COMP 523: Software Engineering Laboratory

Bulletin Description
Organization and scheduling of software engineering projects, structured programming, and design. Each team designs, codes, and debugs program components and synthesizes them into a tested, documented program product.

General Course Info
Term: Spring 2019
Course/section: COMP 523 section 001
Time: MW(F) 11:15 to 12:30
There will also be weekly meetings with the instructor (or TA), many team meetings, and
                        some client meetings
Location: Sitterson 014
Website: http://www.cs.unc.edu/~stotts/COMP523-s19/
Email: help-comp523 @ cs . unc . edu *(send all class email here)*

Instructor Info
Name: Dr. David Stotts
Office: Brooks 144
Phone: 919 . 590 . 6133
Web: http://www.cs.unc.edu/~stotts
Office Hours: Tuesday 1:00 to 2:30, or by appt.

Name: Dr. Jeff Terrell
Office: Brooks 146
Phone: 919 . 590 . 6103

Any email sent to the instructor's personal account risks getting lost in the huge pile received every day, so please use the class account. The class email account will be read regularly by the instructors and TA. The instructor will communicate with students via the email addresses given in ConnectCarolina for the class. Please get in the habit of scanning your email for information from the instructor about the class. Message titles will contain "COMP 523".

Class Meetings
Class meetings will be on M and W 11:15 to 12:30. The Friday slot is reserved officially as class time as well -- **do not schedule personal activities into the Friday slot.** we will have occasionally a Friday class meeting, and that slot also gives us a guaranteed slot in which all teams have a time each week to meet. We repeat -- do not schedule over the Friday slot.

Textbooks and Resources
There are no required or recommended textbooks. Sakai will be used primarily for the returning of comments and grades on individual assignments (if any… there may be none). Individual assignments (if any) will be submitted through Sakai and team assignments will be submitted through the team's website. Server software resources

will be available as needed. All recommended or required readings will be available or referenced on the class website.

Course Description
The goal of this course is to teach the skills necessary for building a software product as a team. The lecture portion of the class will cover the broader picture of software engineering that includes a wide range of software development projects in terms of size, complexity, and criticality. The course carries EE (experiential education) and C (communications intensive) tags and is an APPLES course. There will correspondingly be a written assignment or two to fulfill the reflection component is an APPLES course.

The team aspect of this class will vary from team to team, as each team will work with a different client to develop a different software product. The projects will all be software that the clients want and will find useful, but no team will be working on software that is critical to the well-being of any client or client's business.

Target Audience
This course is intended for upper class majors with an interest in building software for practical use. Students are expected to have enough experience to be able to learn new software systems on their own and to design a system using techniques and principles learned in other courses. This is an ideal course for those interested in getting real world experience in building software and communicating with others.

Prerequisites
COMP 410 and 411 plus two additional COMP courses numbered 426 or higher. The additional two courses will ideally cover software tools, techniques or principles.

Goals and Key Learning Objectives
At the end of the course, each student will have experienced all aspects of a software development project, including
- working with a client to define goals and priorities
- designing a system
- scheduling and planning a multi-person project
- effective communications
- running meetings
- writing technical documentation
- preparing web content
- writing and testing code
- deploying the system
- creating and delivering public presentations

Course Requirements and Grading
The essence of the course is the faculty-coached team project. Teams of 3-4 students spend the semester negotiating, estimating, scheduling, specifying, coding, debugging, integrating, documenting and testing a substantial programming product. Each project has a real client that is expecting a completed project. Each document will be submitted

to the professor in draft form and will be revised based on comments. In addition, documentation needs to be maintained to reflect changes in the product that is being produced. Details of each step (and deadlines) will be delivered using the class website given above.

There will be no written exams; there may be individual assignments given to cover the key concepts of the course that are not well reinforced through the project and to expose you to the literature in the field.

There are a lot of new things happening in the field of software engineering that you are not exposed to within the department's curriculum. Each team will present a technology from a provided list or one that they are using that has not been taught in other classes. Teams will give a 20 minute "tech talk" presentation to the class.

Project grades are based on code, documentation, ambition, effort, teamwork, and accomplishment. This includes attendance at meetings and the fulfillment of milestones for those meetings. Your project manager will be keeping a checkbox form tracking the team and individual satisfactions of these various milestones.

The team will also make a midterm presentation of about 10 minutes. Right before break, your team will explain to the class how the project has ended up being defined, and what overall approach you are taking to building it. The final exam is the third team presentation; in this the end product will be demonstrated, and team experiences will be explained. The final presentation will be about 15 minutes.

Grading Criteria

Overall breakdown
I compute a single grade for the project, based on the following percentages:

- Process 20% (check box form of milestones)
- Code 40%
- Documentation 15%
- Midterm presentation 5%
- Final Presentation 10%
- Tech talk 10%

Your project website will contain all the materials that we will evaluate for the project grade (your code, documentation, presentation PPTs, etc.). We will also have the record of individual attendance at meetings and team satisfactions of deliverables and deadlines. We will consult each client to get an evaluation of how well the team delivered the project and worked with the client for acceptance testing.

We will also collect from each of you a personal report, communicated to the instructors confidentiality in email. Your individual report will give tell us how well you think each team member did in contributing to the overall project.

We then take the team project grade and apply an individual adjustment for each student on the team. This adjustment value is based on manager observations as well as the evaluations by your client, any consultants, and your peers. If your team project is judged to be B level, team members individually could earn different grades, such as C+, A-, etc. based on individual contributions to the project.

A few more details:
- Process includes whether you are interacting appropriately as a team, with me, and with your client. Are you addressing issues as they arise? It includes professionalism in your dealings with your client and your professor and whether you are usually on time with deliverables or habitually late. It includes your web site and the materials that you produce as steps to produce the other artifacts. I will be evaluating 3 process phases: the requirements phase, the design and implementation phase, and the completion. The reason for that is to recognize that there are different processes and issues during different phases of the project.

- The code grade covers function, correctness and readability. The three components are equally important. Have you met the primary requirements? How many bugs was I able to find? I will do a random review of the code that you produce. I expect to be able to understand it. This includes web pages as well as other code that you write. The code portion also covers automated unit testing… was it complete and thorough?

- The documentation covers the formal deliverables: the functional spec, the design document, the user manuals, and the test plan. They are weighted equally. Remember that spelling, grammar, and readability are important; unreadable good content is not sufficient.

- For the midterm, final, and tech talk presentations, your grades will include both content and style. Midterm will be primarily an overview of your design and technological platform decisions. Final will be a working demo, design review, and lessons learned. Tech talk will be a learning experience for the team, communicated to the class.

Technology Talk

Your grade will include both content and style. I will be looking to see if you understand what you are presenting and whether you are communicating well with your classmates.

Course Schedule and Key Dates
Lecture dates and topics, as well as dates of exams (presentations) and any assignments are found in the class calendar:
http://www.cs.unc.edu/~stotts/COMP523-s19/calendar.html
This is also accessible from the menu at the top of the class homepage.

Course Policies
Attendance: I expect student attendance in class. I expect attendance at all peer presentations and guest presentations. Specifically, I expect you to be at all technology talk and demo presentations, and to attend outside speakers. For other classes, I only point out that there is no textbook in this class because the content is not available in any simple form. If you are interested in the content, you need to listen to lectures. We hope to have a number of outstanding not-to-be-missed outside speakers during the semester.

Finally, I expect your attendance at class because the seat you occupy is in high demand. If you do not wish to be in class, there are many who do wish to be there.

The course final is given in compliance with UNC final exam regulations and according to the UNC Final Exam calendar.

Honor Code
Collaboration and peer-learning are necessary for team projects. Only the individual assignments (if any) are not to be done collaboratively. These are open book, open notes, and open network. My goal is to give you work that requires individual thought and reflection and the work must be that of the student. Directly taking text from other sources is not acceptable. Short excerpts from other sources may be quoted and properly cited.

Disclaimer
The professor reserves the right to make changes to the syllabus, including assignment and project due dates, as well as percentages for assignments and exams towards final course grade. These changes will be announced as early as possible and will be reflected on the course website. If there are discrepancies between this syllabus and the website, the website is considered the definitive information.