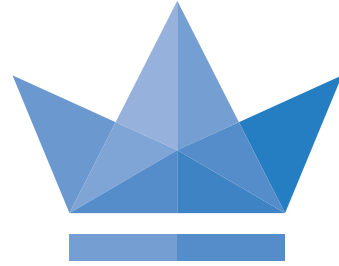


CMPT 320 — 3 (3-0-3)
SOFTWARE ENGINEERING
Fall 2018/19



Lecturer: Dr Andrew Tappenden
andrew.tappenden@kingsu.ca

Office & Hours: A203 *drop in or by appointment*

Class Times:	Lecture	Monday, Wednesday, Friday	N206	10:00 – 10:50
	Lab	Monday ¹	S210	14:00 – 16:50

I. Calendar Description

An experiential introduction to the software development process. Students will work in software development teams to deliver a semester-long software project. Through the completion of the project students are immersed in a modern software engineering environment and are required to develop strong technical and interpersonal skills. Software engineering topics include: the software lifecycle, requirements specification, object-oriented design and analysis, design patterns, verification and maintenance, environments and engineering tools, human-computer interface, social and ethical issues.

II. Objectives

This is a course about the development of real-world software. Software systems are some of the most complex systems that humans build. Consequently, software development projects often turn into expensive failures. Software engineering tackles the size and complexity of real-world software development by adopting a disciplined approach to designing, developing and maintaining the software. CMPT 320 presents a broad overview of the main ideas of software engineering and introduces current software engineering techniques. The course has a strong practical component, including a group project that will flow through the entire course.

By the end of the course, students should:

- Understand the main ideas and issues of software engineering, including the role of software development processes.
- Be able to apply several current software engineering techniques—Traditional & Agile.
- Have gained experience of developing software in teams and acting professionally.
- Be better equipped to contribute to industrial software development projects.

III. Course Conduct

The course is divided into 2 components—3 hours of lectures and a 3 hour laboratory period per week. The lectures will cover the necessary background information and theoretical basis for

¹ Labs will commence on Monday, September 9th.

discussion and assessment. The laboratory component will consist of a hands-on approach to applying the concepts taught during the lectures. The majority of the laboratory time will be for self-guided work on software projects; however, there will be laboratory-only discussions regarding specific assignments if the need arises.

General Expectations:

It is expected that students should behave in an honest and forthright manner in all of their activities. To this end, it is expected that students:

- I. Should be courteous and helpful to one another and discussions about problems and projects is strongly encouraged. However, all of the assignments should be the **sole work** of the individual students unless otherwise directed by the instructor. In all cases, when using code from the textbook, or other sources, including previous course work or discussions with other students, credit as to the source should be clearly indicated. Failure to do so will be considered plagiarism. According to The King's University policy plagiarism must be reported. For the first offence, the assignment will receive an automatic zero. For the second offence, the student will receive an automatic zero for the course.
- II. All assignments should be completed on time. **Late assignments will not be accepted—no exceptions.** Please contact the instructor **prior to the deadline** if for some unforeseen reason you expect that you will not be able to complete the assignment on time.
- III. Students are responsible for conducting themselves in accordance with the University regulations. Specific attention should be paid to the Scholarly Ethics and Academic Honesty policy, as outlined within the Academic Calendar.

IV. Text(s) and Other Resources

Required Text: None
Course website: <http://cs.kingsu.ca/~andrew/cmpt320>
username: cmpt320 password: _____

V. Grade Distribution

Individual Assignments	15%
Group Project	40%
Midterm Examination	15%
Final Examination	30%
Total	100%

Description of Assignments

A. Individual Assignments (15%)

All individual assignments will be handed-out during the laboratory periods and must be completed individually. Assignments will include three in-lab assignments, one individual project spanning three lab periods. Assignments are to be submitted by the due date specified—*late work will NOT be accepted*. Any/all laboratory components of assignments will be assessed within the Linux laboratory environment. **In all cases submissions that fail to compile and/or run within the Linux laboratory environment will receive a mark of zero for the programming component of the assignment.** If students have any concerns regarding the mark assigned, they must see the instructor within one week.

B. Group Project (40%)

Upon completion of the individual lab assignments, a group project will begin in earnest. The group project will be accomplished in groups of three or four students, and will represent the development of a complex GUI application. The group project will be undertaken using the remainder of the laboratory periods, and will require work both inside and outside of scheduled laboratory times. The project will be assessed within the Linux laboratory environment. **In all cases submissions that fail to compile and/or run within the Linux laboratory environment will receive a mark of zero for the programming component of the project.**

The project is broken into several smaller deliverables: Requirements & Design (10%); Application Prototype (10%); Final Deliverable (10%); and an Individual Performance Assessment (5%). As with a real-world project, these are *hard* deadlines, and late work *will not be accepted*. In order to be successful in this course, your group must remain on track and stay focused throughout the semester. **This is no small task—do not let your project slip!**

C. Midterm Examination (15%)

The midterm examination will take place on **Friday, October 25th** and will be held in place of the regularly occurring lecture.

D. Final Examination (30%) (During Exam Week)

The final examination will be administered during the regularly scheduled Exam Week and will be cumulative and based on the text, readings, hand-outs, projects, class and lab activities. The date and time of the exam is to be announced.

VI. Topical Outline for the Course

Why is Software Engineering so Difficult?

Requirements

- What is a Requirement?
- Requirement Gathering
- Use Cases
- Logical/Formal Descriptions
- Requirements within Agile Methods

Design

- Test Driven Development
- Design Patterns

Implementation

- Configuration Management
- Coupling & Cohesion
- Refactoring
- Debugging

Testing

- Functional Testing
- Structural Testing
- Other forms of Testing
- Inspection
- Static Code Analysis

Maintenance (*Time permitting*)

- Legacy Systems
- Reverse Engineering