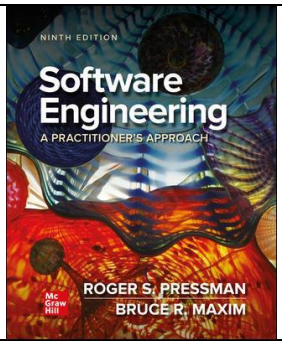
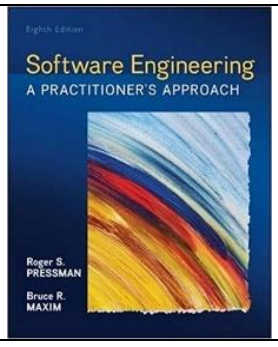
Syllabus for CSC-382
Software Engineering
Summer 2021 | June 28th – August 13th
Three Credit Hours

Instructor: John Hart
Course Location: Zoom Classrooms (Remote due to COVID-19)
Course Times: No Set Timeframe – **Weeks are Mon (12:00AM) - Mon (11:59PM)**
Email: johnha@umich.edu
Phone: (517) 599-4829 (text message preferred)
Office Location: Zoom Office Hours (Remote due to COVID-19)
Office Hours: Mondays from 5:00PM - 6:00PM (Primarily by Appointment)

Prerequisites:
Admission to the CSEP's CS/IS Undergraduate Program

Course Description:
Topics will include software engineering; design methods including formal methods, UML modeling, component-based design, with overview of patterns and frameworks, and architectural-based designs. Modern software processes such as Extreme Programming and Cleanroom software development. Issues and problems associated with large-scale software project failures and techniques for preventing them.

Textbook: (Choose One (1) from Recommended Textbooks)

Software Engineering- A Practitioner's Approach by Pressman, 9th edition (Recommended – option 1) ***Source of Quiz Questions***	*Software Engineering- A Practitioner's Approach* by Pressman, 8th edition (Recommended – option 2)
	

Course Information:
This course provides an overview of software engineering and UML Modelling concepts. It focuses on how software engineering processes and methodologies can be utilized to design, manage, and support effective software applications. The goal is to provide a comprehensive perspective of the roles that software engineering and enterprise

architecture can be utilized to manage and control the development of a software product. Topics discussed are drawn from a variety of sources and are intended to provide the students with the knowledge of a variety of software engineering process models, techniques and standards utilized for software development.

Course Objectives:
The course objective is to provide each student with the management techniques and skills required to be an effective software engineer. As a future engineering professional, you have a responsibility to know what the latest industry standards and software engineering processes are and how they can be successfully applied. The objectives of this course are as follows:

- Provide understanding of software engineering processes.
- Enable you to clearly articulate the key issues associated with managing a software project.
- Give you an in-depth understanding of Object Modeling and UML
- Explore Component-based Software Engineering
- Provide you with a detailed understanding of Design Patterns
- Clearly understand the role of architecture in software design.
- Introduce Formal Methods in Software Engineering
- Provide you an understanding of software project management skill.

Class Format:
The class will provide an active learning environment for everyone. Traditionally, this class would be divided into two sessions. The initial session will consist of a traditional lecture, as well as interactive discussions. The second session will be used to break out and work in groups to reinforce the concepts covered in the lecture portion of the class period. **This semester**, the student/group will be required to practice time-management to ensure all group-related work is completed on time (per the syllabus below). Please make time to meet as a group to discuss the relevant topics to your group exam and share feedback with your team members. <u>All group participation is optional – do not feel pressured into joining a group if you do not wish to do so.</u>
Homework: Group work allowed, relating to topics from the lectures and textbook/Supplemental reading materials - <u>Short essay (2-4 paragraphs required)</u>. **Group performance summary required if work is done in a group (last paragraph).**
Quizzes: Individual work only; multiple-choice tests from the textbook (Pressman 9th).
Final Exam: Group work allowed, diagramming and long-answer test relating to topics from the lectures and textbook readings. **Group performance summary required if work is done in a group (last paragraph).**

Assessment Mechanisms:
- Your understanding of the course objectives will be assessed by the following deliverables:
    - Homework Submissions – 100 points possible (34% of final grade)
    - Weekly Quizzes – 70 points possible (23% of final grade)
    - Final Exam 130 points possible (43% of final grade)

Other Good Reference Books (likely referenced in discussion):

- *Software Engineering* by Sommerville
- *Object-oriented Software Engineering* by Jacobson
- *Use Cases – requirements in context* by Kulak and Guiney
- *Writing Effective Use Cases* by Cockburn
- *Using CRC Cards* by Wilkinson
- *The Object Primer* by Ambler
- *Design Patterns Explained* by Shallow and Trott
- *Software Architecture* by Shaw and Garlan

Other Selected Readings:

Articles that you **must** read throughout the semester (provided on Blackboard):

- R. R. Loka, "Software development: What is the problem?", *IEEE Computer*, pp. 110-112, Feb 2007.
- C. Symons, "Measuring software industry performance: What you measure is what you get", *IEEE Software*, vol. 27, no.6, pp. 66-72, 2010.
- V. Rajlich, "Changing the paradigm of software engineering", *Comm. of the ACM*, vol. 49, no. 8, pp. 67-70, 2006.
- L. Cao and B. Ramesh, "Agile requirements engineering practices: An empirical study", *IEEE Software*, vol. 25, no.1, pp. 60-67, 2008.
- P. Denning and R. Riehle, "Is software engineering engineering?", *Comm. of the ACM*, vol. 52, no. 3, pp. 24-26, 2009.
- B. Dobing and J. Parsons, "How UML is used", *Comm. of the ACM*, vol. 49, no. 5, pp. 109-113, 2006.
- D. Harel, "State charts in the making: A personal account", *Comm. of the ACM*, vol. 52, no. 3, pp. 67-75, 2009.

Recommended articles to read throughout the semester:

- Zachman and Systems related papers
  - G. Booch, "Enterprise architecture and technical architecture", *IEEE Software*, vol. 25, no.1, pp. 54-59, 2010.
  - Zachman, "Architecture is architecture is architecture", Zachman International Website, 2007.
  - M. Varga, "Zachman framework in teaching information systems", *Proc. 25th International Conf. on Information Technology Interfaces*, pp. 161-166, 2003.
  - G. Lewis, E. Morris, S. Simanta, and D. Smith, "Service orientation and systems of systems", *IEEE Software*, vol. 28, no.1, pp. 58-63, 2011

- Architecture-related papers
  - P. Avgeriou and U. Zdun, "Arcthitectural patterns revisited-A pattern language", *Proc. 10th European Conf. on Pattern Languages of Programs*, pp. 431–470, 2005.
  - P. Krutchen, H. Obbink, and J. Stafford, "The past, present and future of software architecture", *IEEE Software*, pp.22-30, March/April 2006.
  - J. Tyree and A. Akerman, "Architecture decisions: Demystifying architecture", *IEEE Software*, pp. 19-27, March/April 2005.

- Reuse and Component-based development-related papers
  - M. Sparling, "Lessons learned through six years of component-based development", *Comm. of the ACM*, vol. 43, no. 10, pp. 47-53, 2000.
  - M. Ramachandran, "Software reuse guidelines", ACM SIGSOFT Software Engineering Notes, Vol. 30, No. 3, 2005.
  - C. P. Ayala, D. S. Cruzes, Ø. Hauge, and R. Conradi, "Five facts on the adoption of open source software", *IEEE Software*, vol. 28, no.2, pp. 95-99, 2011

- Frameworks and Design patterns-related papers
    - F. Buschmann, K. Henney, and D. Schmidt, "Past, present and future trends in software patterns", *IEEE Software*, pp. 31-37, July/Aug 2007.

- Formal methods-related papers
    - P. Lindsay, "Introduction to formal methods", *Technical report 98-25*, School of Information Technology, University of Queensland, Australia, 1998.

- Test and Validation
    - H.M. Olague, L. H. Etzkorn, S. Gholdston, and S. Qattlebaum, "Empirical Validation of Three Software Metrics Suites to Predict Fault-Proneness of Object-Oriented Classes Developed Using Highly Iterative or Agile Software Development Processes", *IEEE Trans. On Software Engineering*, vol. 33, no. 6, pp. 402-419, 2007.
    - R. C. Martin and G. Melnik, "Tests and requirements, requirements and tests: A mobius strip", *IEEE Software*, vol. 25, no.1, pp. 54-59, 2008.

Tools:

- Visual Paradigm. Eval. licenses available - Personal copies may be purchased (available at Visual Paradigm).
- *Visible Analyst*. Personal copies may be purchased (available at Visible Analyst for $49 student pricing).
- *Star UML*. Eval licenses available (available at Star UML for $49 student pricing).

Grading:

- Homework, Quizzes, and Final Exam – generally combine to 300 points total for 100%.
    - Homework: 100 pts, Quizzes: 70 pts., and Final Exam: 130 pts
    - Each student will be pursuing 291+ points to achieve an A+ in the course. **NOTE**: No curving will be provided, please do not ask.
    - Class participation is an expected portion of the course work and may be used to improve (round up) a final score.
- Grading Scale:

A's	B's	C's	E-grade
97-100: A+	87-89: B+	75-79: C+	65 and Below: N
92-96.9: A	82-86.9: B	72-74.9: C	
90-91.9: A-	80-81.9: B-	70-71.9: C-	

Detailed Guidelines and Rubrics for Grading of Homework:

NOTE: For all written essay-type homework in this class, a page is single space, 12-point font, with one inch or less side margins, and 1.5 inch or less top and bottom margins. Write in complete sentences and provide references as needed. DO NOT SIMPLY GOOGLE ANSWERS AND CUT AND PASTE AS THAT IS PLAGIARIZING!!!

Homework Essay Question Grading Rubric (5 points possible)		
Area of Concern	**Levels of Performance**	**Points Awarded**
Grammar and Organization	Answer is easy to read.	1
	Answer is difficult to read due to grammatical errors and/or lack of organization.	0
Completeness	Answer should leave no 'why questions' and/or no missing technical content.	2
	Answer leaves some why questions and/or some missing technical content.	1
	Question is not fully answered.	0
Accuracy	All statements are accurate.	2
	Some statements are inaccurate.	1
	Mostly or completely inaccurate.	0

Homework Diagram Question Grading Rubric (5 points possible)		
Area of Concern	**Levels of Performance**	**Points Awarded**
Diagram Organization	Diagram is professionally drawn.	1
	Diagram is sloppily drawn.	0
Completeness	Diagram fully captures the essence of the problem.	2
	Diagram misses some of the essence of the problem.	1
	Diagram misses considerable portions of the essence of the problem.	0
Accuracy	All structural aspects of diagram accurate.	2
	Some structural aspects of diagram are inaccurate.	1
	Structural aspects of diagram mostly or completely inaccurate.	0

NOTE: For all diagrammatic homework assignment answers, the diagrams must be done in an electronic tool (PowerPoint can be used as last resort). Any hand-written assignments will receive zero points.

Attendance:

- Regular class attendance is highly recommended unless student has informed the instructor that they are a distance learner. Prior approval to be a distance learner is required from the instructor. **Note**: International students in the U.S. may not take the class as distance learners due to U.S. government F-1 visa requirements.

Cyber classroom:

- The videos for the classes can be found on:
  Blackboard

Late Homework Policy:
There will be ***no*** late homework accepted - homework is due by end of class. Once class ends, the blackboard submission link will close, and assignments will not be accepted.

Academic Misconduct:

- Copying others work, plagiarizing external references without giving due credit, and cheating in exams are strictly forbidden. The following is extracted from the UM-Flint Course Catalog:
- Intellectual integrity is the most fundamental value of an academic community. Students and faculty alike are expected to uphold the highest standards of honesty and integrity in their scholarship. No departure from the highest standards of intellectual integrity, whether by cheating, plagiarism, fabrication, falsification, or aiding and abetting dishonesty by another person, can be tolerated in a community of scholars. ***Such transgressions may result in action ranging from reduced grade or failure of a course to expulsion from the University or revocation of degree.***
- Please take pride in your own work, and feel free to ask me if you need help or assistance to maximize your learning. As shown above, the consequences for not doing your own work can be severe.

Any questions? Visit: Academic Integrity Worksheet

Accessibility and Disability Statement:
The University of Michigan–Flint strives to make learning experiences as accessible as possible and complies with Section 504 of the Rehabilitation Act of 1973 and the American with Disabilities Act. The university provides individuals with disabilities reasonable accommodations to participate in educational programs, activities, and services. Students with disabilities requiring accommodations to participate in class activities or meet course requirements must self-identify with Disability and Accessibility Support Services as early as possible at (810) 762-3456 or dassflint@umich.edu. The office is located at 264 University Center, inside the CAPS Office. Once your eligibility for an accommodation has been determined you will be issued an Accommodation Letter. Please present this letter to each faculty member in each class at the beginning of the term, or at least two weeks prior to the need for the accommodation (test, project, etc.).

Planned Schedule: (Note this schedule is subject to change)

Week	Topics
Week 1 (6/28/21)	**Lecture:** Class Introduction, Syllabus Review, and Overview of Software Engineering. **Read (before class):** - Pressman 8th: Chapters 1 and 2 - Pressman 9th: Chapter 1
Week 2 (7/5/21)	**Lecture:** Waterfall and Agile Software Development Methodologies **Read (before class):** - Pressman 8th: Chapters 4 and 5 - Pressman 9th: Chapter 2, 3 and 4 - Loka - Denning and Riehle **Quiz Due: Q1 by 11:59PM (Assigned 6/28/21 – Pressman 9th – Ch. 1)**
Week 3 (7/12/21)	**Lecture:** Systems Engineering & Business Process Modeling (Zachman) and Object-oriented Analysis I – UML Overview & Use Cases **Read (before class):** - Zachman (**critical**) - Pressman 8th: Chapters 7 and 8 - Pressman 9th: Chapters 5, 6, and 7 - Booch - Varga - Lewis, et al. - Dobing and Parsons **Quiz Due: Q2 by 11:59PM (Assigned 7/5/21 – Pressman 9th – Chs. 2, 3, and 4)** **Homework Due: HW1 by 11:59PM (Assigned 7/5/21)**

Week 4 (7/19/21)	**Lecture:** Object-oriented Analysis II – Activity Diagrams, Swim Lanes & CRC Cards **Read (before class):** - Pressman 8th: Chapter 9 - Pressman 9th: Chapter 8 (8.3 and 8.5) **Quiz Due: Q3 by 11:59PM (Assigned 7/12/21 – Pressman 9th – Chs. 5, 6, and 7)** **Homework Due: HW2 by 11:59PM (Assigned 7/12/21)**
Week 5 (7/26/21)	**Lecture:** Object-oriented Analysis III – Class Diagrams & Associations, Aggregations, Dependencies and Testing Principles **Read (before class):** - Pressman 8th: Chapter 10, 22, 23, and 24 - Pressman 9th: Chapter 7, 8, 19, and 20 - Martin, et al. **Quiz Due: Q4 by 11:59PM (Assigned 7/19/21 – Pressman 9th – Ch. 8)** **Homework Due: HW3 by 11:59PM (Assigned 7/19/21)**
Week 6 (8/2/21)	**Lecture:** Object-oriented Analysis IV – Sequence Diagrams & State Diagrams and Designing for Reuse & Component-based Development **Read (before class):** - Pressman 8th: Chapters 11 and 14 - Pressman 9th: Chapters 8 (8.4 and 8.5) and 11 - Sparling - Ayala, et al. - Ramachandran **Quiz Due: Q5 by 11:59PM (Assigned 7/26/21 – Pressman 9th – Chs. 7 and 8)** **Homework Due: HW4 by 11:59PM (Assigned 7/26/21)**

Week 7 (8/9/21)	**Lecture:** Intro to Architecture, Frameworks, and Design Patterns **Read (before class)** - Pressman 8th: 13 and 16 - Pressman 9th: 10, 14 - Avgeriou and Zdun - Tyree - Buschmann, et al. **Quiz Due: Q6 by 11:59PM (Assigned 8/2/21 – Pressman 9th – Chs. 8 and 11)** **Homework Due: HW5 by 11:59PM (Assigned 8/2/21)**
Week 8 Final Exam (8/16/21)	**Quiz Due: Q7 by 11:59PM (Assigned 8/2/21 – Pressman 9th – Chs. 10, 14, 19, and 20)** **Final Exam (take home)** – Assigned: 8/2/21 @ 11:59PM | Due: 8/16/21 by 11:59PM