MET CS 633 OL
Software Quality, Testing and Security Management
Spring 1, 2020 - SYLLABUS

Course Description:

Theory and practice of quality assurance, testing and security for each step of the software development cycle. Verification vs. validation. Test case design techniques, test coverage criteria, security development and verification practices, and tools for static and dynamic analysis. Standards. Test-driven development. QA for maintenance and legacy applications. From a project management knowledge perspective, this course covers the methods, tools and techniques associated with the following processes -- Plan Quality, Perform Quality Assurance, and Perform Quality Control. 4 credits.

Course Objectives:

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

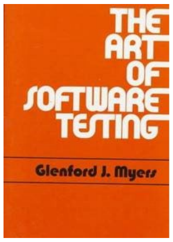
- Manage a Software Quality Assurance function covering all phases of a global product development life cycle
- Play an effective role of a Software Engineering Manager in a context of IEEE CSDP (Certified Software Development Practitioner) with a specific emphasis on organizational policies as guiding 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.

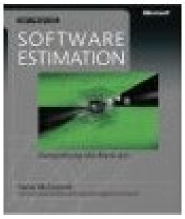
- Structure a project asset library aiming at single-click navigation to a requested artifact.
- Provide leadership to a process program that is mapped into PMI and/or SEI CMMI as an improvement model.

Course Resources:

There are several books referenced throughout the course. These books are optional. 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.

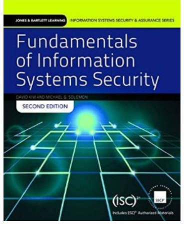


Glenford Myers is an American computer scientist, entrepreneur, and author. He founded two successful high-tech companies, authored eight textbooks in the computer sciences, and made important contributions in microprocessor architecture.

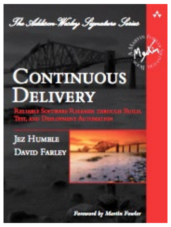


Steve McConnell is a well-known industry practitioner, Chief Software Engineer at Construx. Software Development magazine named Steve one of the three most influential people in software industry along with Bill Gates and Linus Torvalds. Steve's book on estimation is available in electronic form and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

David Platt teaches User Experience Engineering at Harvard University Extension School and at companies all over the world. He's the author of 12 programming books, including Why Software Sucks (Addison-Wesley, 2006) and Introducing Microsoft .NET (Microsoft Press, 2003). Microsoft named him a Software Legend in 2002.



"Fundamentals of Information Systems Security", David Kim and Michael Solomon, ISBN-13: 978-1284031621. David Kim is the president of Security Evolution Inc (SEI) provides consulting services around the world.



Jez Humble is a principle consultant at Thoughtworks helping organizations deliver high quality software fast and reliably. This groundbreaking book sets out the principles and technical practices enabling rapid, incremental software release. It is a part of the sea change that brought about a culture of continuous delivery.

Instructor:

Alex Elentukh
Email: 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 a little 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.

Most importantly, I'm here to help - if you have any questions or need anything at all during the course, please don't hesitate to contact me…the most effective way is via email.

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.

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

The following table details the translation on a numerical grade to a letter grade.

LETTER GRADE	NUMERICAL GRADE
A	>=95 - <=100
A-	>=90 - <95
B+	>=85 - <90
B	>=80 - <85
B-	>=77 - <80
C+	>=74 - <77
C	>=70 - <74
C-	>=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 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: Attendance is **optional** but encouraged at Live Classroom sessions. You are responsible for all material discussed in class.

Quizzes

Quizzes in this course are an important method to learn new material. Quizzes complement other methods, assignments, discussions, and term project. Quizzes offer a slightly 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 big picture focus.

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 code, we would detect potential defects sooner. We would also have a chance to suggest more uniform coding best 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 two live sessions per module.

– 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. So 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 meeting of a project team of a partially-working system with coded modules and prototypes - as early as second module. The last module is dedicated to testing and bug fixing, hence coding should be pretty much done by the module previous to last.

Module	Module Topic	Reading References	Discussion Topics Post Comments	Individual Assignments	Quizzes	Term Project	Live Classroom Zoom
Module 1 ***DELIVERABLES DUE DATE:*** ***Monday, Jan 27, 6:00 AM EST***	Definition and scope of Quality Assurance Globalization Trends Requirements Engineering Engineering Management	Module 1 Course Notes Karl Wiegers Pivotal tutorial GitHub tutorial	Introduce yourself and welcome other students Compare Company Policies of Google and GM Document a Policy for your Project	• A 1 "Provide alternative definitions" • A 2 "Improve one requirement" • A 3 "Motivation Principle"	Quiz 1	• Register with Pivotal • Create GIT account • Propose Project Scope • Propose team's composition	Tuesday, Jan 21 9-10 pm EST ***Course Lectures*** Thursday, Jan 23 9-10 pm EST ***Assignments & Quizzes***
Module 2 ***DELIVERABLES DUE DATE:*** ***Monday, Feb 3, 6:00 AM EST***	Software Configuration Management (SCM) Estimation	Module 2 Course Notes Steve McConnell Section4 "Cone" Section 11 "Analogy" Section 12.3 "Story Points	Comment on a paper "The Art of Changing the History", see Assignment 1, Question 2 Respond the common question, while teaching estimation class	• A 1 "Branching and Merging" • A 2 "Select estimation strategy" • A3 "Select the Best PM"	Quiz 2	• Document Personas • Develop Requirements in Pivotal • Start coding	Tuesday Jan 28 9-10 pm EST ***Course Lectures*** Thursday Jan 30 9-10 pm EST ***Assignments & Quizzes***
Module 3 ***DELIVERABLES DUE DATE:*** ***Monday, Feb 10, 6:00 AM EST***	Agile Static Testing	Module 3 Notes Scrum Guide Nov 2017	Give a hand to a Moderator in a contentious peer review	• A 1 "Sequence Stories" • A 2 "Analyze Velocities" • A3 "Write critical comments in a predefined format"	Quiz 3	• Provide Estimation record • Compile CI List • Document Tools Connectivity Diagram	Tuesday, Feb 4, 9-10 pm EST ***Course Lectures*** Thursday, Feb 6, 9-10 pm EST ***Assignments & Quizzes***

Module 4 ***DELIVERABLES DUE DATE: Monday, Feb 17, 6:00 AM EST***	IS Security Elements of Software Design Common Tools Supporting Common Process	Module 4 Course Notes David Pratt David Kim	Should Ed Snowden be pardoned? How to manage a repository of best coding practices	- A1 "Arrive at a decision based on Cost Of Delay" - A2 "Position 18 software tools into 9 categories"	Quiz 4	- Document Selected Use Cases - Converge on a Components Interaction diagram - Document State Transitions	Tuesday Feb 11 9-10 pm EST ***Course Lectures*** Thursday Feb 13 9-10 pm EST ***Assignments & Quizzes***
Module 5 ***DELIVERABLES DUE DATE: Monday, Feb 24, 6:00 AM EST***	System Test Unit Test Continuous Delivery	Module 5 Course Notes Glenford Myers Chapter 1 Self Assessment Test Jez Humble Ch 5, Deployment Pipeline	Link two diverse notions from Andy Grove and Jez Humble	- A1 "Analyze architecture of a test case repository" - A2 "Data-driven Test Design" - A3 "Essential Test Coverage Based on State Transitions"	Quiz 5	- Transform mock-ups into wireframes - Complete coding	Tuesday Feb 18 9-10 pm EST ***Course Lectures*** Thursday, Feb 20 9-10 pm EST ***Assignments & Quizzes***
Module 6 ***DELIVERABLES DUE DATE: Monday, March 2, 6:00 AM EST***	Quality Assurance Process Improvement Course review in preparation to Final Exam	Module 6 Course Notes CMMI, version 1.3 ISO 9001 2008	Outline a path toward an effective adoption of a new process	- A1 "Position a test into a context" - A2 "Forecast Defects" - A3 " SQA Application of ML"	Quiz 6	- Develop test cases in a standard format - Reduce data-driven combinations using Allpairs	Tuesday Feb 25 9-10 pm EST ***Course Lectures*** Thursday, Feb 27 9-10 pm EST ***Assignments & Quizzes***
DELIVERABLES DUE DATE: Tuesday, Mar 3, 6:00 AM EST	Study Period			- A1 "Optional Assignment: Release to Regression" - A2 "Optional Assignment: Outline Test Cases" - A3 " Optional Assignment: SOA facilitates CD " - A4 " Optional Assignment: Practice Assessment"		- Present basic functionality of Term Project - Submit final report as an assignment	Tuesday March 3 9-10 pm EST ***Q&A***
Final Exam **Wed, Mar 4 - Sat, Mar 7**							