TTMG 5006T
MANAGEMENT OF SOFTWARE ENGINEERING PROJECTS

Winter 2012
Department of Systems and Computer Engineering
Carleton University

Professor Weiss
ME 4480
weiss@sce.carleton.ca

This course outline is a living document. Improvements may be made as necessary during the term. Version 0.3

Instructor availability
The instructor is available via e-mail any time. Office hours before class by appointment (online/offline).

Calendar description
TTMG 5006T [0.5 credit] Management of Software Engineering Projects
Models for the development of software. Software project management tools. Quality control. Risk assessment and management. Examples are drawn from software development in telecommunications applications.

Prerequisites: TTMG 5001, TTMG 5002

Course objectives
This course examines topics relevant to the management of software engineering projects. It emphasizes the perspective of the engineer or computer scientist in the role of technical project manager, responsible for planning and controlling the development activities that result in the delivery of software products. We examine theory, processes, methods, and tools. Although our scope includes well-established traditional practices, we are particularly interested in emerging practices, on-going research, and exploring the controversies within the field. The specific focus of the course will be on customer development, agile practices, and open source.

Rationale
This course is designed to build capability and knowledge in the management of large, complex and changing software systems. Students will learn about different perspectives on managing software projects, gain familiarity with the practitioner and research literature, and become proficient with practical managerial skills which can add value in their companies.

Benefits
This course prepares students to undertake thesis research or applied projects in the areas of software project management, process improvement, management of risk and quality, design and development of software products, and the management of operations within software-intensive technology companies.

Class Sessions
This course will be offered in-class and online. Remote students can participate by logging into conference room "TTMG 5006" with password "student" at http://present.carleton.ca. For the audio portion of the conference, you can call into the conference server using the phone numbers provided on login, or using the built-in VOIP feature (click on the headset icon). When using VOIP, you must use a headset.

Please see the tutorials on participating in an online classroom on the present.sce.carleton.ca site.

For the weekly sessions there will be assigned readings and tasks.

During the student group presentation sessions, groups will be asked to make short presentations on their assignments (max. 10 minutes except where explicitly stated otherwise; please practice so you stay on time). Each

group decides who presents what and in which order. Before 6 p.m. EST the day prior to when presentations are due, each group will distribute to all members of the class the slides to be presented the next day. No exceptions.

The course material and recordings of the class sessions will be made available on the Moodle learning content management system at http://cms.sce.carleton.ca.

Student Evaluation
Course participants are required to complete two group assignments and participate actively in class (discussion and assigned tasks). To determine the course grade, these weights apply:

- Assignment 1 (group) 30%
- Assignment 2 (group) 30%
- Class participation (individual) 10%
- Exam (individual) 30%

Assignments submitted late and presentations not made will receive a grade of zero. All students in a group receive the same grade. Final grade reports will follow Carleton University guidelines.

Assignment 1 (30%)
Groups of 2-3 people.

Document a pattern that addresses a software project management problem. A pattern is a reusable solution to a common problem. Patterns follow a specific format. Each pattern must discuss why this problem is a challenge, and present a solution to the problem from the literature and or personal experience. The pattern also needs to discuss the consequences of applying the solution, and describe known uses of the solution.

- Format: pattern to be posted to the course wiki (max 1000 words)
- Provide feedback on the first draft of all patterns by other groups: in-class workshop

Assignment 2 (30%)
Groups of 2-3 people, or depending on number of people enrolled in the course.

Develop a lecture on an open source development topic in Fogel (2006):

- Format: 30 min presentation with 15 min discussion period
- Post three questions on the chapter(s) assigned to you one week ahead of class

Assignment topics (first three topics will be covered in Session 5, the remaining three in Session 6):

1. Introduction, Chapter 1, 9-17
   Getting started, Chapter 2, 18-35
2. Technical infrastructure, Chapter 3. 36-62
3. Social and political infrastructure, Chapter 4, 63-70
   Example instructions for reporting bugs, Appendix D, 175-176
   Money, Chapter 5, 71-83
4. Communications, Chapter 6, 84-110
5. Packaging, releasing, and daily development, Chapter 7, 111-129
6. Managing volunteers, Chapter 8, 130-152

Please keep in mind when preparing your slides that you are making a 30 min presentation. At 2 min per slide, you will only be able to cover 10-15 slides in 30 min. The goal of the presentation is not to cover every aspect discussed in the book in elaborate detail. A good presentation adds insight on the material covered in the book by synthesizing, commenting, and linking to lessons from other sections of the course.

Group size depends on the number of students in the class. Priority goes to topics 2-6.

Class participation (10%)
Active class participation is an important component of this class:

1. Participation in class discussions (contribute to lessons learned at the end of each class, lead 4 discussions in class or online, provide feedback on the assignments of your classmates).
2. Summarize one chapter of the book by DeMarco et al. (2008) and post your summary to the course wiki. Your summary should contain three key insights from the chapter.

Take-home exam (30%)
The exam will handed out during the last class: March 29
The exam is due on: April 5 at 6 pm, submit online

Group work and free loaders
Group work is an important component of this course. You may elect to work in the same group to prepare both assignments or work in two different groups. Group conflicts are to be dealt with by the group in a way that is fair, fast and without personal attacks. The instructor does not settle group disputes.

The instructor will dissolve a group that is late submitting an assignment. A group of three is expected to deliver better work than a group of two.

Free loaders are not welcome anywhere. This course is no exception. The best way to deal with free loaders is to not include their names in the first page of the group assignments. If a student's name does not appear in an assignment submitted by his or her group, the student must submit his or her own assignment. Failure to do so, the student will receive zero for the assignment. There is zero tolerance for free loaders.

Students with disabilities
Students with disabilities who require academic accommodations in this course are encouraged to contact the Paul Menton Centre (PMC) for Students with Disabilities to complete the necessary forms. After registering with the PMC, make an appointment with me in order to discuss your needs at least two weeks before the first assignment is due. This will allow for sufficient time to process your request

Plagiarism
Plagiarism (copying and handing in for credit someone else's work) is a serious instructional offence that will not be tolerated. Please refer to the section on instructional offences in the Graduate Calendar for additional information. Plagiarism is against the TIM culture. A case of plagiarism will be referred to the Chair of the Department and the Carleton University Ethics Committee. The instructor will not deal with the matter directly. The university has clear processes to deal with students who are suspected of plagiarism.

Administrative details
These are the rules of conduct for this course:

- Please notify the instructor via e-mail, if you will not attend a class.
- **You must be prepared for each class.** You do so by reading the material assigned and being prepared to discuss in class how what was read can be applied in product development organizations.
- Each presenter must make his/her slides available to all other students by noon the day before.

Better Journals
Journal of Project Management
IEEE Transactions on Engineering Management
IEEE Software
ACM Communications
ACM Transactions on Software Engineering and Methodology
Empirical Software Engineering
Management Science

Management of Software Engineering Project: Schedule

Date	Topic	Readings
Jan 4	Session 1: Introduction	Course outline
Jan 11	Session 2: Management challenges	Sauer & Reich (2008) Baskerville et al. (2003) Royce (2005) Persson & Mathiassen (2010) Clements et al. (2005)
Jan 18	Session 3: From opportunity to requirements I	Leonard & Rayport (1997) Meyer (2007)* Lehtola et al. (2009) Martin et al. (2009)
Jan 25	Session 4: From opportunity to requirements II	Beyer et al. (2004) Gulliksen et al. (2003) Dieste et al. (2008) Alexander & Beus-Dukic (2009)*
Feb 1	Session 5: Feedback	Cao & Ramesh (2007) Petter (2008) Cao & Ramesh (2008) Bergin (2006b)
Feb 8	Session 6: Planning and execution I	Bergin (2006a)
Feb 15	Session 7: Planning and execution II	Yap (2006) Hoda et al. (2009)
Feb 22	Winter break	
Feb 29	Session 8: Workshop on first version of Assignment 1	
Mar 7	Session 9: Producing open source I Assignment 2	Fogel (2006)
Mar 14	Session 10: Producing open source II Assignment 2	Fogel (2006)
Mar 21	Session 11: Architecture and organization	
Mar 28	Session 12: Reuse and software product lines	
Apr 4	Exam due at 6 pm	

Readings

The course is organized around the content of three books, complemented by articles:

DeMarco T., Hrushka, P., Lister, T., McMenamin, S., Robertson, J., & Robertson, S. (2008), *Adrenaline Junkies and Template Zombies*, Dorset House Publishing.

Fogel, K. (2006), *Producing Open Source Software: How to Run a Successful Free Software Project*, O'Reilly.

Coplien, J. & Bjørnvig, G. (2010), *Lean Architecture for Agile Software Development*, Wiley.

Coplien & Bjørnvig (2010) can be accessed as an e-book in the library. Fogel (2006) is freely available under at http://producingoss.com. De Marco et al. (2008) can be purchased as an inexpensive ebook from Dorset House.

Readings for Session 2

Sauer, C., & Reich, B. (2008), Rethinking IT project management: evidence of a new mindset and its implications, *International Journal of Project Management*, 27, 182-193.

Baskerville, R., Ramesh, B., Levine, L., Pries-Heje, J., & Slaughter, S. (2003), Is Internet-speed software development different?, *IEEE Software*, 20(6), 70-77.

Royce, W. (2005), Successful software development style: steering and balance, *IEEE Software*, 22(5), 40-47.

Persson, J., & Mathiassen, L. (2010), A Process for Managing Risks in Distributed Teams, *IEEE Software*, 27(1), 20-29.

Clements, P., Jones, L., Northrop, L., & McGregor, J. (2005), Project Management in a Software Product Line Organization, *IEEE Software*, 22(5), 54-62.

Readings for Session 3

Leonard, D. & Rayport, J. (1997), Spark innovation through empathic design, *Harvard Business Review*, 75, Nov-Dec, 102-115.

* Meyer, M. (2007), *The Fast Path to Corporate Growth*, Oxford University Press. I will be summarizing Chapter 4 on Understanding User Needs. (You are not required to buy the book.)

Lehtola, L., Kauppinen, M., Vähäniitty, J., & Komssi, M. (2009), Linking business and requirements engineering: is solution planning a missing activity in software product companies?, *Requirements Engineering*, 14(2), 113-128.

Martin, A., Biddle, R., & Noble, J. (2009), XP customer practices: A grounded theory, *Agile Conference*, 33-40, IEEE.

Readings for Session 4

Beyer, H., Holtzblatt, K., & Baker, L. (2004), An agile user-centered method: Rapid contextual design, *Extreme Programming and Agile Methods – XP/Agile Universe*, LNCS 3134, 527-554, Springer.

Gulliksen, J., Göramsson, B., Boivie, I., Blomkvist, S., Perrson, J., & Cajander, Å. (2003), Key principles for user-centred systems design, *Behaviour and Information Technology*, 22(6), 397-409.

Dieste, D., Juristo, N., & Schull, F. (2008), Understanding the customer: What do we know about requirements elicitation?, *IEEE Software*, 25(2), 11-13.

* Alexander, I. & Beus-Dukic, L. (2009), *Discovering Requirements*, Wiley. I will be summarizing Chapter 15 on Putting it All Together. (You are not required to buy the book.)

Readings for Session 5

Cao, L., & Ramesh, B. (2007), Agile software development: Ad hoc practices or sound principles, *IT Professional*, 9(2), 41-47.

Petter, S. (2008), Managing user expectations on software projects: lessons from the trenches, *International Journal of Project Management*, 26(7), 700-712.

Cao, L., & Ramesh, B. (2008), Agile requirements engineering practices and challenges: an empirical study, *IEEE Software*, 25(1), 60-67.

Bergin, J. (2006b), Patterns for agile development practice, Part 3, *PLoP* (2006).

Readings for other sessions will be updated in the next version of this course outline

Suggested Books

Many of these are available online (on Safari via the Carleton library, or on the Web).

Adzic, G. (2009), *Bridging the Communication Gap*, Neuri Limited.

Alexander, I. & Beus-Dukic, L. (2009), *Discovering Requirements*, Wiley

Cho, A. (2011), *The Jazz Process: Collaboration, Innovation, and Agility*, Addison-Wesley.

Ebert, C. (2006), *Global Software Engineering*, Ready Note, IEEE.

Elssamiadisy, A. (2008), *Agile Adoption Patterns: A Roadmap to Organizational Success*, Addison Wesley.

Harrison, N. & Coplien, J. (2006), *Organizational Patterns of Agile Software Development*, Addison Wesley.

Highsmith, J. (2004), *Agile Project Management*, Addison Wesley.

Hohmann, L. (2003), *Beyond Software Architecture: Creating and Sustaining Winning Solutions*, Addison Wesley.

Lopp, M. (2007), *Managing Humans*, Apress/Springer.

Meyer, M. (2007), *The Fast Path to Corporate Growth*, Oxford University Press.

Poppendieck, M., & Poppendieck, T., *Implementing Lean Software Development*, Addison Wesley.

Richardson, J., & Gwaltney, W. (2007), *Ship It!*, The Pragmatic Bookshelf.

Rosenberg, S. (2007), *Dreaming in CODE*, Crown.

Royce, W., Bittner, K., & Perrow, M. (2006), *The Economics of Iterative Software Development*, Addison-Wesley.