

University of Montana

ScholarWorks at University of Montana

University of Montana Course Syllabi, 2011-2015

Fall 9-1-2015

CSCI 323.01: Software Science

Robert A. Smith

University of Montana, Missoula, robert.smith@umontana.edu

Follow this and additional works at: <https://scholarworks.umt.edu/syllabi2011-2015>

Let us know how access to this document benefits you.

Recommended Citation

Smith, Robert A., "CSCI 323.01: Software Science" (2015). *University of Montana Course Syllabi, 2011-2015*. 63.

<https://scholarworks.umt.edu/syllabi2011-2015/63>

This Syllabus is brought to you for free and open access by ScholarWorks at University of Montana. It has been accepted for inclusion in University of Montana Course Syllabi, 2011-2015 by an authorized administrator of ScholarWorks at University of Montana. For more information, please contact scholarworks@mso.umt.edu.

CSCI 323: Software Engineering

Fall 2015 Syllabus

Instructor: Rob Smith, Social Science 413, robert.smith@mso.umt.edu
Office Hours: Posted on office door

Class times:
TuTh 2:10-3:30PM, SS 362

Overview:

This class will expose students to the rudiments of software engineering from a real-world perspective. At the conclusion of this class, the intent is that students have gained or honed the following skills:

- Articulation of technical ideas.
- Requirements elicitation and implementation.
- Familiarization with software design scheduling.
- Familiarization with software development processes.
- Group programming.
- Software personnel management.
- Software testing and debugging.
- Architectural design.
- Version control.

The difficulty of most CS classes seems to be the math and programming aspects. It is anticipated that the challenges in this class will be primarily social and intellectual in nature: how to solve dirty, real world problems within a realistic, group context.

Attendance:

My intent is that every class will provide enough value to entice you to attend. While there is no explicit attendance policy, there are performances and deliverables for which you will receive a grade, and in your absence you will receive a zero. Students who miss class are expected to contact a classmate to find out what they missed. There is no make up policy, however, students in extraordinary situations (think death of immediate family member or personal dismemberment) may approach me on a case-by-case basis.

Grading:

Group Course Project (75% of total):
 Requirements elicitation (5%)
 Requirements implementation (50%)
 Scheduling (5%)
 Testing (15%)
Seminar Participation: 10%
Quizzes: 10%

Final Exam (Essay): 5%

Group Participation:

Real world software development is hardly done alone. Therefore, a significant portion of your assessment for this class is for how well you can perform AS a group. Those who slack in their group participation are unlikely to receive a passing grade.

Class Mechanics:

Classes will consist of seminars, where class members will engage in a discussion topic based on readings, and project classes, where class members will role play various portions of software engineering (including development) in relation to the semester project. For the former, I will serve as the moderator and provide extra insights and questions to drive the conversation. For the latter, I will serve in various roles from customer to project manager to irate fellow employee. Because each class is different, and much of the specific content of the course will be customized depending on the class members' background and interests, it will not always be possible to indicate which classes will be seminar classes and which classes will be project classes.

Final Essay:

Write a single spaced essay responding to the question: *Given that there is no "silver bullet," how can the strengths and weaknesses of software engineering methods be leveraged pragmatically under different circumstances to yield a best practice?* Specifically address the strengths and weaknesses of different approaches and the circumstances where each apply (topics given below).

The essay must be organized with one paragraph for the introduction, one paragraph for the conclusion, and one paragraph for each of the following topics (with bold headings for each paragraph): Requirements (elicitation, implementation, and validation), waterfall vs. agile, testing, time estimation/scheduling, architectural design, documentation, and version control. There is a page limit of 6 pages. There is no minimum page count, but note that in the past very few essays of less than 6 pages contained sufficient information and organization to earn a high grade. This essay can be submitted at any time up until the opening of the last day of class. It must be submitted as a printed hardcopy.

Example: **Documentation.** The primary tools in documentation...

Rubric for essay (100 pts total):

Cohesiveness.

The essay has a clear, overarching message systematically addressed in each subtopic (15 pts).

The essay has a message that is frequently supported throughout (10 pts).

The reader can tease out a message from this essay (5 pts).

The points made in this message do not seem linked (0 pts).

Completeness.

The essay addresses each of the above specified topics (10 pts).

The essay addresses some but not all of the topics (5 pts).

The essay addresses none of the topics (0 pts).

Correctness. (Meaning accuracy of information about software engineering topics).

The technical statements made in the essay are correct (20 pts).

The technical statements made in the essay are correct (10 pts).

This essay contains an abundance of technical error (0 pts).

Content.

The essay contains a high level of relevant information about each topic (35 pts).

There is relevant information missing about some topics, but the essay as a whole still contains a great deal of information (25pts).

The relevant information missing in this essay significantly weakens the otherwise content-rich essay (10 pts).

The amount of missing information in this essay is consistent and substantial (0 pts).

Persuasiveness.

The essay is persuasive towards the premise (15 pts).

The essay is mostly persuasive towards the premise (10 pts).

The essay is somewhat persuasive towards the premise (5 pts).

The essay does not effectively support the premise (0 pts).

Communication.

The essay is free from spelling errors, is well formatted, and uses effective vocabulary (5 pts).

The essay contains some minor spelling errors, is mostly well formatted, and/or uses mostly effective vocabulary (4 pts).

The essay contains a significant amount of spelling errors, is not well formatted, and/or is poorly written (0 pts).

Cheating:

Any form of collusion or dishonesty will result in an "F," and may be prosecuted to the full extent allowable by University standards. Please see the University code of conduct for definitions of cheating.

Incompletes:

"Incomplete for the course is not an option to be exercised at the discretion of students. In all cases it is given at the discretion of the instructor...." Some guidelines for receiving an incomplete are listed in the catalog which include having a passing grade up to three weeks before the end of the semester and being in attendance.

"Negligence and indifference are not acceptable reasons." Also note that there may be financial aid implications.

Late Drops:

The University's policy on drops after 45 days of instruction is very specific. The Computer Science Department follows this policy rigorously. There are five circumstances under which a late drop might be approved: registration errors, accident or illness, family emergency, change in work schedule, no assessment of performance in class after this deadline. Except in very unusual circumstances, I will only approve late drops if there is documented justification for one of these circumstances.

Disabilities:

This course is accessible to and usable by otherwise qualified students with disabilities. Those requesting specific accommodations, such as extra time to turn in an assignment, must notify the instructor with a Disability Services letter during the first two weeks of class in order to provide ample time for class reorganization.

Text Book:

There is no required text book for this course. I will provide any required reading electronically.