

CS3307A - Object-Oriented Design and Analysis Outline

The University of Western Ontario
London, Canada

Department of Computer Science

CS 3307A - Objected-Oriented Design and Analysis

Course Outline - Fall 2017

Course Description

Software development comprises many activities (e.g., technical, managerial, governance, customer relations, market analysis, etc.) among which are those that take requirements as input, and yield, as output, software code that is intended to satisfy the requirements. In this development process is a key artefact, the *design of the system*, that embodies the various components of the system interacting with each other so as to provide the desired solution. Prior to, or even during, designing the system is the *analysis* of the environment in which the system is to operate, including discovering the system's requirements. Arguably, without these two sets of activities – analysis and design -- in the development process, it would be difficult to build large-scale, quality systems.

The most widely-used approach to system design and requirements analysis nowadays is the object-oriented approach. Here, the Unified Modeling Language (UML) notation is used to capture the user interaction with the system being built, and how the system will be structured, and how it will interact within to provide the desired outcomes. Over the years, the software engineering community has identified many useful patterns of how system components are structured and how they interact with one another. Thus, the developer can attempt to reuse these patterns as partial solutions in the design of new systems, saving design time and costs and attaining higher quality. These design patterns can be implemented using object-oriented (OO) programming languages such as Java and C++.

In this course, students will learn about object-oriented analysis and design, and will be introduced to the programming language C++. They will subsequently carry out specific assignments, and a group project where they are expected to design and implement a reasonably large-scale system in the language C++.

Lecture Hours

2:30 - 3:30pm, Tuesdays, TC 141

2:30 - 4:30pm, Thursdays, TC 141

Either (Computer Science 2212A/B/Y) or (Computer Science 2210A/B, 2211A/B, Electrical
Prerequisites: and Computer Engineering 3375A/B, and registration in the fourth year of a BESC program in
Computer Engineering or Mechatronic Systems Engineering.)

Antirequisites: Software Engineering 2203A/B

Note: *Unless you have either the prerequisites for this course or written special permission from your Dean to enroll in it, you may be removed from this course and it will be deleted from your record. This decision may not be appealed. You will receive no adjustment to your fees in the event that you are dropped from a course for failing to have the necessary prerequisites.*

Instructor Information

Instructor: Michael Katchabaw
Office: MC 28H
Office Hours: 11:30am - 12:30pm, Tuesdays and Thursdays
E-Mail: katchab@csd.uwo.ca
Phone: Western extension 84059

Course Texts

While there are no required texts for this course, the following books are recommended as references for this course, and may be available for purchase from the University Bookstore or other sources, such as the Used Book Store:

- [The C++ Programming Language](#), 4th Edition by Bjarne Stroustrup, published by Addison-Wesley Professional, 2013.
- [Programming Principles and Practise Using C++](#), 2nd Edition by Bjarne Stroustrup, published by Addison-Wesley Professional, 2014.
- [Design Patterns: Elements of Reusable Object-Oriented Software](#) by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, published by Addison-Wesley Professional, 1995
- [UML Distilled: A Brief Guide to the Standard Object Modeling Language](#), Third Edition by Martin Fowler, published by Addison-Wesley Professional, 2004.
- [Object-Oriented Analysis and Design with Applications](#), Third Edition by Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young Ph.D., Jim Conallen, Kelli A. Houston. Published by Addison-Wesley Professional, 2007.

Additional references and suggested readings may be provided throughout the course as the project requires them. Please check back to the course website for updates and more information.

Course Topics

The course will address as many of the following topics as time will allow:

- Programming in C++
- Overview of Object-Oriented programming
- The Unified Modeling Language (functional, structural, and behavioural modelling)
- Architectural patterns
- Design patterns (creational, structural, behavioural, ...)
- Object-Oriented metrics and other software metrics
- Quality Assurance (inspections)
- Enterprise-Scale software and collaboration tools (Jira, Confluence, Bitbucket, HipChat, ...)

Lecture Notes

Course lecture notes will be made available in PowerPoint format on the course website on a weekly basis, as they are developed. They are provided as a courtesy by the course instructor. Possessing (and even reading) these notes is not a suitable substitute for attending lectures.

Course Website

The CS3307A website is accessible through OWL at <http://owl.uwo.ca>. Class and project information will be posted on this website on a fairly regular basis. You are responsible for reading this information frequently.

Computing Facilities

Each student will have access to computing facilities administered by the Department of Computer Science and/or Western University. In accepting their accounts, students agree to abide by the Department's [Rules of Ethical Conduct](#). During this course, we may also make use of cloud infrastructure provided either by Western or by Amazon; details on this will be discussed in class.

Note: *After-hours access to certain Computer Science lab rooms is by student card. If a student card is lost, a replacement card will no longer open these lab rooms, and the student must bring the new card to the Systems Group. Likewise, if a student card ceases to provide access where it should, it should be brought to the Systems Group as well.*

E-Mail Contact

We will occasionally need to send e-mail messages to the whole class, or to students individually. E-Mail will be sent to the Western e-mail address assigned to students by Information Technology Services (ITS), i.e. your e-mail address @uwo.ca. It is each student's responsibility to read this e-mail on a frequent and regular basis, or to have it forwarded to an alternative e-mail address if preferred. See the ITS website for directions on forwarding e-mail.

However, you should note that e-mail at ITS (your Western account) and other e-mail providers may have quotas or limits on the amount of space they can use. If you let your e-mail accumulate there, your mailbox may fill up and you may lose important e-mail from your instructors. Losing e-mail that you have forwarded to an alternative e-mail address is not an excuse for not knowing about the information that was sent.

Wherever you receive e-mail, be sure to configure your spam filter to allow e-mail from the instructor's e-mail address given above. Otherwise, important messages could get trapped by your spam filter and missed. This is also not an excuse for not knowing about information that has been sent.

Student Evaluation

Grades will be based on assignment and project work worth 65% and a final exam worth 35%. There will be one individual assignment (15%) and a group project (50%) split into separate deliverables including two intermediate stages (each worth 10%), the final project submission (25%), and final documentation (5%).

To be eligible to receive a passing grade in the course, your mark on the final exam must be at least 40%, and your average on the assignment and project components must be at least 40%. Otherwise, the maximum overall mark you can receive is 45%. To be eligible to receive a grade of C or higher, your mark on the final exam must be at least 50%, and your weighted average on the assignment and project components must be at least 50%. Otherwise, the maximum overall mark you can receive is 58%.

Assignment, Project, and Test Feedback

Every effort will be made to have assignments and project components marked and handed back within 3 weeks of the handin date, preferably sooner. If we are unable to comply with our intended return dates, revised dates will be posted on the course website.

Academic Accommodation for Medical Illness

If you are unable to meet a course requirement due to illness or other serious circumstances, you must provide valid medical or other supporting documentation to the Academic Counselling office of your home Faculty as soon as possible and contact your instructor immediately. It is the student's responsibility to make alternative arrangements with their instructor once the accommodation has been approved and the instructor has been

informed. For further information please see:

http://www.uwo.ca/univsec/pdf/academic_policies/appeals/accommodation_medical.pdf.

A student requiring academic accommodation due to illness should use the Student Medical Certificate when visiting an off-campus medical facility or an Accommodation Certificate from Student Health Services. The Student Medical Certificate form can be found

here: http://www.uwo.ca/univsec/pdf/academic_policies/appeals/medicalform.pdf.

Test and Exams

Final: 3 hours during the December exam period

As an important note, computer-marked multiple-choice tests and/or exams may be subject to submission for similarity review by software that will check for unusual coincidences in answer patterns that may indicate cheating. Furthermore, there will be no cheat sheets, books, or other reference materials allowed for the exam. No calculators, cell phones, or other electronic devices will be permitted either.

Assignment and Project Components

Due Dates (tentative)

Individual Assignment:	15% (medium)	Assigned September 12, 2017	Due October 3, 2017
Group Project - Stage #1:	10% (light)	Assigned September 19, 2017	Due October 19, 2017
Group Project - Stage #2:	10% (medium)	Assigned October 19, 2017	Due November 7, 2017
Group Project - Final Project Submission:	25% (heavy)	Assigned November 7, 2017	Due November 28, 2017
Group Project - Final Documentation:	5% (light)	Assigned November 7, 2017	Due December 8, 2017

If, for any reason, the project schedule given above cannot be adhered to, the project marks will be pro-rated. (The assignment and project components are worth 65% of the overall mark for the course. If a component has to be cancelled for any reason, the remaining project component weights will be prorated to add up to 65%.)

About the Assignment and Project Components

- Component descriptions will be posted on the course website by the dates listed above.
- Any changes, updates, and clarifications to these descriptions will also be posted on the website. It is your responsibility to monitor these pages closely.

- As mentioned earlier, the assignment and project components will involve design, analysis and implementation of a reasonably large-scale software system, implemented using C++.
- While the project is a group project, grades will be assigned to each student based on both group and individual performance for each component. Individual performance will be based on a number of factors, some of which may include peer evaluations, contributions made during class, repository logs, individual reports of work completed, and so on.

Submission

- All components must be type-written for legibility and to facilitate electronic submission. If components require the creation of diagrams or illustrations, these too must be done electronically. Appropriate tools will be discussed in class and in the descriptions of the components.
- You are required to submit each component electronically through OWL. (If final submissions are too large for OWL submission, alternate arrangements will be made.) Details will be given in the descriptions. We reserve the right to use similarity detection software to detect possible plagiarism cases.
- Components are expected to be individual efforts (where individual could also mean designated group in the case of a group project component). Any code that is borrowed from an existing source or book must be clearly identified as such in the appropriate documentation; otherwise, this may constitute a plagiarism offence.

Late Policy

- Late components will be accepted for up to two days after the due date, with weekends counting as a single day; the late penalty is 20% of the available marks per day. Lateness is based on the time the assignment or project component is submitted.
- Extensions will be granted only by your course instructor. If you have serious medical or compassionate grounds for an extension, you **must** take supporting documentation to the Academic Counselling unit of your faculty, who will contact the instructor.

Marking

- Assignment and project components are marked by the instructor and/or a teaching assistant assigned to the course. We will attempt to include some information about the marking criteria in the appropriate descriptions.
- When marking has been completed, you will be informed via the course website and/or e-mail.
- A request for adjustment in a mark must be made within 2 weeks of the date on which it was first available for pickup. (Beyond that date, regrading will not be considered.) Such a request must be submitted in writing, and must include specific reasons why you believe you deserve more marks. The request must be accompanied by all materials that were originally handed in, as well as the original marker's grade summary sheet. Regrading requests will take a minimum of 24 to 48 hours to process; you will be informed when it is complete.

- Component marks may be posted periodically throughout the term through OWL. It is your responsibility to check that your marks have been recorded correctly.

Backups

It is your responsibility to keep up-to-date backups of all assignment and project files in case of system crashes or inadvertently erased files. Retain copies of all material handed in, as well as the actual graded version, to guard against the possibility of lost components or errors in recording marks. It is not safe to discard these materials until you are satisfied that your final mark for the course has been computed properly.

Ethical Conduct

Scholastic offences are taken seriously and students are directed to read the appropriate policy, specifically, the definition of what constitutes a Scholastic Offence, at the following Web site:

http://www.uwo.ca/univsec/pdf/academic_policies/appeals/scholastic_discipline_undergrad.pdf.

Plagiarism: Students must write their essays and assignments in their own words. Whenever students take an idea, or a passage from another author, they must acknowledge their debt both by using quotation marks where appropriate and by proper referencing such as footnotes or citations. Plagiarism is a major academic offence. Please note, however, that students are not allowed to make use of the work of others unless explicitly instructed to do so in the description of an assignment.

All projects are to be exclusively your own work. While project work requires you to work in teams, each team is expected to act individually. You may discuss approaches to problems among yourselves; however, the actual details of the work (coding, documentation, etc.) must be an individual effort. Incidents that are judged to be the result of academic dishonesty will be reported to the [Undergraduate Chair](#). The selection of penalty to be applied is up to the Chair, with consultation of the instructor.

The standard departmental penalty for assignments that are judged to be the result of academic dishonesty is, for the student's first offence, a mark of zero for the assignment, with an additional penalty equal to the weight of the assignment also being applied. You are responsible for reading and respecting the Department of Computer Science's policy on [Scholastic Offenses](#). and [Rules of Ethical Conduct](#).

The University of Western Ontario uses software for plagiarism checking. Students may be required to submit their written work and programs in electronic form for plagiarism checking.

All required papers may be subject to submission for textual similarity review to the commercial plagiarism detection software under license to the University for detection of plagiarism. All papers submitted for such checking will be included as source documents in the reference database for the purpose of detecting plagiarism of papers subsequently submitted to the system. Use of the service is subject to the licensing agreement, currently between The University of Western Ontario and Turnitin.com (<http://www.turnitin.com/>).

Accessibility Statement

Please contact the course instructor if you require material in an alternate format or if you require any other arrangements to make this course more accessible to you. You may also wish to contact Services for Students with Disabilities (SSD) at 661-2111 x 82147 for any specific question regarding an accommodation.

Support Services

Learning-skills counsellors at the Student Development Centre (<http://www.sdc.uwo.ca>) are ready to help you improve your learning skills. They offer presentations on strategies for improving time management, multiple-choice exam preparation/writing, textbook reading, and more. Individual support is offered throughout the Fall/Winter terms in the drop-in Learning Help Centre, and year-round through individual counseling.

Students who are in emotional/mental distress should refer to Mental Health@Western (http://www.health.uwo.ca/mental_health) for a complete list of options about how to obtain help.

Additional student-run support services are offered by the USC, <http://westernusc.ca/services>.

The website for Registrarial Services is <http://www.registrar.uwo.ca>.

Tutoring

The role of tutoring is to help students understand course material. Tutors should not write assignments, projects, or take-home tests for the students who hire them. Having employed the same tutor as another student is not a legitimate defense against an accusation of collusion, should two students hand in work judged similar beyond the possibility of coincidence.