

CptS 322 — Software Engineering Principles I

Washington State University, Tri-Cities

Spring 2005

- Instructor:** A. David McKinnon, Ph.D.
mckinnon@tricity.wsu.edu
- Schedule:** Mon/Wed 17:45–19:00, TEST 203
- Office Hours:** By appointment, typically either before or after class
- Website:** <http://www.tricity.wsu.edu/~mckinnon/cpts322/>
- Prerequisites:** Math 216, CptS 224, c// Engl 402.
Note that CptS 322 is offered as a 'writing in the major' [M] course. It is required that student be enrolled concurrently in English 402 [W]
- Objectives:** Introduction to software systems development with emphasis on requirements analysis, specification and design (preliminary and detailed design reviews). Students will participate in a course project to give them hands-on experience with software engineering principles. There will be documentation (including the appropriate artifacts) required as exit and entrance criteria to each phase of the process. The objective is to experience a particular method for the systematic development of large software systems from reusable fragments (when possible in the classroom setting) that are an essential part of software engineering. (see <http://www.eecs.wsu.edu/Undergraduate/syllabi/cs/cpts322.htm>).
- Required Text:** Software Engineering: An Object-Oriented Perspective, by Eric J. Braude, Wiley, 2001.
- Other Texts:** Software Engineering Sixth Edition, by Sommerville, Ian, Addison-Wesley, 2000.
Software Engineering: A Practitioner's Approach, Fifth Ed., by Pressman, R., McGraw-Hill, 2000.
The Mythical Man-Month Anniversary Edition, by Frederick P. Brooks, Jr., Addison Wiley, 1995.
- Policies:** *Academic Integrity:* Neither plagiarism nor cheating will be tolerated in this class. Established university policies will be followed for all cases of plagiarism/cheating. (The EECS policy is located at: <http://www.eecs.wsu.edu/~syllabus/eeungrad/academic-integrity.html>). Academic dishonesty will result in a failing grade for the assignment, project, or exam in question.
- Code of Conduct:* The ACM Code of Ethics and Professional Conduct and the IEEE Code of Ethics will serve as conduct guides for this class. In particular, all members of the class will be treated with respect and courtesy.
- Disability Accommodations:* Reasonable accommodations are available for students who have a documented disability. Please notify the instructor during the first week of class of any accommodations needed for the course. Late notification may cause the requested accommodations to be unavailable. All accommodations must be approved each semester by the Coordinator for Disability Services Marjorie Seipt room 269B West Building, 372-7351. If you have not already met with Ms. Seipt this semester you will need to do so immediately.

Course Overview

CptS 322 will cover the following topics.

- Software Engineering Overview
- Requirements Analysis
- Software Design and Implementation
- Validation, Verification, and Testing

For more details on the course schedule, please see the detailed course schedule located on the class website.

Grading Policy

The overall class grade will be broken down as follows:

- 55 % Software Engineering Project
 - 3 % Risk Management Project
 - 5 % Presentation
- 37 % Software Artifacts (written documentation)
 - 8% Software Requirements Specification (SRS)
 - 8% Software Design Document (SDD)
 - 8% Software Test Document (STD)
 - 5% Updated & Consistent SRS, SDD, STD
 - 5% User Manual
 - 3% Evaluation / Lessons Learned Write-up
- 10 % Final Product Implementation Evaluation¹
- 20 % Final Exam²
- 12 % Mid-term Exam²
- 10 % Assignments and Homework³
- 3 % Discretionary points⁴

Grading Scale⁵

A	> 93 %
A-	> 90 %
B+	> 87 %
B	> 83 %
B-	> 80 %
C+	> 77 %
C	> 73 %
C-	> 70 %
D+	> 67 %
D	> 63 %
D-	> 60 %
F	>= 0 %

Unless otherwise specified, all assignments, projects, exams, and other work in this class should be your own, individual effort.

Notes:

¹ The implementation evaluation will cap the final grade. An overall class grade may not be more than 10 % higher than the final product implementation evaluation. (i.e., One must earn at least a 'B' on their implementation if they want an 'A' in the class).

² Makeup exams will not be given without prior authorization or written documentation verifying that the student was unable to take the exam as scheduled. Unexcused missed exams will earn a 'zero'. Work-related obligations and personal emergencies are included in the valid reasons for missing an exam.

³ Late work will be penalized 20% per day, unless otherwise noted.

⁴ Discretionary points will be awarded on the basis of class participation and professional conduct. Software engineering requires both good written and oral communication skills. Discretionary points are intended to help encourage this via productive classroom discussions.

⁵ The grading scale is tentative. It will not be revised upwards, but may be revised downwards at the instructor's sole discretion.

Class Project

Students will complete one of two semester-long class projects. They are:

- Sensor Network Application (object location based upon multiple sensor readings)
- Smart-Home Application (HVAC control of a 'smart' home)

Details about these projects will be given in class and also posted on the class web site.

Assignments and Other Homework

From time to time, assignments and other homework will be given out in class. Details about these assignments will be given in class and also posted on the class web site.

E-mail will be used for official communication in this class. Please check your e-mail.