Syllabus for CSC-383
Software Engineering II
Fall 2008

Instructor Information:

- Dr. Michael Farmer
- Office Hours: Mon. 1-2, Tue. 1-4.
- Office: 214 MSB
- Phone: (810) 762-3423
- Email: farmerme@umflint.edu

Course Information:

- To date most of your classes have required you to write many lines of code. You have spent many late nights searching for syntax errors, etc. In other classes such as the algorithms class you have learned the detailed design of some common algorithms. For example, you have learned the cost in terms of memory and speed of many standard tasks, such as searching and sorting. Recall in CS-382 we learned the engineering **methods and techniques** associated with the key stages of engineering a software system. Now in CS-383, we will address the aspects of **management and control** of the development of a software product. The objective of this class is to provide the student with the knowledge of a variety of process models within which to apply the tools and techniques developed in 382. It also will provide the student with the skills to estimate the costs of a development, develop metrics to manage the development, and the standards with which to document the software development.

Course Objectives:

- The goal of the class is to continue the evolution from programmers to software engineers to ultimately software development leaders. You will learn the management techniques needed to accomplish this. Our final products are project management plans which define our process, our metrics for monitoring progress, and our documentation for capturing the essential knowledge of the project. The objective is to provide the students with a good understanding of the standard process models that professional engineers use to minimize the development risk of a software product. In industry this skill is what separates a typical programmer from a potential project leader.

- As with CSC-382, there will be **no code written** in this class. We are focusing on understanding how to establish, manage, and document a software development project. The tools and methods to make this job easier are a key element of what will be learned in this class

- The specific course objectives are as follow:
    - Be able to clearly articulate the key issues associated with managing a software project,

- Be able to clearly articulate the standard prescriptive and agile software development models,
- Be able to realistically estimate the effort and cost of a software project,
- Be able to develop a schedule for a set of software project,
- Be able to define a suitable set of software metrics for tracking performance on a project,
- Be able to clearly articulate how to manage the quality of a software project and its ultimate outcomes,
- Be able to define the key issues associated with software project risk.

Class Format:

- The best way to learn Software Engineering is by **doing.** We will use lectures to explain techniques and methods we will be using. Then we will use on-going in-class case study to try our hand at the techniques. This is where class participation is critical. We will be discussing the merits and shortcomings of the various software management processes, and everyone's contributions are expected.

Assessment Mechanisms:

- Your understanding of the course objectives will be assessed by your writing and my reviewing the following deliverables:

    - Written homework assignments to demonstrate understanding of the individual specific course objectives
    - Develop an actual cost estimate and schedule for your project
    - Develop a set of risks for your project and define how you will manage them
    - Develop a Project Management Plan that describes how you will track all of the issues associated with executing a project
    - Project Presentation (This will include MS Power point slides reviewing the materials defined in the project plan).

Prerequisites:

- CSC-382 or Consent of Instructor

Book:

- *Software Engineering: A Practitioner's Approach, 6/e, by* Roger Pressman.

Other Good Reference Books:

- *Software Engineering: 7/e*, by Ian Sommerville
- *Software Management* by Donald Reifer
- *The Personal Software Process,* by Watts Humphrey
- *The Mythical Man-month* by Frederick Brooks Jr.
- *Extreme Programming,* by Kent Beck

Tools:

- *Microsoft Project*

- *Microsoft Word* – available everywhere
- *COCOMO* (cost estimating software)

Grading:

- Homework (20%), Mid-term (20%), Group Project (20%), Individual Research Paper (20%), Final (20%)
- Class participation is an expected portion of the course work
- Scale:

A's	B's	C's	D's
97-100: A+	87-89: B+	75-79: C+	65-69.9: D+
92-96.9: A	82-86.9: B	72-74.9: C	62-64.9: D
90-91.9: A-	80-81.9: B-	70-71.9: C-	60-61.9: D-
			0-59: E

Attendance:

- Regular class attendance is essential unless student has informed instructor they are a distance learner. Prior approval to be a distance learner is required from the instructor.

Cyber classroom:

- The website for the video lectures is:
  http://mediasitelx.csesp.umflint.edu/mediasite/viewer
  Username: student, Password: CSESP

Late Homework Policy:

- Assignments will be marked down 10% per day late. Please only turn assignments in in-person or via the Blackboard digital dropbox. Do not email assignments to me.

Academic Misconduct:

- Copying others work, plagiarizing external references without giving due credit, and cheating in exams are strictly forbidden. Please take pride in your own work, and feel free to ask me if you need help or assistance to maximize your learning.

Proposed Schedule: (Note this schedule is subject to change)

Week	**Monday**	**Wednesday**
Week 1 (8/25)		Testing (Ch. 13 & read IEEE article)
Week 2 (9/1)	*Labor Day*	Testing II (Ch. 14)
Week3 (9/8)	Overview of the software engineering problem (from Mythical Man-Month)	Software Quality Systems CMM, ISO-9001, SPICE, etc. (Ch. 2)
Week 4 (9/15)	Presciptive Process Models (Ch. 3)	Adaptive and Agile Process Models I (Ch. 4)
Week 5	Adaptive and Agile Process	Personal Software Processes (Ch. 2)

(9/22)	Models II (Ch. 4 plus “Xtreme Programming”)	
Week 6 (9/29)	Project Management Concepts I (Ch. 21)	Project Management Concepts II (Ch. 21)
Week 7 (10/6)	Software Metrics (22.1, 22.2) and (15.1 and 15.2)	Software Product Metrics (Ch. 15.3,15.4, and 15.5)
Week 8 (10/13)	Review for Midterm	**Mid-term**
Week 9 (10/20)	Software Quality and Metrics I (Ch. 22.3-22.6)	Software Quality and Metrics II (Ch. 22.3-22.6)
Week 10 (10/27)	Software Cost Estimation I (Ch. 23)	Software Cost Estimation II (Ch. 23) & COCOMO Cost estimation tool
Week 11 (11/3)	Software Project Scheduling (Ch. 24)	Risk Management I (Ch. 25)
Week 12 (11/10)	Risk Management II (Ch. 25)	Software Quality Management I (Ch. 26)
Week 13 (11/17)	Software Quality Management II (Ch. 26)	*Thanksgiving*
Week 14 (11/24)	Change Management I (Ch. 27)	Change Management II (Ch. 27)
Week 15 (12/1)	Software Documentation Standards	Building the SW Development Environment – CASE Tools
Week 16 (12/8)	Course Review	