CSCE 3444 Software Engineering

Instructor Contact

Name: **Jonathon Doran**
Office: **NTDP E250J**
Student Hours: **Mon 11:00am to 1:00pm, Thursday 10:00am to 12:00pm.**
You may request additional appointment times by emailing me.
Email: **Jonathon.Doran@unt.edu**
IA:
Meets: **Friday 5:30pm to 8:20pm in NTDP K110**
Presentations: **6-May at 5:30pm. Optional slots the week before.**

Communication Expectations

The best way to reach me outside of class or student hours is via email. During the week, I will respond to your email within 24 hours. On the weekends, the response time may extend up to 48 hours. Please do not expect immediate responses. You may discuss personal concerns or questions about the class or an assignment. I strive to get grades back between 1 to 1.5 weeks from the due date, although that may vary depending on class size. Communication is expected to be professional and respectful.

For effective communication please review the Online Communication Tips (https://clear.unt.edu/online-communication-tips) which cover key aspects of respectful and clear online interactions. **Please include "CSCE 3444" in the subject line of all emails for easier identification and a prompt response. Please include your group name as well.**

Course Description

Students demonstrate their mastery of the breadth of computer science learned in their studies. Focus is on the application of computer science techniques to the design of applications involving multiple software components. Students apply the theory acquired from numerous computer science courses to solve real-world design and developmental problems. The design considers realistic constraints including economic, environmental, critical thinking, technical writing and communications skills, and group management skills in completing their design and development project.

Course Prerequisites or Other Restrictions

You are required to have passed the following courses:

- CSCE 3110 (Data Structures and Algorithms)
- CSCE 2100 (Computing Foundations I),
- CSCE 2110 (Computing Foundations II),
- CSCE 1035 (Computer Programming I).

If you are not comfortable with the material covers in these courses, please see me to discuss your situation.

Attendance and Participation

Attendance is required and plays a crucial role in your success in this course. Active participation in class discussions, group activities, and lectures is essential for both your understanding of the material and your contributions to team projects.

Attendance is not just about your individual learning—it's about your responsibility to your teammates. In this course, your active presence is essential not only for your understanding of the material but also for the success of your group. The team-based nature of this course means that your participation directly impacts your peers, and any absence can disrupt the progress of your group.

In-class activities, such as iClicker polls and discussions, form a significant part of the learning experience and cannot be made up if missed. These activities are designed to reinforce the course material and encourage collaborative learning. Your commitment to attending every session on time is a commitment to your team's success. I expect all students to treat this responsibility with the seriousness it deserves.

Half of your iClicker grade is extra credit, allowing for a few missed lectures without serious consequences. However, consistent attendance and participation will not only enhance your understanding but can also boost your overall grade if you engage regularly.

If you must miss a class due to unavoidable circumstances, you are responsible for notifying your teammates as soon as possible and making arrangements to ensure your absence does not negatively impact your group's work. Persistent absences or lack of participation will not only affect your grade but could also hinder your team's ability to meet project deadlines and objectives.

Course Objectives

Course outcomes are measurable achievements to be accomplished by the completion of a course. These outcomes are evaluated as part of our ABET accreditation process. By the end of this course, students will be able to:

1. Use UML for design, such as use cases and class diagrams.
2. Conduct software testing, such as validation, integration, and unit testing.
3. Conduct usability testing, such as heuristic evaluations.
4. Participate in peer reviews such as code inspections.
5. Communicate software product and process results in oral and written form.

Required Materials

> *Software Engineering: A Practitioner's Approach, 8th edition, Pressman, Maxim, McGraw Hill, ISBN 9780078022128*

Teaching Philosophy

I believe the best preparation for professional work is active, hands-on engagement with real design and development problems. You'll be working in teams, facing ambiguity, and producing artifacts that mirror what engineers create in industry. I view my role not only as a professor but as a mentor and

guide. I will challenge you to apply what you know, learn from setbacks, and grow into engineers who can lead teams and communicate effectively. I challenge you because I know that you can meet those challenges. I also believe software engineers have a responsibility to consider the people who will use their systems. We will touch on accessibility, usability, and the role of emerging tools (including AI) to prepare you for the realities of practice

Grade Components

Most the deliverables in this course will relate to a large group project that will be completed based on client requirements. You are expected to participate in all aspects of your team's project. Your effort, including that of coding, will be monitored (e.g. repository check-ins, peer evals and observations). In particular, lack of coding on the project **WILL RESULT** in a **LARGE** penalty to your project grade. Teams will be assigned early in the semester to balance skills and interests. Being an active member of your assigned team is required to pass this course.

There are no examinations in this course. We will use the final examination time for team project presentations.

Your class participation grade is primarily based on your engagement with in-class activities, including iClicker polls and class discussions.

This course has no laboratory component. Class meeting time is not reserved for project work as the course project is an out-of-class assignment. Questions about your specific project should be handled during office hours or by appointment, since they are generally not of interest to the entire class. You may schedule an individual or group appointment and I will discuss your issues in detail.

Project Requirements

These will be spelled out in another document. But groups must meet regularly with me where I will take on the role of your client. Students must treat client meetings as REQUIREMENTS-GATHERING sessions. You may *pitch* ideas, but you may **not** present completed code, claim features are "done," or unilaterally tell the client or instructor what the requirements are. Requirements presented to the instructor must be either (a) documented client statements (email or signed note) **or** (b) agreed in an instructor-mediated meeting. The instructor is the final arbiter of course requirements and scope.

Violations of this policy will result in a warning at first, followed by grade penalties to the final project. If you feel that something was previously approved, you may subject documentation up to 72 hours later to avoid any penalty. This policy is to ensure meetings focus on eliciting and documenting client requirements, keep project scope fair and achievable, and teach professional practices aligned with the course learning objectives.

Assignment	Percentage of Final Grade
Team Project	45%
Individual Assignments	35%
Quizzes and other Activities	20%

Grading Policy

If you believe that there is a mistake in the grading of one of your assignments, you must bring these inquiries to the professor within one week of when the assignment is returned. After this period, it is too late to consider, except for an arithmetic error in calculating the score.

I guarantee that these percentages will be the maximum required for a letter grade. I usually lower these as needed to avoid having a large number of students on a grade boundary. My determination of the boundary is final and will be applied equally to all students. Being close to the boundary does not entitle you to a higher grade.

Grading Scale

A = 90% - 100%
B = 80% - 89%
C = 70% - 79%
D = 60% - 69%
F = 0% - 59%

Sprint Summary

Sprint 0 12-Jan to 25-Jan
Sprint 1 26-Jan to 8-Feb (artifacts due 13-Feb)
Sprint 2 9-Feb to 22-Feb
Sprint 3 23-Feb to 8-Mar
Sprint 4 9-Mar to 22-Mar
Sprint 5 23-Mar to 5-Apr
Sprint 6 6-Apr to 19-Apr
Sprint 7 20-Apr to 1-May

Course Calendar

It is hard to anticipate the exact dates of activities, but here is a tentative list. Note that items listed as due are by week, not day. The exact due date will be given on Canvas. Note that sprints are two weeks long, and are shaded in alternating colors below. Each sprint runs from a Monday through the second Sunday. Please note that sprint artifacts are due one week AFTER a sprint ends. This gives you time to prepare them.

Week	Sprint	Date	Major Topics	Due
1	0	16-Jan	Intro to SE (Ch 1)	Quiz
2		23-Jan	Process Models (Ch 2), Project Management (Ch 3)	Project Proposal
3	1	30-Jan	Requirements (Ch 7,8)	Project Plan/Shell
4		6-Feb	UI Design and Prototyping (Ch 12)	Assignment 1
5	2	13-Feb	Design and Architecture (Ch 9,10)	Initial SRS, Peer Eval,
				Sprint-1 artifacts
6		20-Feb	Design Patterns (Ch 14)	
7	3	27-Feb	Risk Mgmt, Config Mgmt (Ch 26,22)	Design Doc, Peer Eval Sprint 2 artifacts
8		6-Mar	Code Inspection (Ch 16)	Assignment 2
9	4		Spring Break	
10		20-Mar	Testing & Integration (Ch 19, 20)	Peer Eval, Sprint-3 artifacts
11	5	27-Mar	AI in SW Engineering	Test Plan, Peer Eval Sprint-4 artifacts
12		3-Apr	Metrics (Ch 23)	Assignment 3
13	6	10-Apr	Heuristic Evaluation (Ch 12)	Peer Eval, Sprint-5 artifacts
14		17-Apr	Quality (Ch 15,17)	Heuristic Report
15	7	24-Apr	Technical Debt, Refactoring	Peer Eval, Sprint 6-artifacts
16		1-May	Cloud Ops, CI/CD, Security	Sprint 7-artifacts
F		6-May	Presentations at 5:30pm	Scheduled demos

You will be notified by Eagle Alert if there is a campus closing that will impact a class. This calendar is subject to change.

Late Submission and Extension Policy

ABSOLUTELY NO LATE work will be graded, unless specific arrangements are made with me in advance. All work should be turned in on Canvas or on GitLab by the assigned due date. It is your responsibility to make sure that your submission is complete. I suggest that one person agree to double-check your team submissions.

Incompletes: Incompletes are generally not given for group project courses, as there is no way to make them up later. Please contact both myself and the Dean of Students if you have any exceptional situations.

Extensions: **ALL** requests for extensions on assignments must be made **prior** to the due date and must be for a valid "emergency" reason. In extreme circumstances, contact after the due date may be accepted if there is a **COMPELLING** reason. Again, reach out to the Dean of Students Office for assistance in documenting your situation.

Syllabus Change Policy

The instructor reserves the right to change the course as needed, including but not limited to topics, due dates, and assignments/assessment items. The pace of the course may also be adjusted based on student progress and understanding. Any changes will be made after careful consideration of the course objectives and student progress.

Other Class Policies

Activity in your assigned Git repository will be monitored and incorporated into your final grade. Work is only considered delivered when it is merged into the master branch and demonstrated.

Weekly status reports are required; they must reflect actual activities documented in the repository. Your individual reflections are for you to highlight your progress, and are not a place to discuss other students. Group status reports are where you discuss the state of the project and where you are heading.

UNT Policies

Academic Integrity Policy

According to UNT Policy 06.003, Student Academic Integrity, academic dishonesty occurs when students engage in behaviors including, but not limited to cheating, fabrication, facilitating academic dishonesty, forgery, plagiarism, and sabotage. A finding of academic dishonesty may result in a range of academic penalties or sanctions from admonition to expulsion from the University.

Based on this policy, any form of "unauthorized assistance" constitutes cheating. If the use of artificial intelligence is not authorized for the assignment in question, then a violation has occurred.

ADA Policy

UNT makes reasonable academic accommodation for students with disabilities. Students seeking accommodations must first register with the Office of Disability Accommodation (ODA) to verify their eligibility. If a disability is verified, the ODA will provide a student with an accommodation letter to be delivered to faculty to begin a private discussion regarding specific course needs. Students may request accommodations at any time; however, ODA notices of accommodation should be provided as early as possible in the semester to avoid any delay in implementation. Note that students must obtain a new letter of accommodation for each semester and must meet with each faculty member prior to implementation in each class. For additional information see the ODA website (https://disability.unt.edu/).