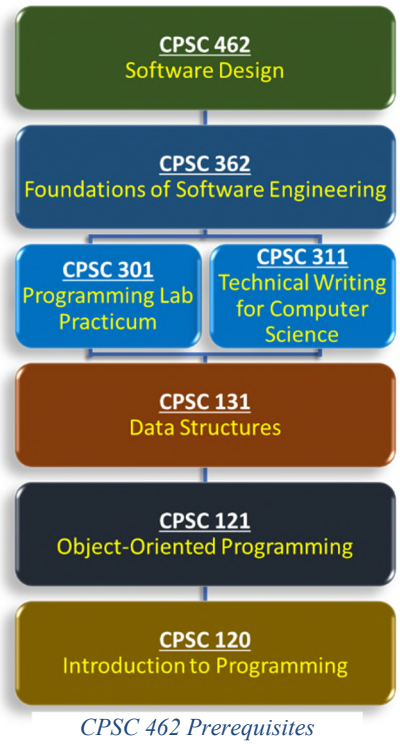
Syllabus
CPSC 462 - Software Design
Fall 2018

Description & Objectives

Concepts of software modeling, software process, and some tools. Object-oriented analysis (OOA) and design (OOD) and Unified Process (UP). Some computer-aided software engineering (CASE) tools will be recommended to use for doing homework assignments.



CPSC 462 Prerequisites

Prerequisites

CPSC 362 - Foundations of Software Engineering

Instructor

Professor Thomas L Bettens
Phone: 657-278-4999
eMail: TBettens@fullerton.edu
Office: CS 401

Office Hours:
- Monday & Wednesday: 2:00 - 2:45*
- Tuesday & Thursday: 1:00 - 2:00*
- By appointment, coordinated at least 24 hours in advance
- During final exam week, office hours are by appointment only

* *arrive at least 15 minutes before end of scheduled Office Hours*

Meeting Information

CPSC 462-01/02		
Room	**Day**	**Time**
CS 408	Monday	5:00 PM – 6:50 PM
	Wednesday	

CPSC 462-03/04		
Room	**Day**	**Time**
CS 101	Monday	3:00 PM – 4:50 PM
	Wednesday	

Important Dates

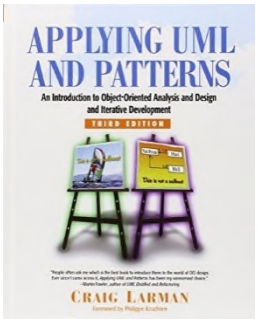
CSUF's Academic Calendar is posted online at «http://apps.fullerton.edu/AcademicCalendar/». The Academic Calendar contains all the campus closures and holidays you should be aware of.

CSUF's Admissions Calendar is posted online at «http://www.fullerton.edu/admissions/Resources/Calendars.asp». The Admissions Calendar contains all the major dates with respect to adding, dropping, and withdrawing from your classes.

Monday, August 27	First day of class
Monday, September 3	Labor Day - CAMPUS CLOSED
Monday, November 12	Veterans Day - CAMPUS CLOSED
Wednesday, November 21	Fall Recess - NO CLASSES
Wednesday, December 12	Last day of class
Monday, December 17 @ 5:00PM - 6:50PM	Section 01/02 Final Exam
-- OR --	
Wednesday, December 19 @ 2:30PM - 4:20PM	Section 03/04 Final Exam

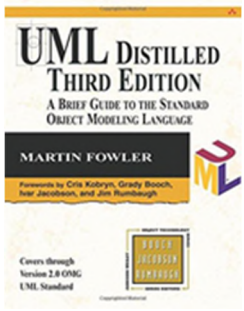
Textbooks

Required



- *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development* 3rd Ed
  Craig Larman; Prentice Hall PTR
  ISBN-10: 0-13-148906-2; ISBN-13: 978-0-13-148906-6

Optional



- *UML Distilled : A Brief Guide to the Standard Object Modeling Language* 3rd Ed
  Martin Fowler; Addison Wesley Professional
  ISBN-10: 0-321-19368-7; ISBN-13: 978-0-321-19368-1

Many popular technical books may be read online through the campus's subscription to Safari Books Online. From outside of the campus network, the campus library's WWW proxy will grant you access, «http://www.library.fullerton.edu/asp/ipcheck.aspx?url=http://proquest.safaribooksonline.com/?uicode=calstate». The Safari Books Online service can be accessed directly from any computer on the campus network, «http://proquest.safaribooksonline.com/».

The librarians at the CSUF Pollack Library have developed a number of research guides to support students studying computer science. The LibGuides are online at «http://libraryguides.fullerton.edu/sb.php?subject_id=17830».

Course Outline

Week	Date	Topic	**Textbook Chapter** Larman's *Applying UML and Patterns*	Notes
		Part 1: Overview		
1	8/27 8/29	Introduction • Introduction of the course / class • Class project overview • Software design & development environment (part 1)		
2	~~9/3~~	*Labor Day - No Class*	Chapters 1-3	*Team enrollment due before next class*
	9/5	Software Process and Methodology • OOA/OOD with Unified Process tour • Software design & development environment (part 2)		
		Part 2: Inception Phase -- Is it feasible and cost effective?		
3	9/10 9/12	Evolutionary requirements & artifacts • System Level Use Cases • Use Case & Activity Diagrams • Project candidates	Chapters 4-7, 28, 30	*Project selection due before next class*
4	9/17 9/19	Project definition • Use Case Selection • Detailed use case definitions		
5	9/24 9/26	Feasibility decision point • Go NoGo (drop or continue?) • Time estimates • Risks		*Inception phase artifact delivery due before next class (see Table 7.1 page 118) plus Risk Registry*
		Part 3: Elaboration Phase Iteration 1 -- Risk reduction, framework, and key use case implementation		
6	10/1 10/3	OO Analysis: Static & Dynamic Views • Domain modeling & models • System Sequence diagrams	Chapters 8-9 Chapter 10	
7	10/8 10/10	OO Design: Static & Dynamic Views I • UML Interaction diagrams • UML Class diagrams	Chapters 14-15 Chapter 16	
8	10/15 10/17	OO Design: Static & Dynamic Views II • UML Interaction diagrams • UML Class diagrams		
9	10/22 10/24	General responsibility assignment software principles (GRASP)	Chapters 17-18	
10	10/29 10/31	• Designing for visibility • Mapping designs to code	Chapters 19-20	• *Elaboration Phase Iteration 1 artifact delivery due before next class (see Table 18.1 page 362)* • *In-class working demo (informal) due*

Week	Date	Topic	**Textbook Chapter** Larman's *Applying UML and Patterns*	Notes
Part 4: Elaboration Phase Iteration 2 -- Additional customer driven use case implementation & risk reduction				
11	11/5 11/7	Assessment and planning • What worked well? What didn't? • Iteration 2 Project Goals • Iteration 2 Use Case selection		
12	~~11/12~~	*Veterans Day - No Class*	Chapters 23-25	
	11/14	• More Objects with Responsibilities (GRASP)		
13	11/19	Pattern identification and application • Applying Patterns • GoF Patterns	Chapter 26, Section 36.7 (abstract factory)	*Yes, class will be held on Monday*
	~~11/21~~	*Fall Recess (Thanksgiving Break) - No Class*		
14	11/26 11/28	Other Design Considerations • State Machine diagrams • Deployment diagrams • Component diagrams	Chapters 29, 38	
Team Walkthrough & Presentation -- Communication with both technical and management stakeholders				
15	12/3	Final touches & preparation day		
	12/5	Project Presentation		• *Elaboration Phase Iteration 2 artifact delivery due (see Table 18.1 page 362)* • *In-class artifact walkthrough* • *In-class working demo (formal) due*
16	12/10 12/12			
Final Exam Week -- Show off what you have learned!				
17	12/17	Section 01/02 Final Exam @ 5:00PM - 6:50PM		
	12/19	Section 03/04 Final Exam @ 2:30PM - 4:20PM		

Software Needs

The following are some of the software packages you may need for your group project:

- C++ development environment
- Visual Paradigm Community free UML software tool (http://www.visual-paradigm.com/solution/freeumldesigntool/)
- Microsoft Office (e.g. Word, Excel, PowerPoint, Visio) or equivalent
- Adobe Acrobat Reader or equivalent
- Web browser
- Optional - Video capturing tool (Snagit, Game DVR {free}, recordMyDesktop (free}, others …)

See Appendix A – Development Tool Resources for additional information

Learning Goals

This course focuses on software design techniques used in the development of large and complex software systems. Learning goals encompass advancing the student's knowledge, proficiency, and demonstratable ability to skillfully recognize and apply:

1. Software object responsibility assignment using the nine fundamental GRASP principles.
2. Phases of the Unified Process software development life cycle process as well as the purpose, products and tools.
3. Architecture, modeling (including UML) and processes for carrying out analysis and design.
4. Balancing an emphasis on design principles with an understanding of how to apply techniques and methods to create successful software systems.
5. Identify, apply, create, and leverage appropriate UML artifacts to communicate architecture, design, and implementation strategies.

G.E. Requirements

This class does not meet any CSU General Education requirements.

Technical Proficiency

Technical proficiency in programming and software engineering should correspond to the prerequisite(s) of the course. Students are expected to be intimately familiar with their development platform of choice and be able to write and debug code in C++ at a level of proficiency that corresponds to the prerequisites of the course.

Technical proficiency with information technology, such as, but not limited to, the use of web-based online services, sending and receiving electronic mail, and desktop computer file systems, is assumed.

Grading

Plus and minus grading is used when determining final grades.

Final grades are computed by first finding the average score in each category described in the second table below. All scores are normalized to a scale of 0 to 100 before being averaged. The average score for each category is then used to compute the weighted average according to the weights in the second table below.

Grade	% of Total Points
A	93% ≤ X
A-	90% ≤ X < 93%
B+	87% ≤ X < 90%
B	83% ≤ X < 87%
B-	80% ≤ X < 83%
C+	77% ≤ X < 80%
C	73% ≤ X < 77%
C-	70% ≤ X < 73%
D+	67% ≤ X < 70%
D	60% ≤ X < 67%
F	X < 60%

Category	% of Final Grade
Participation & attendance	10%
Checkpoints & quizzes	10%
Inception Phase • *Artifacts*	10%
Elaboration Phase Iteration 1 • *Artifacts* • *Demonstration*	10% 5%
Presentation • *Project Walkthrough** • *Artifacts** • *Demonstration** **Grad student "extras" will be factored into scores*	15% 20% 10%
Final Exam	10%

Graduate Grading

Graduate students that use this course on a graduate study plan must perform additional work and will be evaluated on a separate grading scale vis-à-vis their undergraduate counterparts.

An additional programming project is mandatory for all graduate students. The project is proposed by the student and approved by the instructor. The iterative software development nature of this course suggests a logical approach is to identify and implement an additional Use Case or Use Case Scenario. It is your responsibility to identify this "extra" Use Case, have it approved by the instructor during the last increment, and include this as part of your final project.

Plus and minus grading is used for graduate students in the same manner it is used for undergraduate students. See above for details.

Assignments

Programming and written assignments will be discussed in class and posted to the course website in advance of their due dates. Reading assignments are outlined in the syllabus and it is the responsibility of the student to stay up to date with the reading.

Written assignments must be typeset and presented in a professional manner. Presentation, spelling and grammar can be worth up to 30% of a written assignment's grade.

All programming assignments must be written in the C++14 programming language, unless specified otherwise. Coding style must conform to professional norms. At a minimum, code must be commented, have descriptive names for identifiers, and contain a comment at the top of each file with pertinent information such as the student's name, email address, and assignment name. A plain text README.TXT must be included with each assignment submission summarizing and documenting the work submitted. For students unfamiliar with coding style, Google's style guides are an excellent starting point, «https://github.com/google/styleguide», particularly their C++ style guide, «https://google.github.io/styleguide/cppguide.html».

Assignments are to be submitted via the course website *as a single .zip* (or equivalent, i.e. .7z, .rar, .tar, .tar.gz) file, preserving directory hierarchical structure. Please *do not include your executable* (*.exe), object (*.obj), or any other non-relevant files (e.g. project files, precompiled header data, debug data).

At the start of the semester, the instructor will detail the platform and tools used to grade student assignments. It is the student's responsibility to ensure that the assignments execute to his or her satisfaction on the instructor's grading platform.

There are approximately:

- 3 deliveries of working code and supporting documentation
- Each delivery builds upon the last delivery
- No midterm exam
- Several online knowledge check, and checkpoint quizzes throughout the course
- 1 final exam
- 13 weeks of reading assignments

Assignments are to be submitted through our course website on or before the posted due date and time. Late assignments are not accepted. This policy is enforced by our website. Under no circumstances are assignments accepted through eMail.

Exceptions are made on a case by case basis given enough time and evidence to weigh the merits of the application.

Attendance Policy

90% class attendance is mandatory. There are roughly 30 class meetings. This means you may choose to be absent three times (that is, choose to not attend 3 class meetings) during the semester without needing to notify the instructor or justify the absence. This is intended to accommodate unforeseen issues.

Attendance will be tracked via the iFullerton app and also via your participation in class or lab activities. If you logged your attendance on iFullerton but did not participate in a class activity, for example, then you will still be considered absent. You are expected to log your attendance for each class meeting. In case you are unable to log your attendance in the iFullerton app (a rare event), please let the instructor know in writing during class so your attendance can be manually entered later. Be sure you (re)familiarize yourself with CSUF's Course Attendance feature as part of our iFullerton app and TITANium Engagement tool at «http://www.fullerton.edu/it/services/atc/Course_Attendance.php» and watch the Course Attendance Video Instructions. Smartphone checkout for Course Attendance check-in is available to students upon faculty request. For more information, please visit the Student Genius Corner Equipment Checkout website.

Allowed absences will not result in grade deductions for participation, but quizzes and lab exercises may be affected depending on your prior attendance and scores. You will be responsible for catching up on what you missed from class and I will not be obligated to give make-up lectures or activities for that day. Absences outside of the 3-day allowance will affect your grade. You will not receive the corresponding participation, quiz, or lab exercise grade for the day of the unexcused absence.

Missing class as part of a documented accommodation is guaranteed to be excused. The ADA accommodated student must make a reasonable effort to coordinate any absences with the instructor.

Make Up Policy

Exams and quizzes cannot be taken after they have been given in class. Due to an act of nature, personal medical emergency, a family crisis, an act of terrorism, severe civil unrest, etc. students have 10 calendar days to petition the instructor to retake any exam/quiz or submit an assignment without late penalty.

Exceptions shall be made on a case by case basis, provided there is time to evaluate the merits of such an application.

Participation

In the context of this course, participation is defined as the following:

- Arriving to class prepared and on time.
- Taking notes.
- Actively listening to the lecture and asking questions when appropriate.
- Annotating code listings and handouts.
- Bringing any required materials to class.
- When needed/desired, seeking assistance to complete assignments.
- Barring an emergency, not leaving the class session early unless the instructor consents.
- Not distracting oneself or others with smartphones, games, online diversions, etc.
- Respecting and treating the instructor and the student's peers civilly.

Required Material

- Textbook
- A writing instrument
- A personal computer with the requisite development tools and internet access, or regular access to a computer lab
- iFullerton app on your mobile device to record your daily attendance.

Group Project:

Students are required to participate in the group project. A detailed description of the group project will be available through the course website. Assignments are collaboratively worked and presented as a group while artifacts are submitted individually.

Academic Dishonesty

Students are encouraged to assist one another and discuss the course materials with your peers. It is your responsibility to be aware of and follow the spirit of CSU Fullerton's academic honesty policy which can be found at «http://www.fullerton.edu/senate/publications_policies_resolutions/ups/UPS%20300/UPS%20300.021.pdf». Academic dishonesty will not be tolerated. The University Catalog and the Class Schedule provide a detailed description of Academic Dishonesty under *University Regulations*.

By submitting work for evaluation, you acknowledge that you have adhered to the spirit of the university's academic honesty policy and that your submission is an original work by you unless otherwise directed to work in groups. Failure to follow the spirit of the academic honesty policy will result in a severely negative evaluation of the work in question and may result in involving the Department Chair and the Judicial Affairs office to seek a disciplinary remedy.

Students with Special Needs

Please inform the instructor during the first week of classes about any disability or special needs that you may have that may require specific arrangements related to attending class sessions, carrying out class assignments, or writing papers or examinations. According to California State University policy, students with disabilities must document their disabilities at the Disability Support Services (DSS) Office in order to be accommodated in their courses. Additional information can be found at the DSS website, by calling 657-278-3112 or email «dsservices@fullerton.edu».

Student Resources

Any student who wishes to discuss any concern may contact the assistant deans of the college. Assistant deans are student advocates who will help you navigate the university's policies and procedures and assist with resolving any conflict.

Assistant Dean for Student Affairs: Carlos Santana, CS-206A (657) 278-4407 «csantana@fullerton.edu»

Assistant Dean International Programs and Global Engagement: Lillybeth Sasis, CS-206A (657) 278-4881 «lsasis@fullerton.edu»

Emergency Procedures

For your own safety and the safety of others, each student is expected to read and understand the guidelines published at «http://prepare.fullerton.edu/campuspreparedness/». Should an emergency occur, follow the instructions given to you by faculty, staff, and public safety officials. An emergency information recording is available by calling the Campus Operation and Emergency Closure line at 657-278-4444.

Instructional Continuity

Due to an event such as an epidemic or a natural disaster that disrupts normal campus operations, students must monitor the course Titanium site and their campus email address for any instructions and assignments that the instructor announces.

Laboratory Safety

Safety is no accident. Learning and following the appropriate safety practices and protocols is an integral part to all laboratory courses. Following the appropriate safety practices and protocols minimizes the chances of repetitive stress injuries, mishandling hazardous materials, and injury to self and others. Additional campus laboratory safety information regarding hazardous materials is online at «http://riskmanagement.fullerton.edu/laboratorysafety/».

Extra Credit

There are no planned opportunities for extra credit. However, if opportunity arises, it will be either discussed in class, posted to the course website, or both in advance of its due date.

Recording & Transcription of Class Content

Recording class content is governed by UPS 330.230, «http://www.fullerton.edu/senate/publications_policies_resolutions/ups/UPS%20300/UPS%20330.230.pdf». In summary, unless otherwise mandated, recordings of all kinds are strictly forbidden including but not limited to photographs, video recordings, audio recordings, scanning, and screen capture.

Note however, the instructor must permit class content to be recorded or transcribed by students when mandated to do so by the Americans with Disabilities Act or by other federal or state laws. See *Students with Special Needs* above for guidance navigating the university's policies and procedures. In any event, any recording of class content is for private use and study and shall not be made publicly accessible, including sharing with other students, without the written consent of the instructor and students in the class.

Course Rules & Classroom Management

Unless an agreement or accommodation is reached between the student and the instructor, these rules must be followed.

- Attendance at all regularly scheduled lecture and discussion section is mandatory.
- Do not eat during lecture.
- If it makes noise, silence it.
- Computer use is not allowed in lecture except for taking notes.
- The student is responsible to be aware of any course announcements including changes to due dates and requirements.
- Homework, programming assignments, etc. may not be submitted late.
- Third party work (code, artwork, etc.) may not be used in student work without prior instructor consent. Failure to gain and document instructor consent will be construed as willful academic dishonesty.
- When a third party's work is incorporated into student work after gaining instructor consent, failure to wholly document the work's origin, copyright and license will be construed as willful academic dishonesty.

Appendix A – Development Tool Resources

Students enrolled in CSPC 120, CPSC 121, and CPSC 131 are recommended to use the Computer Science Department's official GNU/Linux development environment, Tuffix. Tuffix is Tuffy the Titan's Linux distribution.

Instructions on how to install Tuffix or a Tuffix based VM are online at «http://csufcs.com/tuffixinstall». The Tuffix Titanium Community for Students, https://communities.fullerton.edu/course/view.php?id=1547 is the best venue to receive help with Tuffix.

Students may self-enroll in the community; first login to your portal, then navigate to *Titanium communities*, next under the dashboard to the left – click *Site home*, then click *Search Courses* on the right, search for 'Tuffix', in the results click *Tuffix Students*, under the gear in the upper right select *Enrol me in this course*, finally click on the button *Enrol me*. You may unenroll at any time.

The Tuffix home page is «https://github.com/kevinwortman/tuffix». Instructors are encouraged to join the Tuffix Titanium Community for Instructors. Please contact Kevin Wortman for access.

Students interested in using Microsoft® development tools may request a Dreamspark account at «http://dsreqform.ecs.fullerton.edu/». A student may, at no monetary cost, download full featured versions of Microsoft Visual Studio.

Students interested in using Apple® development tools can freely download Xcode through the App Store application bundled with macOS. Students may download Xcode directly from «https://developer.apple.com/xcode/».

A CentOS (Linux) shell server is available through secure shell (ssh) and secure file transfer protocol (sftp). The hostname is ecs.fullerton.edu. If your email address is taylor@csu.fullerton.edu, then your username is taylor. If you are using a command-line ssh client, then your command to connect to ecs.fullerton.edu will be `ssh taylor@ecs.fullerton.edu`. Your password is the same password as your CSUF Portal password. (Faculty members can login in as well if you specify your Active Directory domain. For example, if your email address is kai@fullerton.edu, then your username is 'AD\kai; thus the command will be `ssh 'AD\kai@ecs.fullerton.edu`.)

Please consider adopting a package management system for your personal computer to facilitate adding, updating and removing the various software development tools you may wish to use.

- Apple OS X
  - MacPorts «http://www.macports.org/» (recommended)
  - Fink «http://www.finkproject.org/»
  - Homebrew «http://brew.sh/»
- Microsoft Windows
  - Chocolatey NuGet «https://chocolatey.org/»
  - Microsoft OneGet «https://github.com/OneGet/oneget»
  - Cygwin «http://www.cygwin.com/»
  - Npackd «https://npackd.appspot.com/»
- GNU/Linux OS
  - dpkg «https://www.debian.org/doc/manuals/debian-faq/ch-pkgtools.en.html»
  - rpm «http://fedoranews.org/alex/tutorial/rpm/»
  - yum «http://yum.baseurl.org/»