)

Once you have determined your team composition, you communicate this information through a pull request to the course repo. Specifically, you modify the TeamComposition.csv file and create a pull request. Your team is official once the pull request has been accepted. The team composition information should indicate which team member will act as the *team liaison*. The team liaison is the individual who will be contacted with team specific information. The liaison will have the responsibility of sharing this information with their teammates. The first line of the `TeamComposition.csv` file shows an example of the information that is required.

Once your team has determined the project they wish to pursue, this information should also be entered in the TeamComposition.csv file, followed by a pull request. (The team membership and project information can either be conveyed in one pull request, or two separate pull requests.) The instructor will review the project proposal and either request changes or merge the pull request.

Every project should have a project title and a project description. The project description should be 100-200 words that explain the project. The description must make it clear that the project has an adequate design element for a team of senior students. Most of the The curated list of projects have already met these criteria, but you still need to create a project description file as part of your pull request. Even for the curated projects, instructor approval is required, since some of the projects require scope clarification based on the team's interests and abilities.

10 Deliverables

The following deliverables define the structure of the course. The dates given are tentative. Any changes to the plan will be discussed in class and reflected in the course's google calendar.

In general, each deliverable will be graded twice, once for Revision 0 and once for Revision 1. The first evaluation is intended to be mostly formative; that is, it is a teaching and feedback opportunity. The second evaluation is summative; that is, the second evaluation is intended to accurately grade what you achieved. As a consequence, you will find the grading on Revision 1 to be "tougher" than the grading on Revision 0, and the weighting is higher for Revision 1. **Please note that you should NOT wait for one deadline before starting on the next deliverable.** The nature of a course is that we need deadlines and deliverables, but the nature of an open-ended project is that you need to be continually experimenting and iterating. If you forget this, then the deadline for the Revision 0 Demonstration will be a challenging one to meet.

Team Formed, Project Selected	September 16	0%
Problem Statement, POC Plan, Development Plan	September 23	2%
Requirements Document Revision 0	October 9	5%†,‡
Hazard Analysis 0	October 23	3%†
V&V Plan Revision 0	November 1	5%†,‡
Proof of Concept Demonstration	November 11–22	5%*
Design Document Revision 0	January 15	5%†,‡
Revision 0 Demonstration	February 3–February 14	10%*
V&V Report Revision 0	March 7	5%†,‡
Final Demonstration (Revision 1)	March 24–March 30	20%*
EXPO Demonstration	April TBD	10%*
Final Documentation (Revision 1)	April 2	30%*,‡
- Problem Statement		
- Development Plan		
- Proof of Concept (POC) Plan		
- Requirements Document		
- Hazard Analysis		
- Design Document		
- V&V Plan		
- V&V Report		
- User's Guide		
- Source Code		

Written reports are due by 11:59 pm on the deadline date.

Each of the $*$ items represent a grading item where an individual grade will be assessed.

The individual grade will be the team grade multiplied by a team contribution factor. The team contribution factor, as determined by the TA and instructor, will have a value between 0 and 1.2.

Each of the † items represents a grade component where part of your grade will be based on you completing a review of the work of another team. The review of the other team's work will be done by creating issues in their GitHub repositories. Each team will be graded on the quality of the issues they create. Your comments wil be due 2 days after the deadline of the other team's documentation.

In the event that a team has an Non Disclosure Agreement (NDA) for their project, no other teams will review their work, but that team will still review the work of other teams.

Each of the † items will also have an informal presentation to the TA. The purpose of the presentation is to answer your questions and to give the TA a chance to provide preliminary feedback. The presentation will take place during the tutorial time.

Each of the ‡ items represents a deliverable that will include a self reflection appendix. Self reflection is an important component of learning, especially life-long learning.

The instructor reserves the right to adjust the grades for any deliverable by increasing or decreasing every score by a fixed number of points.

For projects that are not traditional software development, such as more theoretical projects, the weighting will follow the above guidelines, but the deliverables will be changed to match the project. Any changes of this nature need to be discussed in advance with the course instructor. Note exceptions in an texttttExceptionsGranted.md file at the top level of the repo. (A blank version of this file is provided in the template repo.) This information is provided so that the TAs are aware of any arrangement that deviates from the norm.

10.1 Challenge Level and Extras

Due to differences in interests and background, teams in this course will select topics with varying levels of complexity in both the required domain knowledge and in the implementation. Less advanced projects are an option, but to get full grades for a lower challenge project, teams should add extra work to the scope of their project. Examples of extra work include usability testing, rigorous code walkthroughs, user manual documentation, formal proof, GenderMag personas, design thinking, etc. If advanced projects do extra work, the extra work is evaluated and added to the deliverable's grade as a bonus.

Notes:

1. All deliverables (except the problem description) are required to have a text based version. The text should then be compiled to a pdf version. Usually this will mean writing all documentation in LaTeX, but some teams may prefer to use org mode, markdown, or AsciiDoctor. Having text based documentation allows the instructor

and TAs to see the contribution of each team member to the project. Presentations can be done in whatever form your team prefers.

2. The names and locations for all deliverables should follow our GitHub project template, unless the instructor grants your team an explicit exception, communicated through the `ExceptionsGranted.md` file.

3. All projects should be publicly available on GitHub, unless the instructor grants your team an explicit exception. (This may come up for teams that are doing an industry project covered by an NDA.)

4. The instructor and the TAs should be added as Collaborators on your project. They need access for grading, and to observe the number of commits and issues closed by team members.

5. All teams are required to use Continuous Integration (CI) implemented via GitHub Actions, unless explicitly given permission by the instructor to not include CI, or to coordinate it by another tool.

6. All team meetings should be planned and documented as GitHub issues. The minimal information for each meeting includes the agenda for the meeting and the attendees at the meeting.

7. Given the course-long goal of improving each project, students should be prepared for criticism during their demonstrations. This criticism is part of the learning process. Please let the instructor know if you have any concerns with the feedback you receive.

8. At the end of the year some teams will be selected to present their projects as part of the May at Mac activities. Any selected teams that agrees to participate in May at Mac will get 2% added to their grades.

9. The instructor reserves the right to re-evaluate any of the deliverables via an oral examination.

10. The final grade will be converted to the 12-point scale recommended by the Registrar.

If there is a problem with the grading of an assignment, please contact the marking TA to discuss it. If the problem cannot be resolved through discussion with the TA, then please contact the course instructor. Grades for assignments will only be changed if you report the problem within two weeks of receiving the grade.

11 Policy Statements

This section summarizes the policy statements for the course.

11.1 Improving the Course

Ideas to improve the course are always welcome. Moreover, if you have problems in the course, please contact the instructors as early as possible.

11.2 Discrimination

The Faculty of Engineering is concerned with ensuring an environment that is free of all adverse discrimination. If there is a problem, that cannot be resolved by discussion among the persons concerned, individuals are reminded that they should contact their Department Chair and the Human Rights and Equity Services (HRES) office as soon as possible.

11.3 Academic Integrity

You are expected to exhibit honesty and use ethical behaviour in all aspects of the learning process. Academic credentials you earn are rooted in principles of honesty and academic integrity. **It is your responsibility to understand what constitutes academic dishonesty.**

Academic dishonesty is to knowingly act or fail to act in a way that results or could result in unearned academic credit or advantage. This behaviour can result in serious consequences, e.g. the grade of zero on an assignment, loss of credit with a notation on the transcript (notation reads: "Grade of F assigned for academic dishonesty"), and/or suspension or expulsion from the university. For information on the various types of academic dishonesty please refer to the Academic Integrity Policy, located at

`https://secretariat.mcmaster.ca/university-policies-procedures-guidelines/`

The following illustrates only three forms of academic dishonesty:

- Plagiarism, e.g., the submission of work that is not one's own or for which other credit has been obtained.
- Improper collaboration in group work, including not doing your share of the work.
- Copying or using unauthorized aids in tests and examinations.

Your work must be your own. Plagiarism and copying will not be tolerated! If it is discovered that you plagiarized or copied, it will be considered as academic dishonesty. The instructor may ask students to defend their written work orally.

11.4 Authenticity/Plagiarism Detection

Some courses may use a web-based service (Turnitin.com) to reveal authenticity and ownership of student submitted work. For courses using such software, students will be expected to submit their work electronically either directly to Turnitin.com or via an online learning platform (e.g. A2L, etc.) using plagiarism detection (a service supported by Turnitin.com) so it can be checked for academic dishonesty.

Students who do not wish their work to be submitted through the plagiarism detection software must inform the Instructor before the assignment is due. No penalty will be assigned to a student who does not submit work to the plagiarism detection software. **All submitted work is subject to normal verification that standards of academic integrity have been upheld** (e.g., on-line search, other software, etc.). For more details about McMaster's use of Turnitin.com please go to www.mcmaster.ca/academicintegrity.

11.5 Course with an On-Line Element

Some courses may use on-line elements (e.g. e-mail, Avenue to Learn (A2L), LearnLink, web pages, capa, Moodle, ThinkingCap, etc.). Students should be aware that, when they access the electronic components of a course using these elements, private information such as first and last names, usernames for the McMaster e-mail accounts, and program affiliation may become apparent to all other students in the same course. The available information is dependent on the technology used. Continuation in a course that uses on-line elements will be deemed consent to this disclosure. If you have any questions or concerns about such disclosure please discuss this with the course instructor.

11.6 On-Line Proctoring

Some courses may use on-line proctoring software for tests and exams. This software may require students to turn on their video camera, present identification, monitor and record their computer activities, and/or lock/restrict their browser or other applications/software during tests or exams. This software may be required to be installed before the test/exam begins.

11.7 Conduct Expectations

As a McMaster student, you have the right to experience, and the responsibility to demonstrate, respectful and dignified interactions within all of our living, learning and working communities. These expectations are described in the Code of Student Rights & Responsibilities (the "Code"). All students share the responsibility of maintaining a positive environment for the academic and personal growth of all McMaster community members, **whether in person or online**.

It is essential that students be mindful of their interactions online, as the Code remains in effect in virtual learning environments. The Code applies to any interactions that adversely affect, disrupt, or interfere with reasonable participation in University activities. Student disruptions or behaviours that interfere with university functions on on-line platforms (e.g. use of Avenue 2 Learn, WebEx or Zoom for delivery), will be taken very seriously and will be investigated. Outcomes may include restriction or removal of the involved students' access to these platforms.

11.8 Academic Accommodation for Students with Disabilities

Students with disabilities who require academic accommodation must contact Student Accessibility Services (SAS) at 905-525-9140 ext. 28652 or sas@mcmaster.ca to make arrangements with a Program Coordinator. For further information, consult McMaster University's Academic Accommodation of Students with Disabilities policy.

11.9 Requests for Relief for Missed Academic Term Work

McMaster Student Absence Form (MSAF): In the event of an absence for medical or other reasons, students should review and follow the Academic Regulation in the Undergraduate Calendar "Requests for Relief for Missed Academic Term Work".

11.10 Academic Accommodation for Religious, Indigenous or Spiritual Observances (RISO)

Students requiring academic accommodation based on religious, indigenous or spiritual observances should follow the procedures set out in the RISO policy. Students should submit their request to their Faculty Office ***normally within 10 working days*** of the beginning of term in which they anticipate a need for accommodation *or* to the Registrar's Office prior to their examinations. Students should also contact their instructors as soon as possible to make alternative arrangements for classes, assignments, and tests.

11.11 Copyright and Recording

Students are advised that lectures, demonstrations, performances, and any other course material provided by an instructor include copyright protected works. The Copyright Act and copyright law protect every original literary, dramatic, musical and artistic work, **including lectures** by University instructors.

The recording of lectures, tutorials, or other methods of instruction may occur during a course. Recording may be done by either the instructor for the purpose of authorized distribution, or by a student for the purpose of personal study. Students should be aware

that their voice and/or image may be recorded by others during the class. Please speak with the instructor if this is a concern for you.

11.12 Extreme Circumstances

The University reserves the right to change the dates and deadlines for any or all courses in extreme circumstances (e.g., severe weather, labour disruptions, etc.). Changes will be communicated through regular McMaster communication channels, such as McMaster Daily News, A2L and/or McMaster email. It is the responsibility of the student to check their McMaster e-mail and course websites weekly during the term and to note any changes. Your McMaster e-mail is the one with the address ending in `@mcmaster.ca`. This is a separate e-mail address from your Avenue address.