MET CS 673: Software Engineering
MET CS 473: Introduction to Software Engineering

Fall 2024, Section A1
Tuesday, 18:00 – 20:45, KCB 107

Bora Eryilmaz, Ph.D.

beryilma@bu.edu
LinkedIn Profile

Course website
Blackboard website

Office Hours: Before/after class or by appointment
(Last revised on August 18, 2024)

Course Description: This course is an overview of modern techniques and tools for developing high-quality software. Topics that you will learn about include software development lifecycle management, software project planning, requirements analysis, software process frameworks, software design & architecture, code refactoring, software testing, and software integration & delivery.

The course features a semester-long *group project* where you will design and develop a software application using modern software development methodologies such as Agile and various software engineering tools, including project management & communication tools, programming frameworks, unit & system testing tools, integration & deployment tools, and version control tools.

Prerequisites: This is a *capstone course* to be taken after at least two programming intensive courses toward the end of a program of study. Familiarity with object-oriented design concepts and proficiency in at least one high-level programming language is required. Familiarity with web, desktop, or mobile application development is preferred.

Suggested Textbooks (not required):

- Roger Pressman and Bruce Maxim, *Software Engineering: A Practitioner's Approach*. 9th Edition. McGraw Hill, 2020.
- David Kung, *Software Engineering*. 2nd Edition. McGraw Hill, 2024.
- Eric J. Braude and Michael E. Bernstein. *Software Engineering: Modern Approaches*. 2nd Edition. Waveland Press, 2011.

Course Objectives: At the completion of this course, you will have better skills to:

1. Follow major software development process steps as they apply to professional software development projects.

2. Apply modern software development techniques and best practices in real-world, collaborative projects to develop high-quality software systems on time and within budget.

3. Use various software engineering tools including project management and collaboration tools, software design tools, programming tools, testing tools, and version control tools.
4. Communicate more effectively with your team members and customers.
5. Present clearly software projects in both the oral and written forms.

Credit Hours: 4

Grading and Assignments: Your grade in this class will be based on class participation (including your individual project contributions), lab assignments, quizzes, project presentations, the success of your group project (including code, tests, design documents, documentation, and status reports), and a final exam.

The grade breakdown is shown below. All percentages are approximate and the instructor reserves the right to make necessary changes. The group project success score will depend on the instructor's subjective assessment as well as the relative success of your project against all other projects in the course.

Class participation	5%
Quizzes (3 quizzes)	10%
Lab assignments (3 labs)	10%
Individual project contribution	10%
Project status reports (weekly)	15%
Presentations (mid-semester and final)	15%
Group project assessment	15%
Final exam	20%

The letter vs. numerical grade conversion is shown below.

A	[95-100]	A-	[90-95)		
B+	[85-90)	B	[80-85)	B-	[77-80)
C+	[74-77)	C	[70-74)	C-	[65-70)
D	[60-65)	F	[0-60)		

- **Quizzes**
  - There will be 3 quizzes, which should all be submitted directly to Blackboard.
  - You are expected to work independently on quizzes.
  - **Late submissions will incur a 33% penalty per day**.
- **Labs Assignments**
  - There will be 3 lab assignments, which should all be submitted directly on Blackboard.
  - You are expected to work independently on lab assignments.
  - **Late submissions will incur a 33% penalty per day**.
- **Individual Project Contributions**
  - The weekly project status report, which is described below, is required to contain a short section by each project team member, detailing their individual contribution to the project for the week in question.

- **Group Project**
    - This course features a semester-long group project. Each team will have 4–6 members. Every member is expected to contribute an equal share to the project.
    - You are expected to work as a team on your group project, each of you performing various roles over the course of the semester.
    - You will be graded on the overall success of the group project as well as your individual contributions to it. The project success will be evaluated based on the completeness, correctness, complexity, and quality of your final product (including source code, tests, design documents, and product documentation).
    - Each team is required to submit a Project Proposal (as a PDF or Word file) that should be approved by the instructor before proceeding with project development.
    - Each team is required to submit a weekly project status report (as a PDF or Word file) describing the weekly progress of the project, including all notes from your weekly team meetings. You will also need to include all relevant design documents along with the status report.
    - Each team will give a short mid-semester and a final presentation of their project.
    - All project artifacts should be made available on GitHub (code files, test files, design documents, product documentation, etc.). Your weekly project reports should contain a link to your GitHub project site.

Course Policies:

- **General**
    - Note that this is a live document and is subject to change according to the progress of the class and your feedback. Remember to check the syllabus for updates occasionally.
    - **Late quiz, lab, and report submissions will incur a 33% penalty per day**.
- **Attendance and Absences**
    - Attendance and class participation is expected. Your "Class participation" grade will be based on your active involvment during the class; just coming to the class is not sufficient.
    - Students are responsible for all material covered in the lectures.
- **Academic Conduct Code**
    - Cheating and plagiarism will not be tolerated in any Metropolitan College course. They will result in no credit for the assignment or examination and may lead to disciplinary actions. Please take the time to review the Student Academic Conduct Code.
    - Use of AI tools such as ChatGPT is strictly prohibited for creating any **original content** for labs, reports, project artifacts, etc. You are allowed to use AI tools to **improve** the content that you have first created yourself. However, you must explicitly indicate any such use in your labs, reports, etc. Any other use of AI tools in your submissions will be treated as plagiarism.

Course Outline: The weekly coverage might change as it depends on the progress of the class. However, you must keep up with the reading assignments and all labs, quizzes, and project reports.

You can find the detailed discussion topics for each week, the reading assignments, and the due dates on the Blackboard site for the course.

Week	Topics	Due
Week 1 (09/03)	• Software Development Lifecycle	
Week 2 (09/10)	• Software Processes and Configuration Management	• Group project proposal (draft)
Week 3 (09/17)	• Software Project Management	• Group project proposal (final) • Lab 1 (GitHub)
Week 4 (09/24)	• Requirements Analysis	• Project status report 1
Week 5 (10/01)	• Software Design	• Project status report 2 • Lab 2 (Requirements Analysis)
Week 6 (10/08)	• Mid-semester project presentations	• Presentation slides
Week 7 (10/15)	• No class - Substitute Monday schedule	• Quiz 1 (all material)
Week 8 (10/22)	• Software Architecture	• Project status report 3

Week	Topics	Due
Week 9 (10/29)	• User Experience (UX) and UI/API Design	• Project status report 4
Week 10 (11/05)	• Software Implementation	• Quiz 2 (Design, architecture, UX/UI/API)
Week 11 (11/12)	• Software Testing: Unit Testing • Zoom class (visit Blackboard for link)	• Project status report 5
Week 12 (11/19)	• Software Testing: Integration and System Testing	• Lab 3 (Unit Testing)
Week 13 (11/26)	• Software and Digital Accessibility	• Project status report 6
Week 14 (12/03)	• Final project presentations	• Presentation slides • Quiz 3 (Accessibility)
Week 15 (12/10)	• No class - Study Period	
Week 16 (12/17)	• Final Exam (In-class)	