



## MET CS 473 / 673 SC1

# Introduction to Software Engineering

## Summer 1, 2025 - SYLLABUS

### **Course Description:**

---

Today software systems are developed through the global collaboration of high-performance teams. The course provides an in-depth examination of key software engineering techniques. Term project consists of in-process submissions that are thoroughly reviewed among peers, as well as the final delivery of a working system prototype.

The course represents a deliberate attempt to advance several individual disciplines into a systematic approach with a single goal of delivering product more effectively. It is uncharacteristic of a software engineer to be unaware of requirements or peer reviews. Software engineer might not be an expert in system test, although he or she must have a clear concept of how it is done so to be able to contribute, collaborate and lead. Each part of the course offers a significant depth and it is up to you how far you explore each subject.

### **Course Objectives:**

---

Upon successful completion of this course, you will be prepared to:

- Justify, implement and manage a global product development
- Play an effective role of a Software Engineering Manager in a context of IEEE CSDP (Certified Software Development Practitioner) with a specific emphasis on organizational policies as guiding principles.
- Solicit, define and scope requirements as part of the product backlog grooming.
- Assess common security threats and establish corresponding deterrents.
- Select an estimation method that is appropriate for a specific phase. Oversee adoption of a consistent methodology to narrow the Cone of Uncertainty.
- Play a role in a peer review verification, request and provide constructive and concise comments.
- Support the Scrum delivery framework and become aware of several agile

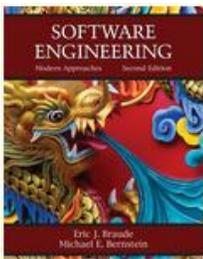
certification paths.

- Evaluate software development tools (approved, allowed, restricted), while following the Magic Quadrant technique.
- Articulate a strategy for system and unit test leading to continuous integration and delivery.
- Structure a project asset library aiming at single-click navigation to a requested artifact.
- Provide leadership to a process program that is mapped into PMI and/or SEI CMMI as an improvement model.

## Course Resources:

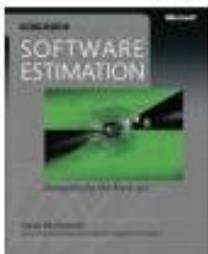
---

There are several books referenced throughout the course. These books are purchased for you by ACDP of Boston University. You do not need to purchase them. Most of the course concepts are expounded through the class notes comprised of some two hundred pages. You will be also required to search through various on-line resources.



Software Engineering – Modern Approaches (2nd edition), by Eric Braude & Michael Bernstein ISBN 10: 1-4786-3230-5 ; ISBN 13: 978-1-4786-3230-6 .

Eric Braude is an Associate Professor of Computer Science at Boston University. He teaches software design, machine learning, and algorithms. Braude has many years of experience as a researcher, software engineer, and senior R&D manager in industry and government. He has served as technology adviser to corporations such as Philips, Lockheed, Lucent, and MITRE Corporation.



Steve McConnell, (2006) Software Estimation: Demystifying the Black Art. Redmond, WA: Microsoft Press, ISBN: 0735605351.

Steve McConnell is a well-known industry practitioner, Chief Software Engineer at Construx. Software Development magazine named Steve one of the three most influential people in software industry along with Bill Gates and Linus Torvalds. Steve's book on estimation is available in electronic form and could be delivered instantly.



Bose, I., Huang, M-H., & Huang, M. (2006). Jharna Software: The move to Agile. Harvard Case No. HKU613.

This case study is used in one of the assignments, where you will apply elements of quantitative analysis and systematic decision making toward agile methodology. Many organizations are looking for a clear path to improve their product delivery with Agile.

# Instructor:

---



[Alex Elentukh](#)

Email: [elentukh@bu.edu](mailto:elentukh@bu.edu)

Tel: (617) 335-6135

**Office Hours & Questions:** You will have ample opportunity for questions during live sessions or through email. In addition, I will be available to stay late after class. Finally, I am available for ad-hoc conference call discussions upon request. Do not be surprised to receive a call from me, as I am always interested to learn what you are thinking.

My name is Alex Elentukh, and I will be your instructor for the course. To give you some background about myself, during my 25 years professional career I taught various software-engineering courses at Boston and Northeastern University and authored numerous papers on quality assurance and software reliability. Most recently taught the MET CS633, CS473 and CS673. At EMC, as an Agile coach, I introduced the backlog grooming to improve collaboration between product owners and engineering. As an enterprise architect at Fidelity, I managed the regression-intensive verification used by multiple scrums, cutting the field complaint rate by fifty percent. I also held the positions of QA director at two successful start-ups, Jupiter and Reveal.

Here are several papers based on student projects relevant to this course,

- A team of students traveled to Jakarta to present the implementation of one of term projects at CITSM 2018 conference.
- The paper “Improving Teaching and Learning Effectiveness of Computer Science Courses” was presented at CSECS 2019 conference in Germany.
- The paper “Empirical Study from Final Exams to Demystify the Notion of An Average Software Engineer” received an award for the best presentation at ICESE 2023 in Athens.
- The paper “People Make Mistakes - A Survey of Common Causes of Software Defects” was presented at CSECS 2023.
- During CSECS 2024 the paper “Building a Chatbot to Adopt an Effective Learning Strategy” was presented in Sofia.

Most importantly, I am here to help - if you have any questions or need anything at all during the course, please do not hesitate reaching out.

# Course Grading Information:

---

## Grade Weighting & Grade Conversion:

---

The following table summarizes the four kinds of graded items along with their weight. These are four avenues for you to learn. These are four angles to illuminate the course content.

| DELIVERABLE  | WEIGHT |
|--------------|--------|
| Quizzes      | 20%    |
| Assignments  | 20%    |
| Term Project | 30%    |
| Final Exam   | 30%    |

The following table details the translation of a numerical grade into a letter grade. For example,

- 95.25 translates into (A)
- 94.75 translates into (A-)
- 89.75 translates into (B+)

| LETTER GRADE | NUMERICAL GRADE      |
|--------------|----------------------|
| A            | $\geq 95 - \leq 100$ |
| A-           | $\geq 90 - < 95$     |
| B+           | $\geq 85 - < 90$     |
| B            | $\geq 80 - < 85$     |
| B-           | $\geq 77 - < 80$     |
| C+           | $\geq 74 - < 77$     |
| C            | $\geq 70 - < 74$     |
| C-           | $\geq 65 - < 70$     |

# **Class Engagement:**

---

**Class Discussions:** A large part of the overall learning experience is gained through discussions and participation amongst the class. The intent here is to encourage a meaningful participation during live sessions and through the discussion board. A typical lecture consists of two parts. First part covers new material. Second part details each assignment and every quiz - to reiterate new material. During a live session, when a quiz is introduced, you will be prompted to respond. At this point, right or wrong answer does not count, although your participation is important. Students are also encouraged to post comments toward several predefined topics at blackboard's class discussions. All posts, along with an abundance of references, are propagated from previous classes to supplement the body of knowledge for the whole course.

**Peer Reviews:** All in-process artifacts (e.g. requirements and test cases), as well as the final deliverable (code), shall be peer reviewed. Providing constructive and concise comments is a part of the defined process elaborated in the class. Each student actively participates in peer reviews, submits a predefined deliverable and incorporates changes suggested by peers.

**Attendance Policy:** Your participation is expected at all class meetings. You are responsible for all material discussed in class.

## **Quizzes**

---

It is important to have a certain expectation toward Quizzes that offer an important method to learn new material. Quizzes complement other methods, assignments, discussions, and term project. Quizzes provide a different path and angle toward the same course concepts. You have multiple attempts to take a quiz and the mean score will go toward the grade. Yes&No answers bear no judgment. They are impersonal, which is quite different from assignments, where, in many cases, there is no right or wrong answer.

## **Assignments**

---

Assignments encourage you to extrapolate from the course material and from your own experiences. Researching a topic and responding questions about grey-area-scenarios will force you to think independently. You should time box your research. These assignments illustrate key points. Exploring too many details is bound to deviate from the focus on the big picture.

# Term Project

---

The purpose of the term project is to follow key steps of software product development. It is a chance to pilot and adopt the best practices covered in the course. In today's environment, collaboration is imperative as it drives the overall effectiveness of a software project. After participating in a term project in a controlled environment, you will be able to apply your skills later to a real situation. You are encouraged to benefit from several dozen projects completed at previous classes. Here is the [testimony](#) from a student. I recently completed my Master's degree in 2017, and have very fond memories of CS633 - my final class before graduation. As a Principal Software Developer at Liberty Mutual - CS633 covers a plethora of topics relevant to my everyday work, providing me with the best-practice tools I need to succeed. Most specifically, the term project, which very accurately simulates a real-life product life-cycle, was exceptionally rewarding. My team and I were able to build a cloud-based, production-ready application, all while fine-tuning and reinforcing the learning that coincided in the classroom sessions. I most highly recommend this class, among the many others here at BU Master of Science in Computer Information Systems!

## Final Exam:

---

You will have three hours to complete the final exam; there should be plenty of time. The final exam will be open book, open notes.

- (1) **Short answers:** A combination of multiple choices/answers based on quizzes from course modules.
- (2) **Essays:** Short essays/responses which will focus on in-depth lecture discussions and on practical knowledge gained throughout Term Project.

You should expect no surprises on your final exam, as it includes no questions that are not covered during the class.

The last question on your final exam is as follows. *As one of the goals of this course is for you to adopt at your day job - some techniques we covered in the class. Which of the best practices from the course do you plan to adopt?*

Here is one of the responses. While there are many elements of this course that I expect to directly apply to my current job, there are two that stand out more than others. Those two are peer reviews and estimation techniques. My company performs peer reviews for functional specifications as well as technical designs, but we do not conduct code reviews. I think if we were to implement a static analysis of team's code, we would detect potential defects sooner. We would also have a chance to suggest more uniform coding practices, so there would be a greater consistency of code among developers. It would also aid in cross-training, in which my team is severely lacking.

Here is another response. The tool evaluation matrix and magic quadrant will be extremely valuable to me in the future as I find that the organization I work for tends to acquire a multitude of tools, which do the same exact thing, and it needs to be revised in order to standardize their use to become more efficient and save money at the same time. Thank you for the toolkit you have provided throughout the semester!

## Academic Integrity:

---

Academic conduct in general, and MET College rule in particular, require that all references and uses of the work of others must be clearly cited. All instances of plagiarism must be reported to the College for action. For the full text of the academic conduct code, please check <http://www.bu.edu/met/for-students/met-policies-procedures-resources/academic-conduct-code/>

# Course Map and Schedule:

- The following schedule is tentative, and is a subject to change according to the progress of the class and the feedback from students. It covers six modules with twelve live sessions. Final exam is administered during the last live session. Each project team schedules a final presentation during a separate online session.
- (\*) designates an optional activity that does not contribute to your grade.
- Refer to the detailed *Description of Term Project*. Regular project submissions correspond to the lecture material for that module. For example, the first module covers requirements. Hence the deliverable for the second module is to write project requirements in *Pivotal*. It is advisable to start coding as early as possible. A project team is expected to meet at least weekly. There is a demo expected at a weekly team's meeting of a partially-working system with coded parts - as early as second module. The last sixth module is dedicated to testing and bug fixing, hence coding should be pretty much done by the fifth module.

| Module  | Module Topic   | Reading References   | Discussion Topics<br>Post Comments<br>(*)  | Individual Assignments   | Quizzes | Term Project   | Live Sessions<br>HAR 220   |
|---|--|--|--|--|---------|--|--|
| <b>Module 1</b><br><br><i>DELIVERABLES DUE DATE:</i><br><b>Thursday, June 5,</b><br><b>6:00 AM EST</b>  | Globalization Trends<br><br>Requirements Engineering<br><br>Engineering Management | Module 1 Course Notes<br><br>Karl Wiegers<br><br>Pivotal tutorial<br>GitHub tutorial                                   | Introduce yourself and welcome other students<br><br>Compare Company Policies of Google and GM<br><br>Document a Policy for your Project | <ul style="list-style-type: none"> <li>• A 1 "Provide alternative definitions"</li> <li>• A 2 "Traverse Project Repository"</li> <li>• A 3 "Converse with Chatbot for Most Effective Learning Strategy"</li> </ul> | Quiz 1  | <ul style="list-style-type: none"> <li>• Register with Pivotal</li> <li>• Create GIT account</li> <li>• Propose Project Scope</li> <li>• Propose team's composition</li> </ul> | Thursday, May 22<br>6 pm - 9:30 pm<br><i>Course Lectures</i><br><br>Thursday, May 29<br>6 pm - 9:30 pm<br><i>Assignments &amp; Quizzes</i>   |
| <b>Module 2</b><br><br><i>DELIVERABLES DUE DATE:</i><br><b>Thursday, June 19,</b><br><b>6:00 AM EST</b> | Software Configuration Management (SCM)<br><br>Estimation                          | Module 2 Course Notes<br><br>Steve McConnell<br>Section4 "Cone"<br>Section 11 "Analogy"<br>Section 12.3 "Story Points" | Comment on a paper "The Art of Changing the History"<br><br>Respond the common question, while teaching estimation class                 | <ul style="list-style-type: none"> <li>• A 1 "Analyze Git's Branching and Merging"</li> <li>• A 2 "Motivation Principle"</li> <li>• A3 "Provide consistent assessment of six requirements processes"</li> </ul>    | Quiz 2  | <ul style="list-style-type: none"> <li>• Document Personas</li> <li>• Develop Requirements in Pivotal</li> <li>• Start coding</li> </ul>                                       | Thursday, June 5<br>6 pm - 9:30 pm<br><i>Course Lectures</i><br><br>Thursday, June 12<br>6 pm - 9:30 pm<br><i>Assignments &amp; Quizzes</i>  |
| <b>Module 3</b><br><br><i>DELIVERABLES DUE DATE:</i><br><b>Thursday, June 26,</b><br><b>6:00 AM EST</b> | Agile<br><br>Static Testing  | Module 3 Notes<br><br>Scrum Guide Nov 2020   | Give a hand to a Moderator in a contentious peer review  | <ul style="list-style-type: none"> <li>• A 1 "Maintain Time Trends for Six Teams"</li> <li>• A2 "Calculate All Permutations Of Cost Of Delay"</li> <li>• A3 "Optimize Peer Reviews"</li> </ul>                     | Quiz 3  | <ul style="list-style-type: none"> <li>• Provide Estimation record</li> <li>• Compile CI List</li> <li>• Document Tools Connectivity Diagram</li> </ul>                        | Thursday, June 19<br>6 pm - 9:30 pm<br><i>Course Lectures</i><br><br>Thursday, June 19<br>6 pm - 9:30 pm<br><i>Assignments &amp; Quizzes</i> |

|   |   |   |   |  |               |   |   |
|---|---|---|---|--|---------------|---|---|
| <p><b>Module 4</b></p> <p><i>DELIVERABLES DUE DATE:<br/>Thursday, July 10,<br/>6:00 AM EST</i></p>  | <p>IS Security</p> <p>Elements of Software Design</p> <p>Common Tools Supporting Common Process</p>   | <p>Module 4 Course Notes</p> <p>David Pratt</p> <p>David Kim</p>  | <p>Should Ed Snowden be pardoned?</p> <p>How to manage a repository of best coding practices</p>    | <ul style="list-style-type: none"> <li>• A1 "Tools for Unit Test"</li> <li>• A2 "Position 22 software tools into 11 categories"</li> </ul>   | <p>Quiz 4</p> | <ul style="list-style-type: none"> <li>• Document Selected Use Cases</li> <li>• Converge on a Components Interaction diagram</li> <li>• Document State Transitions</li> </ul> | <p>Thursday, June 26<br/>6 pm - 9:30 pm<br/><i>Course Lectures</i></p> <p>Thursday, July 3<br/>6 pm - 9:30 pm<br/><i>Assignments &amp; Quizzes</i></p>                    |
| <p><b>Module 5</b></p> <p><i>DELIVERABLES DUE DATE:<br/>Thursday, July 24,<br/>6:00 AM EST</i></p>  | <p>System Test</p> <p>Unit Test</p> <p>Continuous Delivery</p>  | <p>Module 5 Course Notes</p> <p>Glenford Myers Chapter 1 Self Assessment Test</p> <p>MIT courseware</p> <p>Jez Humble Ch 5, Deployment Pipeline</p> | <p>Link two diverse notions from Andy Grove and Jez Humble</p>                                      | <ul style="list-style-type: none"> <li>• A1 "Data-driven Test Design"</li> <li>• A2 "Review online courseware"</li> <li>• A3 "Essential Test Coverage Based on State Transitions"</li> </ul> | <p>Quiz 5</p> | <ul style="list-style-type: none"> <li>• Transform mock-ups into wireframes</li> <li>• Complete coding</li> </ul>   | <p>Thursday, July 10<br/>6 pm - 9:30 pm<br/><i>Course Lectures</i></p> <p>Thursday, July 17<br/>6 pm - 9:30 pm<br/><i>Assignments &amp; Quizzes</i></p>                   |
| <p><b>Module 6</b></p> <p><i>DELIVERABLES DUE DATE:<br/>Thursday, August 7,<br/>6:00 AM EST</i></p> | <p>Quality Assurance</p> <p>Process Improvement</p> <p>Course review in preparation to Final Exam</p> | <p>Module 6 Course Notes</p> <p>CMMI, v 1.3 &amp; 2.0</p> <p>ISO 9001 2008</p>  | <p>Adapt &amp; Adopt - outline a path toward an effective institutionalization of a new process</p> | <ul style="list-style-type: none"> <li>• A1 "Learning Platform"</li> <li>• A2 "Test an app"</li> </ul>   | <p>Quiz 6</p> | <ul style="list-style-type: none"> <li>• Develop test cases in a standard format</li> <li>• Reduce data-driven combinations using Allpairs</li> </ul>                         | <p>Thursday, July 24<br/>6 pm - 9:30 pm<br/><i>Course Lectures</i></p> <p>Thursday, July 31<br/>6 pm - 9:30 pm<br/><i>Assignments &amp; Quizzes</i></p>                   |
| <p><b>Final Exam</b></p> <p><i>DELIVERABLE DUE DATE:<br/>Friday, August 8,<br/>6:00 AM EST</i></p>  | <p>Final Exam three-hours open-book</p>   |   |   |  |               | <ul style="list-style-type: none"> <li>• Submit final report for term project</li> </ul>  | <p>Thursday, August 7<br/>6 pm - 9:00 pm<br/><i>Final Exam</i></p> <p><b><i>Friday August 8<br/>Each Team's Presentation is Scheduled Online<br/>For one hour</i></b></p> |