Syllabus: CS 350 Introduction to Software Engineering (Fall 2025)

Steven Zeil

Last modified: Sep 10, 2025

Contents:

1 Course Description

This course explores the software development process. It will discuss the major activities common to software development processes, and some of the ways in which those activities are organized and managed, with particular emphasis on how to successfully work together within a team.

Heavy emphasis will be placed on the day-to-day skills required during software construction. The course will explore lessons and tools offered by major successful agile and open-source software efforts.

The course requires each student to participate as a member of a team in a significant team project. Each student will be required to demonstrate proficiency in several software development tools.

2 Basic Course Information

2.1 Instructors

Steven Zeil	E&CS 3208
(757) 683-4928	szeil@odu.edu
Office hours: see here	

2.2 Meeting Times and Delivery Method

Students must register for both a lecture section and a recitation section.

- Lectures: No regularly scheduled meetings

Lecture materials for this course will be available on-line, primarily in printed/text form.

Attendance is optional, but recommended.

- Recitations: Check your schedule to see when you are registered.
    - M 7:25-9:25PM
    - T 4:00-6:00PM

Recitations will be conducted by network conferencing (Zoom). Recordings will **not** be available.

Recitation sections will not meet every week, but the scheduled meetings listed in the course outline are mandatory.

- During weeks when recitations are not scheduled, project teams may wish to use that time for their own meetings.
    - Teams using that time for meetings may, at their own option, invite their recitation instructor to join them to answer questions or clarify aspects of the project.

2.3 Communications

Most communications in this course will be conducted via email. Expect instructors to reply to email within 24 hours for messages sent during normal working hours and within 48 hours on weekends and over holidays.

I also try to keep an eye on the CS350 channel on the unofficial CS/STEM Discord.

General questions about course content and about the course project should normally be sent to your lecture instructor.

Issues with the course website should be sent to szeil@odu.edu.

General guidelines:

- All communications are expected to conform to the norms for civility and respect for ones' classmates and instructors that are common to all on-campus speech and writing.
- Use your @odu.edu email account. (Canvas' built-in Inbox messaging is also acceptable, but may delay your getting a response.)
- Use descriptive subject headers and make sure to include the course name "CS350" as part of the subject header. This goes a long way towards keeping your email from being blocked by spam filters.
- Avoid sending screenshots when the purpose of your message is to show plain text to the instructor. These graphics are often hard or even impossible to read. Copy-and-paste the text directly into your email.

2.4 Course Pre-requisites

- CS 252 (Introduction to Unix for Programmers)
- CS 330 (Object-Oriented Programming and Design), *or*

- CS 361 (Advanced Data Structures and Algorithms)
    - Students who have not taken CS 330 or CS 251 are encouraged to take CS 261 (Java for Programmers) as a pre-requisite or, at the very least, to work through that courses's website during the first few weeks of the semester.

2.5 Required Text

Readings from the Internet will be assigned from the course website.

2.6 Computer Accounts

Students will need two network accounts to participate in this class:

- An ODU Midas account. This is the account associated with your @odu.edu email. It will allow you to log into the course's Canvas site. All ODU students automatically receive this account, though you may need to activate yours if you are new to ODU.
- An account on the CS Dept. network. This will be used for access to the CS dept computing resources, and for accessing and submitting assignments. You may have a CS account already if you were registered for a CS class last semester. If not, the account setup can be initiated at http://www.cs.odu.edu/ by clicking on "Account Creation" under "Online Services".

In addition, students will need accounts at GitHub and Trello.

2.7 Hardware & Software Requirements

Students will need frequent access to a PC capable of hosting software development activities or of connecting remotely to CS Dept servers where such activities can be performed.

Students will be attending network conferences **requiring** the use of a microphone. Webcams are optional.

For both remote access to CS Dept servers and for network conferencing, a good-quality internet connection is important.

The course will introduce students to a wide variety of software packages. All of these are open-source, free software, but students may need to install some of these on their chosen development machine (whether their own PC or in their account on the CS network).

3 Course Policies

3.1 Recitation Attendance

Project review sessions will be scheduled for selected weeks during the recitation periods. Attendance at these is mandatory. Failure to attend will result in substantial grade penalties for that portion of the project.

Attendance at other scheduled recitation sessions, as announced in the course outline, is also mandatory. Failure to attend may result in grade penalties.

3.2 Due Dates and Late Submissions

Programming-style assignments that are automatically graded will be accepted one day late at a 10% penalty, two days late at a 20% penalty, but not accepted after that.

Otherwise, late assignments and make-up exams will not normally be permitted.

Exceptions to this and other grading policies will be made only in situations of unusual and unforeseeable circumstances beyond the student's control. Arrangements must be made prior to the due date in any situations where the conflict is foreseeable.

"*I've fallen behind and can't catch up*", "*I'm having a busier semester than I expected*", or "*I registered for too many classes this semester*" are *not* grounds for an extension.

3.3 Academic Honesty

Everything turned in for grading in this course must be your own work, or, for team projects, the work of your own team. Opportunities for teamwork will be clearly identified as such.

Students are expected to conform to academic standards in avoiding plagiarism.

- Among other things, this means that if you use ideas found on the internet (outside of the course website) in your answers to an assignment or exam question (including when coding!), you *must* cite your sources appropriately.

If you use text directly taken from such sources you must appropriately designate the quoted material as such.

The instructor reserves the right to question a student orally or in writing and to use his evaluation of the student's understanding of the assignment and of the submitted solution as evidence of cheating.

Students who contribute to violations by sharing their code/designs with others may be subject to the same penalties. Students are expected to use standard Unix protection mechanisms (chmod) to keep their assignments from being read by their classmates. Failure to do so will result in grade penalties, at the very least.

This policy is *not* intended to prevent students from providing legitimate assistance to one another. Students are encouraged to seek/provide one another aid in learning to use the operating system, in issues pertaining to the programming language, or to general issues relating to the course subject matter.

Student discussions should avoid, however, explicit discussion of approaches to solving a particular programming assignment, and under no circumstances should students show one another their code for an ongoing assignment, nor discuss such code in detail.

Violations of this policy will be reported to the Office of Student Conduct and Academic Integrity for consideration for punitive action.

3.3.1 Self-Reporting and Peer Evaluation

This course is heavily based upon a group project, and an integral aspect of the project involves students tracking their work completed and evaluating themselves and their teammates.

Whether in direct communication with the instructor or when working with the surveys and project tracking tools provided for the course, students are required to be honest in their reporting.

Claiming someone else's work as your own will be considered a violation of the ODU Honor code.

Claiming to have completed more work than you actually performed will be considered a violation of the ODU Honor code.

3.3.2 Git: Do Not Rewrite History

Your git/GitHub repositories should be an honest record of the actions you have taken.

Avoid using the git commands to reset or rebase your repository. Do not delete or rename the main branch. In general, do nothing to attempt to rewrite the history of what you have submitted to GitHub.

Unless an assignment *explicitly* says that such actions are permitted, using git commands to rewrite the history of commits on GitHub will have the effect of freezing your grade as of the last push prior to the rewrite. In egregious cases, it may be taken as evidence of attempts to deceive the instructor or your team members.

This policy does **not** prohibit you from making use of your git history to bring back old versions of files or even of entire directories to replace work that you have decided is worse than the older version.

3.3.3 Use of Online Resources

You may **not** post details of course assignments, projects, or tests at online Forums, Bulletin Boards, Homework sites, etc., soliciting help.

You **may** use information that you have not solicited but have located, subject to the following restrictions:

- If you use someone else's ideas in your your writing or in your code, you must cite your sources appropriately. Within code, such citations appear in comments.

Example:

⋮

double x = 23.0;

double xsqrt = sqrt(x);

// Search algorithm based upon code by S Zeil at

//
https://www.cs.odu.edu/~zeil/cs361/latest/Public/functionAnalysis/index.html#orderedsequentialsearch

int loc = 0;

while (loc < arraySize && numbers[loc] < xsqrt)

⋮

- If you use someone else's words, in your writing or in your code, you must cite your sources appropriately **and** mark the quoted text. Within code, such citations appear in comments.

Example:

⋮

double x = 23.0;

double xsqrt = sqrt(x);

// Begin quoted code from S Zeil at

```
//
https://www.cs.odu.edu/~zeil/cs361/latest/Public/functionAnalysis/index.html#orderedsequentialsearch

int loc = 0;

while (loc < listLength && list[loc] < searchItem)

{

  ++loc;

}

// End quoted code
```

⋮

- Failure to appropriately cite any such "found code" will be taken as evidence of plagiarism.
- Use of ChatGPT or similar AI-enhanced search engines is considered to fall in the "located" information category. Any such material employed in submitted work must cite the engine used and quote any exact text used, as described above.

Students must retain a copy of the prompts supplied to the AI engine, and must supply that as a file PROMPT.txt in the root directory of the assignment/project.

- The overall principle stated in the first sentence of this section remains in effect. "Everything turned in for grading in this course must be your own work." If the bulk of your assignment, code, paper, etc., are copied, even with appropriate citation, to the degree that, in the judgment of the instructor, you have not demonstrated your own knowledge of the course material, you will receive a zero for that exercise.
    - In other words, you only be graded on what *you* have written.

3.4 Grading

Labs	5%
Assignments	20%
Semester Project	50%

Midterm Exam	10%
Final Exam	15%

We will drop the lowest assignment score and the lowest project phase score.

More details on the grading is available here.

3.5 Assignment Grading

Assignments will be submitted via GitHub, with the exception of phases 1 and 2 of the team project, which will be submitted via Canvas and Trello, respectively.

The last submission will be the one graded. If you believe an earlier submission might have scored higher, it is up to you to recover that earlier submission from git and to commit and push the earlier code.

Late submissions will not be accepted except as provided for in Due Dates and Late Submissions, above.

3.6 Exams

The Midterm Exam and Final Exam will be administered online via Canvas.

Students will be required to use SmarterProctoring to arrange for a live or online proctor.

4 Topics

Topics will include:

- Software development processes, including the waterfall, unified OO, and extreme programming models
- Revision management: local, centralized, & distributed approaches
- Configuration Management: project configuration, managing external libraries
- Continuous Integration, including testing in the cloud (AWS)
- Analysis tools, including CheckStyle, SpotBugs, & PMD
- Build Management, including Ant, Maven, and Gradle
- Unit & Integration Testing: coverage, JUnit-style frameworks, mocking, designing for testability
- Test-driven development
- Issue tracking

- Software Forges & Repositories
- Dev-Ops

4.1 Objectives

Students completing this course should be able to:

- Demonstrate an understanding of the overall strategy of software development:
    - Discuss the phases and component activities of software development
    - Assess the likely impact of popular software process development models on a project.
    - Discuss common team organizations and roles in software development.
- Work with software requirements documents
    - Read common forms of requirements documents
    - Write at least one standard form of requirements document
    - Apply requirements to guide the subsequent construction of software
- Apply best practices in collaborative software construction
    - Discuss the issues and problems involved in collaborative development of software.
    - Evaluate the suitability of alternative best practices for a software construction project.
    - Support common best practices of via a modern IDE and associated tools
    - Apply a variety of software measurement and evaluation techniques.

4.2 Expectations

Students will engage in team projects in this course. Students are expected to **actively** participate in and contribute to their teams, and the level of engagement with the team will be part of the grade.

Procrastination and just-before-the-deadline submissions are detrimental to any team dynamic, and will result in lowered grades.

5 Semester Project - Civility among Team Members

You will be working with your team for many weeks, and there will be a lot of communication expected among team members.

In accordance with the Monarch Creed and Code of Ethics, I expect all students to maintain *civility* in their dealings with one another.

Language that is abusive, harassing, or threatening to members of the class or that fosters high levels of personal and emotional anxiety may, at the instructor's discretion, result in expulsion from the team. Given the importance of the team project to this course, that is likely to result in a failing grade. Egregious or repeated violations will be referred to appropriate authorities for possible disciplinary action.

6 Educational Accessibility

Old Dominion University is committed to ensuring equal access to all qualified students with disabilities in accordance with the Americans with Disabilities Act. The Office of Educational Accessibility (OEA) is the campus office that works with students who have disabilities to provide and/or arrange reasonable accommodations.

- If you experience a disability which will impact your ability to access any aspect of my class, please present me with an accommodation letter from OEA so that we can work together to ensure that appropriate accommodations are available to you.
- If you feel that you will experience barriers to your ability to learn and/or testing in my class but do not have an accommodation letter, please consider scheduling an appointment with OEA to determine if academic accommodations are necessary.

The Office of Educational Accessibility is located at 1021 Student Success Center and their phone number is (757) 683-4655. Additional information is available at the OEA website http://www.odu.edu/educationalaccessibility/

7 General University Policies

The ODU Catalog lays out a wide variety of University policies that are binding upon both students and faculty. All students are required to abide by these.