SWE 795: Software Engineering Environments, Spring 2017
Tuesdays 4:30 - 7:10, 2026 Art and Design Building

Syllabus and Schedule

Instructor: Prof. Thomas LaToza
ltatoza@gmu.edu (https://mail.google.com/mail/?view=cm&fs=1&tf=1&to=ltatoza@gmu.edu)
http://cs.gmu.edu/~ltatoza
Twitter: @ThomasLaToza (https://twitter.com/thomasltatoza)
Office: 4431 Engineering Building; (703) 993-1677
Office Hours: Anytime electronically, Wed 1:30 - 3:00, or by appointment

Required Textbooks
None

Objective and Learning Outcomes
This course will explore the nature of software engineering environments, investigating programming tools and development environments intended to support programming activities. The focus will primarily be on individual software development, such as what is known about tools for debugging, reuse, program synthesis, refactoring, and managing crosscutting concerns. The course will consider questions such as What makes debugging hard?, What causes defects?, How do developers check if code works?, and explore how tools have been designed to better support these activities. The course will survey the science of studying programming activities and the design of tools designed to help developers work with code more easily and successfully. This will include what is sometimes called "Empirical Studies of Programmers" and the "Psychology of Programming".

At the end of this course, you should be able to:
(1) Conduct a small study of software development practice
(2) Design a development environment feature to address a challenge software developers face
(3) Implement a prototype of a development environment feature within a modern development environment
(4) Evaluate a development environment feature using a small think-aloud usability study

Readings
We will have weekly readings from relevant research papers. All readings will be posted in the schedule below. Each week, every student will be responsible for reading all 3 papers and writing a brief answer to one of several prompts on Piazza before class.

Additionally, each student will be responsible for being the designated Discussant for one paper once every two weeks, serving as discussant for a total of 6 papers over the semester. For each paper, the discussant will be responsible for giving a short 5 min presentation briefly summarizing the paper and facilitating 10 mins of discussion about the paper with the class.

Project
The homework in this course will be in the form of a project. All project work will occur in two person groups. Rather than creating a written report, each HW assignment be take the form of an in-class presentation, where all groups members will give a 10-min presentation on their work.

HW0: Project Proposal (50 points)
The project proposal should describe a specific aspect of software development that your project will focus on. The project proposal should clearly identify a specific challenge software developers experience in programming work, including a scenario describing a situation a developer might face. The project proposal should also include (1) a brief description of the type of study you will perform to understand this challenge better and (2) an initial idea of how a tool might address this challenge.
HW1: Study of Current Practice (100 points)
The study of current practice will be a small study examining a specific challenge software developers face in their programming work. Several types of study are possible. You might choose to examine posts on StackOverflow or another online repository. You might choose to perform a simple think aloud usability study and observe 2 or 3 participants perform a programming task. Or you might choose to survey professional software developers about their activities. In any case, the outcome of the study should be a better understanding of a challenge that software developers face in their programming work.

HW2: Tool Sketch (100 points)
Based on the challenge identified in HW1, you should create a sketch of a potential solution. Your sketch should not contain any implementation of your tool. Instead, the sketch should provide a storyboard, depicting a series of screenshots describing the behavior of your tool on two or more examples. Additionally, a description and high-level overview of the tool’s design and implementation should be included.

HW3: Tool Prototype (250 points)
Based on your sketch, you will implement a small prototype of your tool, extending an existing development environment such as Eclipse, Visual Studio, WebStorm, or seeCode.run to implement your idea.

HW4: Tool Evaluation (100 points)
To understand the effect of your tool on addressing a challenge programmers face, you will conduct a small study evaluating your tool with a few participants.

Resources
This course will use Piazza for posting the schedule and all assignments and announcements. Additionally, we will use Piazza for a discussion board. Grades will be available through Blackboard.

Makeups
As much of the course work will consist of in-class presentations, it will not be possible to submit HW assignments late. HW assignments submitted late will receive a zero. If you will be unable to attend class on the date of a HW assignment, please contact the instructor about this as early as possible.

Grading
Paper responses: 20%
Paper discussant: 20%
Project: 60%

Honor Code
GMU is an Honor Code university; please see the Office for Academic Integrity for a full description of the code and the honor committee process, and the Computer Science Department’s Honor Code Policies regarding programming assignments. The principle of academic integrity is taken very seriously and violations are treated gravely. What does academic integrity mean in this course? Essentially this: when you are responsible for a task, you will perform that task. When you rely on someone else’s work in an aspect of the performance of that task, you will give full credit in the proper, accepted form. Another aspect of academic integrity is the free play of ideas. Vigorous discussion and debate are encouraged in this course, with the firm expectation that all aspects of the class will be conducted with civility and respect for differing ideas, perspectives, and traditions. When in doubt (of any kind) please ask for guidance and clarification.

Accommodations for Disabilities
If you have a documented learning disability or other condition that may affect academic performance you should: 1) make sure this documentation is on file with Office for Disability Services (SUB I, Rm. 4205; 993-2474; http://ods.gmu.edu) to determine the accommodations you need; and 2) talk with me to discuss your accommodation needs.

Privacy
Students must use their MasonLIVE email account to receive important University information, including messages related to this class. See http://masonlive.gmu.edu for more information.

Other Useful Campus Resources
Writing Center: A114 Robinson Hall; (703) 993-1200; http://writingcenter.gmu.edu
University Libraries: Ask a Librarian
Counseling and Psychological Services (CAPS): (703) 993-2380; http://caps.gmu.edu/
University Policies: The University Catalog, is the central resource for university policies affecting student, faculty, and staff conduct in university academic affairs. Other policies are available at http://universitypolicy.gmu.edu/. All members of the university community are responsible for knowing
Tentative Schedule

1. Course Overview and Conducting Studies (1/24)

Assigned readings: none

2. Analyzing Data (1/31)

Assigned readings:

HWs: HW0 due

3. Information Needs (2/7)

Assigned readings:

HWs: none

4. Debugging (2/14)

Assigned readings:

HWs: HW1 due

5. Crosscutting Concerns (2/21)

Assigned readings:

HWs: none

6. Mental Models (2/28)

Assigned readings:

HWs: HW2 due

7. Software Visualization (3/7)
8. Editing Code (3/21)

Assigned readings:

9. Preventing Defects (3/28)

Assigned readings:

10. Reuse (4/4)

Assigned readings:

11. Program Synthesis (4/11)

Assigned readings:

12. Visual Programming Languages (4/18)

Assigned readings:
Object-oriented programming systems and applications, 639-658.

HWs: HW3 due


Assigned readings:

HWs: none

14. Learning Programming (5/2)

Assigned readings:
M. J. Lee et al., "Principles of a debugging-first puzzle game for computing education (https://doi.org/10.1109/VLHCC.2014.6883023)," Symposium on Visual Languages and Human-Centric Computing, 2014, pp. 57-64.

HWs: HW4 due