CSCE3444: Software Engineering

Instructor: Dr. Jonathon Doran, Jonathon.Doran@unt.edu

Office: Hall Park 126
Office Hours: M,T,W 2-3 & By Appointment

Overview:

The topics in this course include requirements and specifications development, documentation of the design (including UML), testing of software implementation, usability testing, and system and user documentation.

Textbook:

Software Engineering: A Practitioners Approach, Roger Pressman, 8th Edition, McGraw-Hill. ISBN 9780078022128

Objectives:

Students should gain these skills by the end of the course:

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

Grading:

Late assignments are accepted at the rate of -20% credit for each day late unless accompanied by a university approved medical excuse. No make-up quizzes will be given if you are more than 10 minutes late to class.

Cheating will result in failure in the course. Please reference the UNT academic integrity policy for more information on cheating. **Individual work such as homework assignments and pop quizzes must be done on your own. Do not discuss solutions or share copies of individual work.**

The grading breakdown will be as follows:

Pop quizzes (lowest 1 dropped)	40%
Group Project (grades scaled by peer evaluations of individual performance)	30%
Individual Assignments	30%
Total	**100%**

Letter grades will be determined as follows:

A >= 90%, B >= 80%, C >= 70%, D >= 60%, F < 60%

Group project grades will be scaled for individual team members based on peer evaluations, so be sure to help your team as much as possible. It is not an acceptable excuse to say that your team did not give you enough work, because I expect you to speak up and volunteer if you feel that you aren't doing your fair share. If you have a problem that you are not able to work out with your team, please contact me to mediate prior to project submission (and as early as possible).

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 with the exception of an arithmetic error in calculating the score.

Attendance: Attendance and class participation are strongly encouraged, so plan to attend regularly. Students are responsible for any material and announcements covered in class. If you cannot attend, plan to get notes from a classmate.

Major Assignments:

Assignment 1: Use Cases and Misuse Cases (30 pts)

As a team, select the most critical use cases for your project. Each person on the team chooses one use case from this set and elaborates the use case using the provided template. Present a misuse case related to your use case, and elaborate that using the provided misuse case template. Submit the file as a single PDF file.

Assignment 2: Class Diagrams (30 pts)

You will be presented with the requirements of a proposed system, and you will need to translate this into a class diagram using appropriate UML notation. Use Visio or a similar diagramming tool to create the diagram. When you submit this assignment, submit it as a PDF, JPG, or PNG file.

Assignment 3: Test Cases (30 pts)

For a provided system fragment, develop a set of test cases that provides maximum code coverage. You will be provided with a template where you enter the test vectors (the inputs, expected outputs, and relevant notes), and discuss steps to follow, as well as coverage details. Turn in a PDF version of this file.

Schedule

Date	Topics	Readings	Due
8/26	Intro to the Class	Ch 1, 2	**Team member list** due Friday at 11:59 pm
8/28	What makes a good team(mate)		
9/4	Process models and roles	Ch 3-5	**Project Proposals** due Friday at 11:59pm
9/9	Project Management	Ch 31, 6, 8	**Project Plan and Shell** due Sunday at 11:59pm. Status Report due Saturday at 11:59pm
9/11	Requirements		
9/16	Requirements cont.	Ch 9, 10, 15	**Assignment 1** due Wed at 11:59pm Status Report due Saturday at 11:59pm
9/18	UI Design (Prototyping)		
9/23	From Req to Design	Ch 12, 14	**SRS** due Wednesday at 11:59pm Status Report due Saturday at 11:59pm
9/25	More on Design and Best Practices		

9/30	Patterns	Ch 16	**Assignment 2** due at Wednesday at 11:59pm Status Report due Saturday at 11:59pm
10/2			
10/7	Risk Management	Ch 29, 35	**Project Design** due Friday at 11:59pm Status Report due Saturday at 11:59pm
10/9	Configuration Management		
10/14	Code Inspection	Ch 20	**Assignment 3** due Friday at 11:59pm Status Report due Saturday at 11:59pm
10/16	Team Check In		
10/21	Code Inspection Activity		Status Report due Saturday at 11:59pm
10/23	Team Check In		
10/28	Testing Overview	Ch 21, 22	**Test Plan** due Friday at 11:59pm Status Report due Saturday at 11:59pm
10/30			
11/4	Metrics	Ch 32	Status Report due Saturday at 11:59pm
11/6	Meet with Teams		
11/11	Heuristic Evaluation Overview		Status Report due Saturday at 11:59pm
11/13	Heuristic Evaluation Activity		
11/18	Contemporary Topics in SE		Status Report due Saturday at 11:59pm
11/20	Meet with Teams, Project Check In		
11/25	Tools Revisited		Status Report due Saturday at 11:59pm
11/27	Meet with Teams, Project Check In		
12/2	Project Presentations		Status Report due Saturday at 11:59pm
12/4			
Finals	Final Demos + final revisions of artifacts due		

Project

You will work on a term project for this course. The intent is for this to be something that you work on the entire semester (so it is officially assigned now). There will be several checkpoints during the semester, where you will be graded on your progress.

Your team will have no more than six people.

You and your team will propose **three** projects, of which I will select 1. Each proposal needs to be complex enough for your team to work on it for most of the term.

The project **may**:

- Be done in a language of your choice (other than CSS/HTML)
- Be web-based or a traditional application

The project **must**:

- Have appropriate scope for the size of your team for one term
- Have a graphical user interface (GUI)
- Have some portion used by a user (not be entirely batch or automated)
- Have at least one customer outside of your group who can supply requirements

The project **must not**:

- Be trivial
- Be static web pages or screens
- Be a game
- Harm the security or privacy of anyone
- Be used primarily by minors

You will have a set of deliverables through the term, see the schedule for specific dates. These include:

- **Project Plan**
  A GitHub repository shall be created, an appropriate project folder structure devised, and the overall schedule and roles for your project will be documented within.
- **Software Requirements Specification (SRS)**
  Using a provided template, describe the functional and non-functional requirements using text and graphical notation.
- **Test Plan (Including Test Cases)**
  Using a provided template, describe the overall test plan and test cases, and keep track of your test status.
- **Heuristic Evaluations**
  Using the provided heuristic evaluation checklist, evaluate your project to date.
- **Status Reports**
  Each week you will, as a team, report your status. Each team member will report what they did that week, what they plan to work on next week, and any challenges/concerns they have.
- **Final Project Presentation**
  During the final week of class, each team will make an oral presentation to the class on the project.
- **Final Demo**
  During finals week, each team will provide a final demo of the project to the class.