

Course Syllabus

CP317: Software Engineering

Physics & Computer Science, Waterloo Campus
Spring 2019

2018-10-12 05:10

Professor Information

Dr. Abdul-Rahman Mawlood-Yunis | Science Building, N3022F
519-884-1970 x4091 | [amawloodyunis AT wlu.ca](mailto:amawloodyunis@wlu.ca)
By Appointment

Course Information

Discussion of software development activities, including software process models, analysis, design, implementation, testing, project management and advanced topics. Both traditional and object-oriented methods are considered.

3 lecture hours

Credit: 0.50

Prerequisite: CP213, CP217 or CP264

Exclusion: CP417

Outline: [course syllabus](#)

URL: <https://mylearningspace.wlu.ca/d2l/home/302073>

Section Information

Section	Days	Times	Room	Professor
Lecture A	TR	1:00 pm - 2:20 pm	Dr. Alvin Woods Building 2-106	Abdul-Rahman(AR), Mawlood-Yunis

Course Overview and Approach

To learn software engineering concepts by completing a major group project, including:

- software requirements and specifications
- object-oriented analysis
- object-oriented design
- project management
- Unified Modeling Language (UML)
- design patterns

- advanced programming in Java and/or other appropriate languages

Course Tools and Learning Materials

You are responsible for storing all your project work on your own memory sticks, Laurier network storage, or other media. We strongly suggest that you keep at least two copies of your work at all times, for memory sticks can fail, or be lost.

[MyLearningSpace / Brightspace](#)

Laurier [Library](#)

[Centre for Student Success](#) (writing centre, math centre, academic advising, study skills/supplemental instruction, accessible learning)

Student Evaluation

Assessment	Weighting
Tests (2 @ 15% each):	30%
Project:	70%

There are two tests of 1 hour, each to be done at the beginning of a lecture. The tests cover the theoretical material of the course and are based upon the text readings of the preceding weeks as well as lecture and project material.

Tests

Test	Written
Test 1	Tuesday, June 18
Test 2	Tuesday, July 30

Project

Projects will be done in teams of 5 members (or more). Each team should come up with the project they would like to work on. The instructor will assign projects where students could not decide on the project they work on.

Weekly Schedule

Note: this schedule is subject to change.

May 7 and 9 (Week 1)

- Tue, May 07 -- Course Introduction, Working in Teams, Projects

- Chapter 1 -- [Software and Software Engineering](#) (Slides 1-26)
 - Additional readings: [Some disasters attributable to bad numerical computing](#)
 - Thu, May 09
 - Team Assignments
 - Chapter 1 -- [Software and Software Engineering](#) (Slides 36-57)
 - Additional readings: [The pragmatic programmer \(chapters 0 and 1\)](#)
 - A good link to regularly test your knowledge of software engineering . [software engineering flashcards](#)
-

May 14 and 16 (Week 2)

- Tue, May 14
 - Consultation Session. Help teams determine if their projects are feasible and sized appropriately for one semester (No marking)
 - [Chapter 2 Process Model and Software Life Cycle](#) (slides ...)
 - Thu, May 16 -- **Project Charter due on the next class**
 - Problem Statement: Short and succinct (one or two sentences)
 - Project Objectives: What the project will achieve
 - Stakeholders: Persons who will be actively involved with the project (e.g. project sponsor, types of users, etc.)
 - Project Deliverables: The major results or services that will be produced, what are the specific things the software will do.
 - Project Charter Grading Rubric is [here](#)
 - [Chapter 2 Process Model and Software Life Cycle](#) (slides ...)
 - [Chapter 22, Software Engineering: Teamwork](#) (slides 38 to 51)
 - Additional Reading [The Programmer's Guide to Working on a Team](#)
-

May 21 and 23 (Week 3)

- Tue, May 21, **Project Charter due today**
 - [Software Engineering, Chapter 3, Agile software development](#) (Slides ...)
 - Thu, May 23 -- Last Day to Discuss Project Charter
 - Code Repository Setup is due
 - Project Name due (Choose a name for your project)
 - [Software Engineering, Chapter 3, Agile software development](#) (Slides ...)
 - [Software Engineering, Chapter 25 Version Control](#) (slides 8 to 31)
 - Additional references:
 - [SCRUM Software Design Framework \(slides\)](#)
 - [Scrum Primer , by Deemer/Benefield/Larman/Vodde \(slides\)](#) (slides)
 - [Scrum Primer , by Deemer/Benefield/Larman/Vodde](#) (PDF)
-

May 28 and 30 (Week 4)

- Tue, May 28
 - Software Engineering, Chapter 4, [Developing Requirements](#) (slides)

- Functional and non-functional requirements
 - Requirements engineering processes
 - Requirements elicitation
 - Thu, May 30
 - Product Backlog (Requirements Document) Grading Rubric is [here](#)
 - [Software Engineering, Chapter 4, Developing Requirements](#) (slides ...)
 - Requirements specification
 - Requirements validation
 - Requirements change
 - Additional readings:
 - [Requirements \(CP317: - Fall 2018\)](#)
 - [IEEE Recommended Practice for Software Requirements Specification](#)
-

June 4 and 6 (Week 5)

- Tue, June 4 -- Last Day to Discuss Product Backlog (Requirements Document)
 - Software Engineering, Chapter 5, [System modeling](#) (slides)
 - Context models
 - Interaction models
 - Thu, June 6 -- Product Backlog (Requirements Document) is due
 - Software Engineering, Chapter 5, [System modeling](#) (slides)
 - Structural models
 - Behavioral models
 - Model-driven engineering
-

June 11 and 13 (Week 6)

- Tue, June 11
 - [Software Engineering, Chapter 6, Architectural design](#) (slide)
 - Architectural design decisions
 - Architectural views
 - Thu, June 13
 - [Software Engineering, Chapter 6, Architectural design](#) (slide)
 - Architectural patterns
 - Application architectures
-

June 18 and 20 (Week 7)

- Tue, June 18
 - Test 1
 - Thu, June 20
 - [Software Engineering, Chapter 7, Design and Implementation](#)
 - Object-oriented design using the UML
 - Design patterns
 - Design Document Grading Rubric is [here](#).
-

June 25 and 27 (Week 8)

- [Tue, June 25](#)
 - [Software Engineering, Chapter 7, Design and Implementation](#)

- Design patterns
 - Software Reuse
 - Thu, June 27 **Design Document is due today**
 - [Software Engineering, Chapter 8, Software Testing](#)
 - Development testing
 - Test-driven development
 - Release testing
 - User testing
-

July 2 and 4 (Week 9)

- Tue, July 2
 - [Testing](#)
 - Release testing
 - User testing
 - Thu, July 4
 - [Final Project Presentation Rubric](#)
 - [Software Engineering, Chapter 22, Project management](#) (slide)
 - Software pricing
 - Plan-driven development
 - Project scheduling
 - Estimation techniques
-

July 9 and 11 (Week 10)

- Tue, July 9
 - [Final Project Presentation Rubric](#)
 - [Software Engineering, Chapter 23, Project planning](#) (slide)
 - Software pricing
 - Plan-driven development
 - Project scheduling
 - Estimation techniques
 - Thu, July 11
 - Final Project Presentation
-

July 16 and 18 (Week 11)

- Final Project Presentation
-

July 23 and 25 (Week 12)

- Final Project Presentation
-

July 30 (Week 13)

- Tue July 30
 - Test 2
-

Book chapters are from

- Software Engineering, Ian SommerVille

- Object-Oriented Software Engineering Practical Software Development using UML and Java
- Object Oriented Software Engineering Using UML, Patterns and Java
- Object-oriented and classical software engineering , 8th ed.
- UML Distilled
- Object-Oriented Design Heuristics -- free pdf available on Internet
- <https://link-springer-com.libproxy.wlu.ca/article/10.1007/s12176-018-0020-3>
- <https://www.shopify.ca/partners/blog/rapid-prototyping>

University and Course Policies

1. **Academic Calendars:** Students are encouraged to review the [Academic Calendar](#) for information regarding all important dates, deadlines, and services available on campus.
2. **Special Needs:** Students with disabilities or special needs are advised to contact Laurier's [Accessible Learning Centre](#) for information regarding its services and resources.
3. **Plagiarism:** The University has approved the following wording for inclusion on all course syllabi about the use of the institutionally supported plagiarism software tool. "Wilfrid Laurier University uses software that can check for plagiarism. If requested to do so by the instructor, students are required to submit their written work in electronic form and have it checked for plagiarism." (Approved by Senate May 14, 2002) .
4. **Academic Integrity:** Laurier is committed to a culture of integrity within and beyond the classroom. This culture values trustworthiness (i.e., honesty, integrity, reliability), fairness, caring, respect, responsibility and citizenship. Together, we have a shared responsibility to uphold this culture in our academic and nonacademic behaviour. The University has a defined policy with respect to academic misconduct. As a Laurier student you are responsible for familiarizing yourself with this policy and the accompanying penalty guidelines, some of which may appear on your transcript if there is a finding of misconduct. The relevant policy can be found at Laurier's [academic integrity](#) website along with resources to educate and support you in upholding a culture of integrity. Ignorance is not a defense.
5. **Final Examinations:** Students are strongly urged not to make any commitments (i.e., vacation) during the examination period. Students are required to be available for examinations during the examination periods of all terms in which they register.
6. **Foot Patrol, the Wellness Centre, and the Student Food Bank:**
 - **[Waterloo Student Food Bank](#):** All students are eligible to use this service to ensure they're eating healthy when overwhelmed, stressed or financially strained. Anonymously request a package online 24-7. All dietary restrictions accommodated.
 - **[Waterloo Foot Patrol](#):** 519.886.FOOT (3668). A volunteer operated safe-walk program, available Fall and Winter daily from 6:30 pm to 3 am. Teams of two are assigned to escort students to and from campus by foot or by van.
 - **[Waterloo Student Wellness Centre](#):** 519-884-0710, x3146. The Centre supports the physical, emotional, and mental health needs of students. Located on the 2nd floor of the

Student Services Building, booked and same-day appointments are available Mondays and Wednesdays from 8:30 am to 7:30 pm, and Tuesdays, Thursdays and Fridays from 8:30 am to 4:15 pm. Contact the Centre at x3146, wellness@wlu.ca or @LaurierWellness. After hours crisis support available 24/7. Call 1-844-437-3247 (HERE247).

- [Good2Talk](#) is a postsecondary school helpline that provides free, professional and confidential counselling support for students in Ontario. Call 1-866-925-5454 or through 2-1-1. Available 24-7.

Resources

Java IDE

Eclipse is an open extensible IDE that can be used for many languages - we shall use it for Java development. It is available on our lab machines, and you may download Eclipse Neon for home use from <http://www.eclipse.org/>. (*Eclipse* is a Java program and should run on most platforms).

UML IDE

MagicDraw 17 is a commercial product for producing UML diagrams. It is available in the department labs, but is *not* available for free download. It has hooks for use with *Eclipse* so that you can perform round-trip generation of Java code from UML diagrams and reverse engineering. For more information see <http://www.magicdraw.com/>

The course instructor will offer labs and tutorials in these products as necessary.

The educational materials developed for this course, including, but not limited to, lecture notes and slides, handout materials, examinations and assignments, and any materials posted to MyLearningSpace, are the intellectual property of the course instructor. These materials have been developed for student use only and they are not intended for wider dissemination and/or communication outside of a given course. Posting or providing unauthorized audio, video, or textual material of lecture content to third-party websites violates an instructor's intellectual property rights, and the Canadian Copyright Act. Recording lectures in any way is prohibited in this course unless specific permission has been granted by the instructor. Failure to follow these instructions may be in contravention of the university's Code of Student Conduct and/or Code of Academic Conduct, and will result in appropriate penalties. Participation in this course constitutes an agreement by all parties to abide by the relevant University Policies, and to respect the intellectual property of others during and after their association with Wilfrid Laurier University.

© 2018 David Brown. All Rights Reserved.

Last updated: 2018-10-12 05:10