CSCE 247 - Syllabus

1: Course Overview

Course Name: CSCE 247 - Software Engineering
Semester: Spring 2019
Instructor: Greg Gay (ggay@cse.sc.edu)
Lecture Hours: Monday and Wednesday, 3:55 - 5:10 PM, 300 Main St. B102
Office Hours*:* Tuesday and Thursday, 4:00 - 5:00 PM,
          2247 Storey Innovation and Engineering Center
Website: https://dropbox.cse.sc.edu/course/view.php?id=194 (Dropbox)
          http://greggay.com/courses/spring19csce247/ (Backup - may be out of date)

Course Description

Software engineering is concerned with the development and evolution of high-quality software systems in a systematic, controlled, and efficient manner. Software engineers are concerned with safety and reliability of the product as well as the cost and schedule of the development process. The lectures and the group projects will cover all aspects of the software life cycle, from development team management, problem specification and analysis, system design techniques, implementation and documentation practices, testing, to maintenance and evaluation of the final product.

Where does this fit in with the rest of my computer science courses?

Many computer science classes deal with taking complex domain-specific problems and deriving solutions from the appropriate mathematical and computational theories. In an AI course, you reason about intelligence problems and design software that solves such problems.

Software engineering is the study of software itself. It is focused around designing, developing, and documenting reliable, functionally complete, and usable software. In software engineering, you will learn to reason about *software* itself, and will learn lessons that apply to any program you design in the future[1].

Learning Outcomes

1. The students will be able to distinguish between software development processes and choose an appropriate process for a particular project, including the selection of appropriate source control and project management tools.

---

[1] For more, see http://www.davidbudden.com/degrees-demystified-1/

2. The students will be familiar with requirements elicitation and be able to create a requirements specification document.
3. The students will learn about software architectural models and understand how to analyze how control and data flow through a system.
4. The students will understand the principles of object-oriented software design, including how to describe and model the structure of a system.
5. The students will understand and be able to apply software design patterns.
6. The students will be familiar with the fundamentals of requirements-based and structure-based software testing and the accompanying test selection methods.

Outline of Topics to be Covered

- Introduction (1 week)
- Requirements Specification (2 weeks)
- Project Management (1 week)
- Design Fundamentals (1 week)
- Software Architecture (1 week)
- Design (OO) (3 weeks)
- Implementation (1 week)
- Testing (3 weeks)
- Reliability and Maintenance (1 week)

See the course schedule for specifics.

Textbooks

- *Head First Design Patterns*. Eric Freeman, Bert Bates, Kathy Sierra, Elisabeth Robson.
- *Software Engineering*. Ian Sommerville. Tenth Edition (Ninth Edition is acceptable).
    - An electronic version of this textbook is available from http://www.mypearsonstore.com/bookstore/software-engineering-9780133943030
- *UML Distilled: A Brief Guide to the Standard Object Modeling Language*. Martin Fowler. Third Edition.

All three books are purely optional, and serve as reference books for course content. I do not assign homework directly from any single book. These books are used as additional reading and back-up. Therefore, you do not have to purchase a copy of any. I recommend at least purchasing a copy of "Head First Design Patterns".

Additional readings will be assigned as the course progresses. These readings will be available on the course web page or handed out in class.

2: Course Requirements and Grading

Requirements

You should have knowledge of some higher level programming language (such as C, C++, or Java). Programming assignments for the course will use Java; therefore, you should be able to pick the language up on your own if you do not already have experience in it. You are expected to understand basic data structures (such as lists, sets, and trees), algorithms (such as sorting, searching, and tree traversals), recursion, data abstraction, and finite state machine models.

Exams

There will be an in-class midterm examination, as well as a final exam. All exams are closed-book.
- Midterm: Wednesday, March 6, in class
- Final: Monday, May 6, 4:00 - 6:30 PM **(confirm with final exam calendar)**

Assignments and Group Participation

We will assign several homework assignments throughout the semester. A core project, composed of approximately six individual deliverables will be completed in groups. There may be additional assignments to be completed individually.

You need to pull your weight on all group assignments. Substandard work is obvious to your fellow students and the instructor and will be reflected in your grade. Peer evaluations will be turned in throughout the semester, and will be used during grading. Additionally, the exams will reflect project content. If you have not done the project work, you *will* do poorly on the exams.

Grading

40% of your grade will be based on the individual assignments and group project. You are graded on the quality of the work you produce, not on how many hours a week you spend. The details of how much each deliverable is worth will be announced with the projects and assignments. The midterm will be worth 20%, and the final will be worth 30%. The remaining 10% will be awarded based on in-class activities and participation in the group.

Students are required to perform satisfactorily on both exams and assignments to receive a passing grade. All assignments and tests will be awarded 100 points. A **general guideline** for grading will be the following:

Total Score	Letter Grade
100 >= score >= 90	A
90 > score >= 87	B+
87 > score >= 80	B
80 > score >= 77	C+
77 > score >= 70	C
70 > score >= 67	D+
67 > score >= 60	D
60 > score >= 0	F

3: Policies and Procedures

This section contains some general rules that will be enforced during this course. Please review these guidelines carefully. The course is governed by the policies in the Carolina Community: Student Handbook & Policy Guide (http://www.sa.sc.edu/carolinacommunity/judicial/). Violations of this code can result in actions varying from a failing grade to expulsion from the university.

Integrity and Ethics

The homework and programs you submit for this class must be entirely your own. If this policy is not absolutely clear, then please contact me. Any other collaboration of any type on any assignment is not permitted. It is also your responsibility to protect your work from unauthorized access. Any violation of this policy will result - at minimum - in a 0 on the assignment. Further infractions will result in a failing grade in the course and further disciplinary action.

Classroom Climate:

All students are expected to behave as scholars at a leading institute of technology. This includes arriving on time, not talking during lecture (unless addressing the instructor), and not leaving the classroom before the end of lecture. Disruptive students will be warned and potentially dismissed from the classroom.

Make-Up

The midterm (held during regular lecture hours) and final are required. If any of the tests fall on a religious holiday, the tests will be rescheduled.

Make-ups for graded activities may be arranged if your absence is caused by a documented illness or personal emergency. A written explanation (including supporting documentation) must

be submitted to your instructor. **If the explanation is acceptable, an alternative will be arranged.** Whenever possible, make-up arrangements must be be completed prior to the scheduled activity. A student not taking an exam or not turning in an assignment will receive a score of 0. Alternative times for the final exam will be arranged only under university criteria for rescheduling a final exam.

Late Submissions

Homework assignments are due at the time noted on the assignment handout. Late work is not accepted without prior approval. Any assignment turned in after the due date will be considered late and will be subject to a penalty of 10% per day, including weekends and holidays. Submitting all assignments is a necessary condition for passing this class.

Attendance Policy

Attendance will not be formally checked. However, be aware that the course does have a participation grade. Failing to take part in too many of the in-class activities may result in loss of participation credit.

Special Needs

It is university policy to provide, on a flexible and individual basis, reasonable accommodations to students that have disabilities that may affect their ability to participate in course activities or to meet course requirements Students with disabilities are encouraged to contact their instructor early in the semester to discuss their individual needs for accommodations.

Diversity

Someday you will graduate, and in the real world, you will have to work with a wide variety of people. Now is the time to abandon preconceived prejudices about others. Students in this class are expected to respectfully work with all other students, regardless of gender, race, sexuality, religion, or any other protected criteria. There is a zero-tolerance policy for any student that discriminates against other students.