Course Syllabus

✎ Edit

CPSC 410: Advanced Software Engineering

Specification, design, construction and validation of multi-version software systems.

3 credit course. *Prerequisites:*
Either (a) CPSC 310 or (b) one of EECE 310, CPEN 321 and one of EECE 315, CPEN 331.

Contact Information and Helpful Links

Instructor Information

Student Resourc

Name: Caroline Lemieux (she/her/hers)

Email: clemieux@cs.ubc.ca

TAs: Zack Grannan (zgrannan@cs.ubc.ca), Zung Nguyen (dung2000@student.ubc.ca), Partha Paul (pppaul@cs.ubc.ca)

Office Hours: Please see **Piazza** **(https://piazza.com/class/mj7lmga4dwt40p/post/6)** for details

UBC Guide to Canvas **(http://students.canvas.ubc.ca/help/)**

Canvas Student Guide **(https://community.canvaslms.com/d**

Student IT Helpdesk **(https://it.ubc.** **about-it-products-and-support)**

UBC Student Resources **(https://canvas.ubc.ca/courses/17544**

Course Schedule **(https://canvas.ubc.ca/courses/17544** **schedule)**

Piazza **(https://piazza.com/ubc.ca/winterterm**

(https://canvas.ubc.ca/courses/17544

SYLLABUS DETAILS

Welcome to 410!

In this course you will build your software engineering skills by building two projects as part of a 4-5 person team. The topics of the projects are around software engineering tooling; you will need to propose a project within these topicss. In lecture, we will discuss some software engineering and evaluation skills, but primarily provide a deeper dive into

In the course, we will use examples written in Java, but you are free to conduct your project in any language your group is interested in. The Learning Objectives are not specific to Java, and many of the high-level topics apply analogously also to other programming paradigms.

Looking forward to exploring the course together!

Teaching Team

My name is Caroline Lemieux. I joined UBC as an Assistant Professor in July 2022 (after a PhD at UC Berkeley and Postdoc at Microsoft Research, NYC). My research interests are in building developer tools that help improve the correctness, performance, and security of software. I particularly interested in methods that scale to large code bases, including automated testing techniques such as fuzz testing.

Your TAs are Zack Grannan (**zgrannan@cs.ubc.ca (mailto:zgrannan@cs.ubc.ca)**), Zung Nguyen (**dung2000@student.ubc.ca (mailto:dung2000@student.ubc.ca)**), and Partha Protim Paul (**pppaul@cs.ubc.ca (mailto:zwb@cs.ubc.ca)**), all graduate students in the Software Practices Lab at UBC, where they conduct research in Software Engineering and Programming Languages.

Course Overview

This course builds on your existing software engineering knowledge and experience with a hands-on approach to collaborative software engineering. The course is project-heavy and requires you to plan, coordinate and work with a team of your colleagues. You will design, build and evaluate two moderate-sized team projects throughout the course of the term. We will use these as vehicles for exploring what it takes to work with a team break down a software project, and coordinate with other team members in a limited time budget.

The topics of the projects will also provide a grounding in programming language design and implementation, and tools and techniques to aid programmers in understanding and analysing their code. In particular, we will explore *two main focus areas* (one for each project):

1. *Domain Specific Languages* (what they are for, how to design and build them to be useful to your users)
2. *Program Analysis* (extracting, analysing and presenting rich information to programmers about their code's potential behaviours)

Interwoven with these focus areas, we will discuss empirical evaluation methods appropriate to software engineering projects in general, as well as relevant object-oriented programming techniques for implementing languages and program analyses. You will exercise good practices in development and teamwork via the team projects throughout the term, while regular exercises will cover both specific techniques and encourage you to problem-solve, exploring and evaluating alternative designs and implementation choices.

Course-Level Learning Objectives

Upon successful completion of this course, students will be able to:

1. Manage and evaluate a development strategy and timeline for a project built in parallel by a team of programmers.
2. Decompose a software design into modular components and development tasks, enabling separate testing, smooth adaptation to changes and easier final integration.
3. Collaborate as a responsible and effective team and course member, taking shared responsibility for their team successfully delivering on their project goals, and personal responsibility for keeping to deadlines in a timely fashion.
4. Design and implement Domain Specific Languages tailored to a given task and group of users, incorporating early feedback and design revisions into your process.
5. Differentiate static and dynamic techniques for program analysis, contrasting their strengths and weaknesses.

6. Create simple program analyses to help programmers understand their own programming challenges and software engineering processes.
7. Evaluate software engineering research questions and projects via rigorous empirical studies.

Learning Activities and Expectations

This will be a practical course: during the term you will complete two *group* projects (one on each of the focus areas above), as well as roughly-bi-weekly *individual* exercise sheets which will help you form a continuous assessment of your progress throughout the term. You can see a provisional plan for the topics we will cover in the **Course Schedule (https://canvas.ubc.ca/courses/175440/pages/course-schedule)** (subject to change during the term).

Lectures are the only fixed contact hours: on Mondays and Wednesdays during Winter Term 2, from 2:00pm - 3:30pm (note that there is no teaching on the week starting 16th February due to mid-term break).

Lectures will be in person, and will include a mixture of Caroline presenting new content and the course participants trying out exercises (often in small groups). Unless you are sick you are expected to attend and participate in the lectures and these live exercises, which have been designed to help you succeed in the other course components. You are also very welcome and strongly encouraged to ask questions at any time.

If you do have to miss a lecture then please use the slides, Piazza group, discussions with your course colleagues, and office hours to catch up. If in doubt, please reach out and ask for help.

The course will be assessed in *three complementary ways*:

- Two group projects (these will take a large slice of the time you spend on 410). In both cases you'll work in groups of 4-5 and build something to show to the class. Each project is worth 22.5% (45% in total).

- There will be six roughly-bi-weekly individual exercise sheets, together worth 10% of your grade. These will be graded for completion, with one free pass (i.e., only 5 out of 6 exercise sheets need to be submitted). Each of the submitted sheets will be worth 2% of your grade. These serve as your primary preparation for the final exam.
- A final exam, worth the remaining 45% of your grade. The final exam covers all the low-level learning objectives of the course. If there is interest we will arrange an exam Q&A session near the end of term.

You'll find the schedule for exercises and projects on Canvas (under Assignments). The project descriptions will include descriptions of what we will be looking for when we grade them.

Plan to spend at least 6 hours a week on average on this course *in addition* to the contact time, and be sure to plan your individual and team activities carefully from the beginning to avoid falling behind. Note that this is an *average*: some weeks may be more or less intensive and your mileage may vary; you will definitely benefit from spreading the work for each deadline out, especially for the projects.

Make sure to agree clearly with your project groups what your expectations are of each other, communicate regularly and revise your plans as needed; a key purpose of the projects is learning to work together as an effective development team, and you and your team members are collectively responsible for ensuring your team’s success and the wellbeing of your other team members.

Late submissions of projects will not be accepted; all deadlines in the course are firm. Apart from being for fairness to all of the class participants, this will allow us to freely discuss and present solutions once the deadlines have passed.

As a general rule, you're expected to make sure to keep up with these activities and the lecture, participate reliably in your groups, and take advantage of the course support available. If you do feel you might be getting behind, you should catch up using the various forms of course support (e.g. Piazza, office hours, slides). The combination of lecture, exercise sheet and project work (intended to complement one another) means we all need to stick to the timetable, since we'll quickly move on to building on what we've learned and worked on so far.

Waitlist Policy

Please note firstly that *we do not manage the waitlists or which students get a place in the course*; this is handled by the department.

We plan two moves off the waitlist, modulo Workday issues. One move will be on January 6th, and another the morning of Friday 9th January. **If you are not in the course by 1pm on Friday 9th January, we are not able to offer you a spot in the course**. This is because project registrations need to be done by the end of Friday 9th January. If you do not have a place on the course by this time, unfortunately it will not be possible to join us this term. Dropping the course during a project would naturally also cause difficulties for your project teammates - please aim to avoid this!

If you’re considering dropping the course (whether you are registered or on the waitlist), *please try to do so in the first week* (ideally before Friday 9th January), to allow other students the chance to take the course: there are many students still on the waitlist.

Collaboration and Assessment

Be *extremely careful* to work **individually** on the exercise sheets, and **within your groups** for the projects. If you draw on external sources or discussions, you *must cite these* (where appropriate, that may even impress the person grading!). As usual, these sources must not be the work of other course participants.

For your group projects, you are encouraged by default to *write all code yourselves*; if you choose to make use of any parts of the code we develop in class then you must clearly explain this when you submit. Using someone else's uncredited work in your assessments is serious (and unfair on your colleagues), and will result in no marks for that assessment and a referral for academic misconduct according to UBC's policies; this has happened regularly in previous iterations of the course. In particular, be sure not to discuss or share solutions to the assessed exercises until (at least) the day after their submission deadlines; you are better off (and will get more useful feedback) submitting your own best efforts at the exercise sheets, even if they are incomplete. Hopefully none of the above paragraph will need mentioning ever again!

Some links to UBC policies and other sources of support can be found **here (https://canvas.ubc.ca/courses/175440/pages/course-orientation-guide)** .

Generative AI Policies: You will be programming a fair bit in this course, especially for the projects, and might wonder whether/when usage of generative AI tools such as ChatGPT, Copilot etc. is acceptable. Please explicitly mark generated code with comments as follows:

/** Start Code Generated by [TOOLNAME] on [DATE] */

/** End Code Generated by [TOOLNAME] on [DATE] */

If you use AI-based autocomplete (i.e., Copilot), where marking each block of code is not possible, you should document this in the function signature:

/** Contains Code Generated by [TOOLNAME] on [DATE] */

The course is designed to be an opportunity to practice developing reliable code in a collaborative environment, and having a clear understanding of the code you produce is an essential part of that role. For this purpose, we encourage you to focus on writing code yourself. However, if you are a seasoned user of generative AI and believe you can use it while retaining a deep understanding of the code you generate, you are free to try. The instructors and TAs will not help you to debug/understand code which was generated by a tool. You must make sure that your project teammates don't end up in this situation: the code you produce as a software engineer is your own responsibility!

Oral Examinations: The course staff reserves the right to conduct oral examinations of students to evaluate their understanding of their submitted work. The results of these examinations may affect exercise sheet and project grades.

Team Collaboration

Note that in group projects, many things can and often do not go according to plan, and it's very important to **identify and address any problems in your team as as early as possible**. In order to do this, it is important to check-in with your team regularly, and plan to make *regular progress throughout the project time* so that you can identify any issues early. There will be *regular, compulsory short surveys* for you to check in about how you think your team is doing. By default (and in particular, if we don't hear about problems until close to the deadline) **all team members will receive the same grade for each project**.

You may often be able to resolve issues by discussions within your team, but Caroline or the TAs are happy to advise and intervene if you need help with a difficult situation. If you can make the instruction team aware of concerns you have about your group while the project is ongoing, we will be able to suggest ways of resolving issues and steering the team back on course.

In case any team members' contributions are falling behind, do reach out *early* and let us know. Make sure to work and check-in in a way that guarantees you will be able to identify problems *substantially before* the end of the project.

If you do have a problem that requires our intervention, that's OK: we will meet with the team and make expectations clear for the remainder of the project. Usually this is sufficient to get teams back on track. In exceptional cases, if there are further problems during the project after we have intervened, we may consider adjusting the grades of members who are not cooperating with their team.

If you require instructor/TA intervention: contact us **12 days** before the project due date, and arrange a meeting **9 days** before the project due date with all team members. After this, course staff will not be able to intervene in a way that may lead to an adjustment of team member's grades.

Required Materials

No textbooks etc., but to participate in the lecture exercises you should have a computer ready for Java development (from Lecture 2 onwards). You may want to set up an IDE/editor that you're comfortable with in advance. For example, Caroline will use Intelli-J Community Edition for the Java examples shown in the lectures.

Course Support and Participation

This course has been designed to be taught in person, and the content and in-class exercises in the lectures, the individual quizzes and the group projects are meant to *complement one another* to help you succeed; you are expected to attend the lectures in person for this course.

We will offer you support during the course in a variety of ways: you will be assigned a main TA as a standard point of contact throughout the course, and both Caroline and the TAs will offer weekly office hours. These will be posted on Piazza.

Discussions about the course material will take place via a Piazza group (linked on the left); TA office hours will also be kept up-to-date there. It's possible to post on Piazza anonymously (other students cannot see your identity, although the instruction team still can) but you are encouraged to interact publicly where possible, to get the maximum benefit for the whole class. If you have more of a personal concern, you are very welcome to email Caroline or your TA about it. You're strongly encouraged to ask questions during the lectures, too - this always helps and also makes the lectures more enjoyable!

Accessing Support

BC provides resources to support student learning and to maintain healthy lifestyles but recognizes that sometimes crises arise and so there are additional resources to access including those for survivors of sexual violence.

UBC values respect for the person and ideas of all members of the academic community. Harassment and discrimination are not tolerated nor is suppression of academic freedom. UBC provides appropriate accommodation for students with disabilities and for religious, spiritual and cultural observances.

UBC values academic honesty and students are expected to acknowledge the ideas generated by others and to uphold the highest academic standards in all of their actions. Details of the policies and how to access support are available **here (http://senate.ubc.ca/policies-resources-support-student-success)** .

Early Alert: *(text from early alert)* During the term, I will do my best to reach out and offer support If I am concerned about your wellbeing or academic performance. I also encourage you to come and speak with me if you need assistance.

In addition, I may identify my concerns using Early Alert. The program allows academic, financial, or mental health concerns to be identified sooner and responded to in a more coordinated way. This provides you with the earliest possible connection to resources like academic advising, financial advising, counselling, or other resources and support to help you get back on track.

The information is treated confidentially and is sent because I care about your wellbeing and academic success. For more information, please visit **earlyalert.ubc.ca (http://earlyalert.ubc.ca)** .

Your personal health

If you're sick, it's strongly advised that you stay home – no matter what you think you may be sick with (e.g. cold, flu, other).

- If in doubt, a self-health assessment is strongly recommended before attending campus using this tool: **https://bc.thrive.health/covid19/en (https://bc.thrive.health/covid19/en)**
- Please do not come to class if you have COVID/flu symptoms, have recently tested positive for COVID, etc. Your precautions will help reduce risk and keep everyone safer. In case you have any concerns, please feel free to reach out to Caroline at any time; we want to find comfortable and safe ways to work together so that you can prioritize your health and those of our community and still be able to participate and succeed.

If you do miss class because of illness:

- Make a connection early in the term to another student or a group of students in the class. You can help each other by sharing notes (but be careful not to share work on the exercise sheets). If you don't yet know anyone in the class, post on the discussion forum to connect with other students; shortly into the course you will naturally need to be in regular contact with your project team, so this will be a good potential support.

- Consult the class resources on Canvas and Piazza. We will post all of the slides, assessments and any other materials regularly.
- You are encouraged to post questions on Piazza (and to answer others' questions or make suggestions there).
- Come to office hours.
- If you are concerned that you will need to miss a particular key activity due to illness, contact Caroline or one of the TAs to discuss your situation as soon as possible.

If you are sick on a final exam day, you should consider not attending the exam. In this case, you would need to apply for deferred standing (an academic concession) through Science Advising no later than 48 hours after the missed final exam. Students who are granted deferred standing write the final exam/assignment at a later date. Learn more and find the application online: **https://science.ubc.ca/students/advising/concession (https://science.ubc.ca/students/advising/concession)**

For additional information about academic concessions, see the UBC policy here: **http://www.calendar.ubc.ca/vancouver/index.cfm?tree=3,329,0,0 (http://www.calendar.ubc.ca/vancouver/index.cfm?tree=3,329,0,0)**

Online classes (in case of emergency only)

We may have synchronous online sessions for exceptional reasons, notably:

- UBC cancels in-person classes due to weather events (snow);
- Caroline tests positive for COVID or another infectious illness (but is still well enough to teach).

In these cases, we may have a synchronous online session. If this happens, you will receive an email as well as an announcement on Piazza telling you how to join the class via Zoom. You can anticipate that this may possibly have to be a last minute email. In this situation, we will have a TA broadcast the zoom to the classroom, so the classroom will still be available if you can make it to campus.

Finally

Well done for making it to the end of a long syllabus. This term will be a busy one, but with good communication, considerate and careful practice and teamwork we will make it successful (and hopefully fun) together. Do please reach out for support whenever you are worried and we will help wherever we can!

Acknowledgement: UBC’s Point Grey Campus is located on the traditional, ancestral, and unceded territory of the xwməθkwəy̓əm (Musqueam) people. The land it is situated on has always been a place of learning for the Musqueam people, who for millennia have passed on in their culture, history, and traditions from one