Winter 2008

CEG 468/668-01: Managing the Software Development Process

John A. Reisner
Wright State University - Main Campus, john.reisner@wright.edu

Follow this and additional works at: https://corescholar.libraries.wright.edu/cecs_syllabi

Part of the Computer Engineering Commons, and the Computer Sciences Commons

Repository Citation
https://corescholar.libraries.wright.edu/cecs_syllabi/1179

This Syllabus is brought to you for free and open access by the College of Engineering & Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact library-corescholar@wright.edu.
CEG 468/668: Managing the Software Development Process  
Winter Quarter, 2008

Course Description
This course will cover some of the challenges and issues associated with managing software projects. Emphasis will occur on two fronts: (1) the software project manager’s view (that is, what considerations and obstacles confront project managers during software development), and (2) the organizational view (that is, how organizations can foster a climate where software project management is performed effectively throughout an organization). Topics covered will include:
- Software project management via POMA: Planning, Organizing, Monitoring, and Adjusting
- The CMM (Capability Maturity Model) and CMMI
- The rise of agile methodologies in response to heavyweight CMM methodologies and processes
- Balancing the advantages and strengths of both “agile” and “disciplined” approaches to software project management

Course Textbook & Other References
B. Boehm and R. Turner, Balancing Agility and Discipline: A Guide for the Perplexed, Addison-Wesley, 2004. This is a required textbook for this course.

This course will draw from materials in other texts as well. The materials related to POMA will draw heavily from the book Managing Software Projects by Frank Tsui (Jones and Bartlett Publishers, 2004). The course will be structured in such a way that students will not need to obtain this textbook.

The course will also teach much on the CMM. It is strongly recommended that students obtain at least one CMM or CMMI reference for use throughout the course. Some possibilities include:
- a book borrowed from the university library
- a used textbook, perhaps purchased on-line
- a downloaded e-book
- a document available on the worldwide web

When looking for a CMM or CMMI textbook to purchase or borrow, try to find one that discusses the Key Process Areas (KPAs) for each of the five levels (this is often readily apparent simply by examining the Table of Contents, or skimming the book). Some examples of decent CMM and CMMI texts include:
- The Capability Maturity Model: Guidelines for Improving the Software Process by the SEI at CMU*
- Implementing the Capability Maturity Model by James R. Persse
- Interpreting the CMMI: A Process Improvement Approach by M.K. Kulpa and K.A. Johnson
- CMMI Distilled by Dennis M. Ahern
- CMMI: Guidelines for Process Integration and Product Improvement by Mary Beth Chrissis, et. al.

*Note: At the time of this writing, several used copies of this book are available at Amazon.com for less than $5. Also, there is another less-than-five-dollars reference available on Amazon.com: A Guide to the CMM: Understanding the Capability Maturity Model for Software (Spiral-bound) by Kenneth M. Dymond. I have not yet reviewed this resource, but I have ordered a copy for myself. Ask me in a week or so for a review if you are curious.

This is not an exhaustive list, just a sampling to get you started in the right direction.

Also, in lieu of obtaining a textbook, there are many CMM and CMMI materials available over the web. The Software Engineering Institute (SEI) website has several such references on-line; one good reference that provides an overview of the CMM can be found at http://www.sei.cmu.edu/pub/documents/93.reports/pdf/tr24.93.pdf. Note: the website will tell you that SEI is no longer maintaining the CMM, because it has been superceded by the CMMI. While this assertion is true, a reference to the CMM will suffice for the purposes of this course. The aforementioned document is listed as a “good” reference because it is a mere 81 pages long. The instructor wishes to avoid the ire that would be incurred if he were to insist a class full of students all print a 600-page reference after the first class.
Instructor Contact Info

John Reisner
Office Hours after class or by appointment
Work Phone: 255-3636 x7422 (Wright-Patterson AFB)
email: john.reisner@wright.edu (if you want a timely response, please CC: john.reisner@afit.edu)

The instructor is an adjunct faculty member. Most contact will be done via WebCT, or in after-class discussions. Other meetings can be arranged.

If, at any time, you are having trouble accessing course materials via WebCT, please send me an email immediately. The sooner I am aware of a problem, the sooner I can fix it. Because I have the instructor's view of WebCT, I sometimes mistakenly believe materials have been posted when in fact students cannot access them. Your support in this matter is greatly appreciated.

Course Objectives

Each student should be able to
- Better understand some of the strategies used to manage the development of large-scale software systems.
- Understand the goals of various software process models.
- Explain the difference between a software process model and a software lifecycle model.
- Comprehend how the Software Engineering Institute's (SEI) Capability Maturity Model (CMM) can be used to measure and improve an organization's software development process.
- Understand some of the differences between the CMM and CMMI, and understand the motivations for the model's evolution.
- Describe the benefits, limitations, and misuses of CMM and CMMI evaluations within a software development community.
- Understand some of the advantages, disadvantages, and tradeoffs between agile methods and more formal alternatives.

Course Format

This course will be taught in a collaborative manner – meaning that, during class time, much of the material will be discussed among the class, rather than presented in a strict lecture format. Students will be expected to have done any readings or research assigned prior to the lecture, and able to contribute to the discussion in an informed, intelligent, and constructive manner. See notes under “Class Participation” in the Course Grading section, on the following page, for more information.
Course Grading

20% Homework Assignments
- Homework assignments are designed to facilitate deeper comprehension about a lecture topic (in other words, these are “think and respond” assignments).
- There may be up to two assignments per week, but some weeks may have one or zero assignments.
- Answers to these homework assignments generally run about half page to one page in length, and should not take too long to complete.
- Details about these assignments will be found on WebCT.
- Normally, these assignments will be due on Tuesday of the week following the assignment. In other words, you will have one week to complete an assignment corresponding with a Tuesday lesson, and five days to complete an assignment given on Thursday. Any exceptions to this policy will be mentioned when the homework is assigned.
- Assignments are due at the start of the class/lab session; please have them printed out and ready to turn in at the start of class. If you are unable to attend class, email will be accepted. Emailed assignments should be timestamped before class time (skipping class does not give you a homework extension).
- These assignments will be graded using the SUE grading system (explained on the following page).

20% Class Participation
- The instructor will annotate a class roster during class discussions, tracking the interaction of each student individually.
- Students should not feel compelled to blurt out something every lecture, but long-term contributions will be assessed.
- The instructor will attempt to accommodate these grades with respect to individual personality traits and potential language barriers.
- As a general rule, if you come ready to participate, you will do fine. If you come with an indifferent and apathetic attitude, your grade will suffer accordingly.

20% Class Project
- Each student will make a contribution, individually or in small groups, to a class project.
- This work will be done throughout the course, and turned in on the day of the last class.
- More information will become available as the class progresses. This information will be posted on WebCT for reference.

20% Mid-term Exam
- Mixed-format exam, administered in class.

20% Final Exam
- Comprehensive, mixed-format exam, administered during scheduled exam time.

Final course grades will be assigned at the instructor’s discretion, after all grades have been calculated. Grades over 94 will be A, over 86 will be B, over 78 will be C, over 70 will be D, although this scale can be (and frequently is) curved.
The SUE Grading System

Homework for this class will be ordinarily graded on a three-tier scale: the work will be graded as Satisfactory, Unsatisfactory, or Exemplary.

If your submission was Satisfactory, then your grade will be S, which will translate to a 90. Don't think of a 90 as "losing 10 points," think of it as getting ample credit for satisfactory work.

Occasionally, I receive an assignment with great originality and insight, reflecting much forethought and effort. Not only do I find these assignments enjoyable to read, I sometimes find myself thinking, "This is as good as or better than anything I could put in an answer key." Such exemplary work is graded E, which translates to 100. If submitted work indicates either a lack of understanding of basic concepts, or an apparent apathetic carelessness, then it will be graded as Unsatisfactory, and a numeric grade will be assigned accordingly. If I think the problem lies with misunderstanding the basic ideas, then I will usually provide some personal feedback, with the aim of helping you understand the material better.

Of course, I also reserve the right to stray from this guideline some. For example, I might grade with values such as 85 or 95 (corresponding to S- or S+), or even an S++, which would be a 98. After reading 25 or so essays the same topic, I get a pretty good idea of which papers are more well-thought-out than others. The ones that are "more than satisfactory" receive grades such as 92, 95, or 97, while the truly superior works will receive an E (100). Again, don't ask me what was wrong if your grade is a 90. A 90 means you understood the assignment and did a good job of presenting your response.

I also reserve the right to deduct points for late assignments, depending upon how late the work was turned in, how much advanced notice I was given about when I could expect the work, and any extenuating circumstances that may have applied.

Overall, my goal is to assign homework that requires thought, thereby reinforcing understanding and increasing retention. Many of these assignments don't have answers that are "right or wrong," but, rather, they are either well-supported and articulated, or they are not. Hence, it is impossible to grade these assignments without some measure of subjectivity. I do everything I can to grade each assignment fairly and equitably.
## Course Schedule (possibly subject to change)

### PART I. Software Project Management Across the Project Lifecycle

<table>
<thead>
<tr>
<th>Lesson</th>
<th>Date</th>
<th>Lesson Topics</th>
<th>Assigned Textbook Reading</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tue Jan 8</td>
<td>Course Introduction</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Thu Jan 10</td>
<td>Project Management During Requirements</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Tue Jan 15</td>
<td>Overview of the CMM</td>
<td>Appendix C</td>
</tr>
<tr>
<td>4</td>
<td>Thu Jan 17</td>
<td>PO – Planning, Organizing</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Tue Jan 22</td>
<td>CMM Level 2</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Thu Jan 24</td>
<td>Project Management During Design</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Tue Jan 29</td>
<td>CMM Level 3</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Thu Jan 31</td>
<td>MA – Monitoring and Adjusting</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Tue Feb 5</td>
<td>The CMM: A Panacea? Problems with the CMM; Advent of Agile Methods</td>
<td>Appendix B, <em>plus</em> Chapter 1</td>
</tr>
<tr>
<td>10</td>
<td>Thu Feb 7</td>
<td>Project Management During Coding</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Tue Feb 12</td>
<td>MIDTERM EXAM</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Thu Feb 14</td>
<td>Agile Methods Compare &amp; Contrast</td>
<td>Appendix A</td>
</tr>
<tr>
<td>13</td>
<td>Tue Feb 19</td>
<td>Selection Factors</td>
<td>Chapter 2</td>
</tr>
<tr>
<td>14</td>
<td>Thu Feb 21</td>
<td>“A Day in the Life”</td>
<td>Chapter 3</td>
</tr>
<tr>
<td>15</td>
<td>Tue Feb 26</td>
<td>CMM Level 4</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Thu Feb 28</td>
<td>CMM Level 5</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Tue Mar 4</td>
<td>Attaining a Healthy Balance, Part 1</td>
<td>Chapter 5</td>
</tr>
<tr>
<td>18</td>
<td>Thu Mar 6</td>
<td>Attaining a Healthy Balance, Part 2</td>
<td>Chapter 6</td>
</tr>
<tr>
<td>19</td>
<td>Thu Mar 11</td>
<td>Mock CMM Assessment</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>Thu Mar 13</td>
<td>Project Demos and “Debriefings”</td>
<td></td>
</tr>
</tbody>
</table>