Software Engineering
COMP 4081, Fall 2025

Tuesday, Thursday 2:40-4:05 p.m.
Dunn Hall 233

https://memphis.instructure.com/courses/168864

Instructor: Kathryn Bridson <kbridson@memphis.edu>
Office Hours: In person: Mon 10-11am, Th 1-2pm, Fri 10-11am
Use course Teams to receive help outside of in person hours
Office: Dunn Hall 303; meetings held in Teams by appointment

Teaching Assistant: Alex Illev <Alex.Iliev@memphis.edu>

**Please send all emails to all instructors and TAs, and reply-all to all emails.
Please begin all email subject lines with the course number in brackets, then
your topic (e.g., [COMP 4081] Missing class 9/7).**

1 Catalog Description

COMP 4081 - Software Engineering (3)
*Scope of software engineering; software life cycle models; software process;
team organization; requirements analysis and design methodologies; metrics,
inspections, testing strategies, and maintenance; software risks; professional
and ethical responsibilities. Computer Science majors should plan to take COMP
4882 during the following spring semester. It is recommended that students take
COMP 3115 before taking this course. PREREQUISITE: COMP 2150, permission of
instructor.*

2 Topics

The course will emphasize the following topics (a subset of the knowledge areas in the *Guide to the Software Engineering Body of Knowledge*):

- **Software Requirements**, including elicitation, specification, and validation
- **Software Design**, including principles, methods, patterns, and notations
- **Software Construction**, including tools, platforms, and APIs
- **Software Testing**, including levels, techniques, and measures
- **Software Maintenance**, including types and cost
- **Software Configuration Management**, including version control
- **Software Engineering Management**, including planning and estimation
- **Software Engineering Process**, including life-cycle models and assessment
- **Software Engineering Models and Methods**, including information modeling, analysis, and Agile methods
- **Software Security**, including access control policies, constructing for security best practices, security testing methods and tools (SAST, DAST)

3   Course Outcomes

The following are some key learning goals that students in the course are expected to achieve:

1. Use modern software development tools (e.g., Git) for effective configuration management.
2. Design and model software using modern modeling languages and notations (e.g., UML).
3. Convey ideas orally through presentations to peers, clients, and faculty.
4. Gather and specify software requirements (e.g., as user stories).
5. Apply a modern software engineering process (e.g., Scrum) to developing a software project.
6. Produce high-quality software development artifacts for each phase of the development cycle.
7. Apply modern software testing tools and techniques (e.g., black- and white-box testing).
8. Develop software in collaborative teams.
9. Plan work and distribute tasks among team members to maximize team productivity.

4   Course Technology

4.1   Required Equipment

Students will be required to bring a laptop computer to lecture that can reliably run programming tools and a development environment.

Minimum System Requirements:

- **CPU:** Dual-core (Intel i5 / AMD Ryzen 3 / Apple M1)
- **RAM:** 8 GB
- **Storage:** 50 GB+ free
- **OS:** macOS 14+, Windows 11 (64-bit)+, or Linux (Ubuntu 22.04+)
- **Battery:** At least 2 hours of use on a full charge
- **Connectivity:** Stable Wi-Fi and ability to connect to campus networks

Other Considerations:

- A functional webcam and microphone are required for remote activities.
- The device must be capable of running the development environment and screen sharing software simultaneously without significant performance degradation.
- Tablets, Chromebooks, and iPads do not meet the requirements for this course, as they cannot run all necessary development tools.

It is the student's responsibility to have and maintain a working laptop development environment to use for this class.

If you do not have a suitable machine in your possession by the end of the first day of class, I strongly recommend that you postpone taking the course until you do.

4.2   Required Software Tools

University Email: Check daily. Used for official or record-keeping communication (e.g., attendance issues, grade inquiries, or other administrative matters).

Microsoft Teams: Used primarily for course announcements and getting help. You can configure email notifications for missed messages and activity when you are away from the app. This is your best option for posting technical questions or getting help with assignments in the evenings.

Canvas: Your main source for assignment instructions, due dates, submissions, grades, and feedback. Canvas announcements will not be used in this course; announcements will be posted in Teams. Do not use Canvas comments to ask questions about a grade you received; email the instructor *and* TA directly.

GitHub: Private GitHub repositories under the course GitHub organization will used as part of the deliverables for some coding assignment submissions.

5   Required Textbooks

There are no required textbooks for this course. The instructor will provide all reading materials.

6   Evaluation

Grading weights are as follows:

- 5% Participation Activities
- 45% Skills Development
    - 30% Rails Skills Assignments (6 assignments @ 5% each)
    - 15% Software Design Assignments (3 assignments @ 5% each)
- 50% Team Project
    - 25% Team Achievement
        - 7% Milestone 0 (Initial Planning)
        - 4% Milestone 1 (Development)
        - 4% Milestone 2 (Development)
        - 10% Final-Product Evaluation
    - 25% Individual Productivity
        - 10% Milestone 1 Regular Productivity
        - 10% Milestone 2 Regular Productivity
        - 5% Above and Beyond Productivity

To convert from percentages to letter grades, see the following table:

≥ 97%	91–96%	89–90%	87–88%	81–86%	79–80%	77–78%	71–76%	69–70%	67–68%	60–66%	≤ 59%
A+	A	A-	B+	B	B-	C+	C	C-	D+	D	F

I reserve the right to *lower* the percentage threshold for letter grades as I see fit (i.e., I may make the grading scale better for you but never worse).

6.1   Attendance and Participation Policy
Attendance is expected for all scheduled class sessions. Active participation is crucial both for your individual success and for effective collaboration with classmates and project teams.

6.1.1 In-Class Participation Activities

There will be <u>unannounced in-class activities</u> that contribute to your participation grade. These cannot be made up if you are absent. If you miss such an activity, refer to Section 6.5.1 for late work/absence policies.

6.1.2 Project Phase Attendance

During the team project phase, class sessions often function as "project studio" time where you strategize and plan with your team. Daily attendance is expected.

Each milestone has a participation component based on your engagement and presence. Poor attendance or lack of communication with your team can reduce your milestone grade.

6.1.3 Missing Class

If you miss or know you are going to miss class, contact the instructor as soon as possible—preferably before the absence. If you believe the absence may be excused, provide documentation as required (see Section 6.6)

During the project phase, you must also inform your team and, if possible, continue communicating remotely so you don't miss out on planning and collaboration.

6.1.4 Time Expectations and Study Habits

- **General Guideline:** Plan to spend **2-3 hours working outside of class for every hour in class**. Typically, a new assignment is introduced each class session, and you will need those 2–3 hours to complete it before the next class.
- **15-Minute Rule:** If you cannot solve or make progress on a problem within 15 minutes, ask for help—whether from classmates or the instructor/TA. This mirrors professional developer practices and prevents wasted effort.
- **Avoid Restarting Entirely:** Instead of discarding all progress when you hit a roadblock, debug and iterate on your existing work. This approach cultivates real problem-solving and reflects industry standards more accurately than "starting over."
- **Time Management:** If you struggle to balance coursework with other responsibilities, consider seeking out time-management resources or discuss strategies with the instructor. Consistent practice and efficient problem-solving techniques will help you succeed.

6.2 Skills Development

The first half of the course will focus on the development of skills necessary for software engineering and development.

6.2.1 Rails Skills Assignments

The Rails Skills Assignments will have students perform actual coding tasks using their laptop software development environments. A key aim of these Skills Assignments is to help students acquire the skills needed to perform core development tasks quickly and correctly. Such skill and efficiency are essential for an effective software development team member.

- There will generally be three activities in each Skills Assignment with separate due dates throughout the week:
    - **Active Reading:** Read and perform the steps in the specified development demos. Instructor videos will be provided for each demo to guide you through the process.
    - **Practice Task:** Complete a practice task that combines the concepts from all demos in the Active Reading activity. An instructor solution video will be

provided after you have attempted the task, allowing you to review and understand the correct approach.

- o **Explanation Video:** Record yourself performing a task that is similar (isomorphic) to the Practice Task. While recording, explain each step of your solution as you perform it, demonstrating your understanding of the material.
- Each Skills Assignment activity will be graded separately as pass/fail (i.e., no partial credit).
- Students must pass all assigned activities in a Skills Assignment to receive any credit for that assignment. **Be aware:** <u>This means that if you miss one explanation video you lose 5% on your final grade even if you complete the active reading and practice task.</u>

6.2.2 Software Design Assignments

The Software Design Skills Assignments will focus on critical aspects of software design, emphasizing analysis, modeling, application security, and the application of design patterns. These assignments may or may not involve coding tasks but will require students to engage deeply with design principles and methodologies, either individually or in groups. The primary aim of these assignments is to develop the students' ability to think critically about design choices, effectively communicate their ideas, and collaborate on complex problems. Mastering these skills is essential for becoming a well-rounded and effective member of any software development team.

6.3 Team Project

The centerpiece of this course is a team software project. Teams of 3-4 students will work together to develop a software system for a customer.

I reserve the right to assign the teams and to reshuffle them as I see fit.

Team projects in an educational setting must balance two concerns: (1) the need for students to work together as cohesive teams and (2) the need for individual accountability. Thus, half of your project grade will be based on what your team achieves, and half will be based on your individual contributions to the project (i.e., your productivity).

6.3.1 Team Achievement

Teams will complete a series of project milestones, each with its own goals and instructions. Milestones are evaluated based on criteria which include the following:

- Quality of planning and design artifacts
- Effectiveness of communication
- Discipline in following software engineering processes and procedures

Additionally, at the end of the project, there will be a final evaluation of the product produced by the team. This evaluation will focus mainly on the quantity and quality of features built.

The score awarded for the above items will generally apply to the whole team (i.e., every team member will receive the same grade); see exceptions for low individual productivity below.

6.3.2 Individual Productivity

6.3.2.1 Regular Productivity

The majority of your individual productivity points are associated with *regular productivity*. Each team member will be assigned certain tasks for each milestone. It is expected that each team

member will complete their assigned tasks in a timely manner. It is also expected that team members will be continuously productive instead of rushing to complete all their assigned tasks at the last minute. Failure to do so may result in deductions from your regular productivity grade.

6.3.2.2   Above and Beyond Productivity

To achieve the highest grades in the course (A/A+), you will need to go above and beyond the call of duty; thus, your individual productivity grade also accounts for *above and beyond productivity* (aka *A&B*). Throughout the course, you will have the opportunity to take on special roles or to complete bonus tasks for A&B points in addition to your regular task assignments. There is no limit on the number of A&B points you can earn, but note that you will need at least 5 A&B points to get full credit.

6.3.2.3   Deduction for Unproductiveness

A student who demonstrates unsatisfactory productivity may also lose points on the scores they receive for the team component of the associated milestone and the final-product evaluation. Such deductions are meant to account for the lack of contribution made by an unproductive team member to the project.

6.4   Assignment Submission Policies

All assignment instructions and official due dates will appear in Canvas.

Note that some assignments have multiple parts with their own due dates. These will be listed within the Canvas assignment description and/or announced in Teams. Therefore, be aware that not all tasks will show up in Canvas's "Upcoming Deadlines"—you must track them yourself.

You may be required to submit work via Canvas dropboxes and/or by pushing code to a GitHub repository, depending on the assignment.

6.4.1   Canvas Dropbox Submissions
- **Due Date:** Typically, **11:59 pm** on the stated day.
- **Grace Period:** Usually open until **5:00 am** the following morning, with **no penalty** for submissions in that window.
- **Closure:** After 5:00 am (or the stated grace-period end), your work is **late**.
- **No "Work-Arounds"**: Submissions sent by email or Teams or attached in Canvas comments *after closure* are invalid. If you are having temporary Canvas issues, you may send an email with your submission to the instructor *before closure*.
- **Submitting Links**: For most assignments, you will be expected to upload files to the Canvas dropbox. However, since this is sometimes infeasible for code projects or large files, you may need to submit a link to an externally hosted video file or GitHub repository.
  - **Grant Access to Graders**: You must provide access to the instructor and TA before the due date. If we discover a permission error during grading, we will notify you by email and give you 24 hours to fix it. If you do not resolve the permission issue within 24 hours, your link will be treated as not submitted.
  - **Verify Your Link**: Always confirm your link is working before submitting. A link to a resource that does not exist will be treated as not submitted.
  - **No Modification After Submission**: Externally hosted files can sometimes be changed after submission. We will only consider work that was completed

before the due date (or grace period end). Timestamps and version histories may be checked to ensure compliance with deadlines.

- **Team Project Dropboxes**: Any assignments during the team project that are stated to be the team's responsibility will use Canvas group dropboxes, meaning that only one team member needs to make a submission. However, given the policy stated in Section 6.3.2.3, Canvas requires each team member's grade to be entered separately, and the student submission page may not indicate that it is a group dropbox.

6.4.2 GitHub Repository
- **Push Deadline:** Only commits pushed before the due date (or grace period end) count as on time.
- **Push Timestamp:** This is the official submission time. If a push-time exploit is discovered, I reserve the right to introduce new constraints.
- **In some cases, the version history may be important.** Following best practices, you should make small frequent commits to show how your submission evolved over time.

6.5   Late Work Policy
Any work not submitted before the due date or grace-period end (when applicable) is considered late.

6.5.1   In-class Participation Activities
No late work accepted. If you miss such an activity due to an excused absence (see below), that activity's grade will be dropped.

6.5.2   Rails Skills Assignments
Each Rails Skills Assignment activity has a first and second chance submission.

On the first chance, no late work is accepted. <u>You must make a submission on the first chance to receive a second chance.</u>

The second chance follows the "one-chance assignment" policy (below).

6.5.3   One-Chance Assignments
Most individual assignments (e.g., Design Assignments, some individual project-related tasks) have one submission window.

No late work will be accepted unless you have a valid, documented excuse (see below).
If you do not have a valid, documented excuse, you will receive a zero.

6.5.4   Team Assignments
Some assignment submissions are the responsibility of the entire team. In these cases, even if one team member has an excused circumstance, the team is still expected to make a submission by the stated due date (or grace period end). In grading, some allowances may be made for the contribution of the team member with the excused circumstance.

6.6   Acceptable vs. Unacceptable Excuses for Absences or Late Work
Late work will be accepted only for documented circumstances that prevent on-time submission. Below are some examples (not exhaustive).

Other Circumstances: If you face a situation not listed here, notify me as soon as possible (preferably before class) with any relevant documentation. I will review it on a case-by-case basis according to university policies.

6.6.1 Acceptable Excuses (with Documentation)
- Medical Emergencies: Serious illness or hospitalization, with a note from a licensed healthcare provider.
- Family Emergencies: Documented bereavement or critical family situations.
- University-Sanctioned Events: Official travel to conferences, athletic events, or other approved academic commitments (with documentation from an official university sponsor).
- Natural Disasters: Incidents like flooding or storms that cause campus closures or power outages, with verifiable proof.

6.6.2 Unacceptable Excuses
- Forgot About the Deadline: "I didn't realize it was due today."
- Travel Plans: Personal vacations or early departures for holidays not recognized by the university.
- Procrastination: "I started too late" or "I had other assignments."
- Minor Technical Issues: Losing internet access briefly or forgetting a charger (unless documented by an IT service disruption that was severe and widespread).

7   Academic Integrity

The University of Memphis expects all students to behave honestly. The Student Code of Rights and Responsibilities explains what constitutes a violation of our Academic Integrity policy. For more information, please see the Office of Student Accountability's website: https://www.memphis.edu/osa/. Plagiarism, cheating, and other forms of academic dishonesty are prohibited. Students who violate the academic misconduct policy, either directly or indirectly, through participation or assistance, are immediately responsible to the instructor of the class in addition to other possible disciplinary sanctions which may be imposed through the regular institutional disciplinary procedures.

Examples of academic dishonesty include, but are not limited to:
- Cheating – A student uses a smart phone to access the internet while taking a quiz.
- Copyright infringement – A student uses a photograph found on the internet in a presentation without obtaining permission from the photographer.
- **Deception – A student gives a dishonest excuse when asking for a deadline extension.**
- Denying access to information or material – A student makes library or shared resource material unavailable to others by deliberately misplacing those resources.
- Fabrication – A student invents data in an academic work.
- Facilitating academic misconduct – A student knowingly allows a portion of their work to be used by another student.
- Plagiarism – A student represents the ideas of another in a paper without citing and referencing the work or a student turns in the same or nearly the same assignment for credit in more than one class.

- Sabotage – A student prevents others from completing their work by opening a window to affect a temperature-controlled experiment.
- Unauthorized collaboration – A student works with other students on a paper without the specific permission of the instructor.

7.1 Course-Specific Policies

- ***As per department policy, any student caught cheating in the course will receive at minimum a 0 grade on the assignment/test and be reported to the [Office of Student Accountability](#).***
- Teammates (i.e., members of the same team) may collaborate and share work however they see fit; however, if asked to report each team member's contributions, students must provide honest responses.
- Students from different teams may not collaborate in this way.
- Teammate collaboration is limited to project work and is not allowed on any other coursework (e.g., homework, quizzes, exams), unless expressly noted.
- ***During the project only***, students are free to use outside resources (e.g., coding blog posts or guides, YouTube tutorials, generative AI, etc.) to help complete assigned coding tasks. However, ***you must credit all sources and collaborators***. Failure to do so will be considered cheating.

8   Classroom Behavior

Students should be aware of the [Student Code of Rights and Responsibilities](#) which describes examples of unacceptable classroom behavior. Disruptive classroom behavior will not be tolerated. Instructors are empowered to remove students from class and refer behaviors for sanctioning to the Office of Student Accountability.

9   Disability Resources for Students (DRS) Accommodations

Please see the instructor if you need accommodations for a disability, or to fulfill cultural or religious obligations. Students with requests for accommodations should contact [Disability Resources for Students](#) to register and learn about the services available to support their learning. Students with disabilities are encouraged to speak with us privately about academic and classroom accommodations. It is strongly encouraged that you register with Disability Resources for Students (DRS) to determine appropriate academic accommodations. Disability Resources for Students is located in 110 Wilder Tower, their phone number is (901) 678-2880 (V/TTY), their email is drs@memphis.edu, and their website is https://www.memphis.edu/drs/. Disability Resources for Students coordinates all accommodations for students with disabilities.

Qualified students with disabilities will be provided reasonable and necessary academic accommodations if determined eligible by the appropriate Disability Resources for Students staff at the University. Prior to granting disability accommodations in this course, the instructor must receive written verification of a student's eligibility for specific accommodations from the Disability Resources for Students staff at the University. It is the student's responsibility to initiate contact with University's Disability Resources for Students staff and to follow the established procedures for having the accommodation notice sent to the instructor.

10 Mental Health

As a student you can sometimes feel overwhelmed, lost, experience anxiety or depression, and struggle with relationship difficulties or diminished self-esteem. Mental health challenges can interfere with optimal academic performance. However, many of these issues can be effectively addressed with some help. If you find yourself struggling with your mental or physical health this semester, please feel free to approach me. I will try to be flexible and accommodating. As your instructor, I am not qualified to serve as a counselor, but UofM offers confidential counseling services on-campus and via telehealth that are available to students taking six or more credits at no cost. UofM Counseling Center is staffed by experienced, professional psychologists, clinical social workers, and counselors, who are attuned to the needs of college students. I strongly encourage you to take advantage of this valuable resource. To connect with Counseling Center services, please visit 211 & 214 Wilder Tower, or call 901.678.2068. To know more about their services, you can visit their website at https://www.memphis.edu/counseling. In a crisis situation, please call 901.678.HELP (4357) to speak to the On-call counselor. Remember, getting help is an intelligent and courageous thing to do -- for yourself and for those who care about you.

11 Personal or Academic Challenges including Food & Housing Insecurity

If you are experiencing personal or academic challenges including, but not limited to food or housing issues, family needs, or other stressors, please visit the Dean of Students Office to learn about resources that can help. Any student who faces personal challenges including, but not limited to securing their food or housing and believes this may affect their performance in the course is urged to contact the Dean of Students Office at 901.678.2187 located in the University Center, Suite 359 for assistance. If you are comfortable doing so, please also let the instructor know you are experiencing challenges as they may be able to assist you in connecting with campus or community supports.