CS 496 Senior Project, Winter 2008

Syllabus

Instructor: Christopher (Chris) Reedy
Office: CF463
Email: mailto:Chris.Reedy@wwu.edu?subject=CS496
Phone: 650-4838
Office Hours: To be arranged by team

Description

Execute a software development project, including requirements, design, implementation, testing, and documentation.

During finals week, each project team will demonstrate their project. The senior project demonstration is tentatively scheduled for 1pm-3pm, Thursday, March 20 (scheduled time for 4pm class with Tuesday or Thursday meeting). Please let me know if this date and time does not work for you.

Students must take the Major Field Test (MFT). (See below.)

Prerequisites

Senior status, senior evaluation, permission of department

Administrative

Send me an e-mail with your schedule from your preferred email account. Include "CS496" in the subject line. I need to know about jobs and other similar obligations as well as any courses you are taking. I will use this e-mail address if I need to contact you on short notice (which usually happens once or twice a quarter).

Major Field Test

All students are required to take the Major Field Test. The exam will be given in early November. (Exact date is to be determined.) There will be a $35 charge for the exam which is the student's responsibility. More information on the MFT is posted on the web site. There will be some (at least one) review sessions for the MFT.

You must take the major field test in order to pass the course.
Project

The class will be divided into teams. Each team will execute a computer/software development project. Here are some ground rules:

- Teams should be three to five students. In exceptional cases, teams can be two or six students. Experience shows that three or four is the optimal size.
- At the end of the quarter, each team will demonstrate the results of its project. The demonstrations will consist of a short briefing describing the project followed by a demonstration of what was built. Each team will provide a poster for their project.
- Today will be the last meeting of the entire class until the demonstrations. For each team, we will arrange a time for a meeting with me every week (approximately 30 minutes).
- The nature of the project is up to the team and the interests, capabilities, etc. of the team members. If you're fishing for ideas, there is a list of sample projects attached to this syllabus. (Or, look at sourceforge.net or similar sites for project ideas.)
- There are no constraints other than resource availability on the technologies used for the project.
- The first team meeting will be during the week of January 14 to 18. For the first meeting, each team should come prepared with a short (about two page) project plan covering the items below. I expect the plan to evolve as the project progresses, and you should too.
  - a project name (referring to teams by number doesn't work for me)
  - a description of what you plan to build
  - a list of features with priorities (must have, like to have, do this if we have extra time) for those features
  - any resources required (other than what is available in the CF 162, 164, 405 labs). This should include other software or hardware, faculty support, access to special labs, etc.
  - what tasks need to be done, when they will be done, and who will do them
- Guidance: Generally, you should plan to have something built and integrated around midterm with additional features or capabilities added after that. You should assume a week with little or no progress around midterms.
- I will approve (or reject) the project at the first meeting. Reasons for rejecting a project: too big, too small, too risky, non-availability of resources, inability to demonstrate. (Research projects are usually considered too risky.)
- During subsequent weekly meetings, each team should be prepared to discuss the work accomplished during the past week, any problems, issues, etc. that you've encountered, and what you are planning to accomplish during the next week.
- After the demonstration, each team will provide the instructor with a CD or DVD with all project code, executables, web sites, your poster, documentation, etc., etc. Failure to provide the CD will result in an incomplete grade for the course.
Grading

Grades will be based on the following percentages:

- 25%: score on the major field test
- 75%: the project

The project will be graded based on:

- size of the project
- the quality of the result
- the demonstration
- your poster
- teamwork, both with other members of your team and the instructor
- challenge and creativity of the project

At the end of the project each student will be asked to send me a confidential email assessing each team member's contribution (including their own) and the percentage of the final result that represents. If there is a consensus that some members of the team contributed significantly more (or less) than the other members, I will "tweak" the project portion of the grade accordingly. If you do not send me an email, I will assume that you felt that all members of your team contributed equally.

Attendance Policy

All students are required to attend the final project demonstrations. Failure of students to attend, without prior consent from the instructor, will result in a penalty to their team.

All students are expected to attend the weekly team meetings. If you are going to miss the meeting, please let me know (e-mail suffices) in advance. Missing meetings on a regular basis will negatively impact your grade.

Academic Dishonesty

Academic dishonesty is defined in the University Catalog as misrepresentation by deception or by other fraudulent means which compromises an instructor's ability to fairly evaluate a student's work or achievement. It is the instructor's responsibility to confront a student and to take appropriate action if academic dishonesty, in the instructor's judgment, has occurred. Please refer to the University Catalog for further information.

Any student who violates the University policy on academic dishonesty will receive an F grade for the course.
Top ten things NOT to do on your Senior Project

(thanks to Bill Hurd)

1. Try to integrate all your code the night before the demo.
2. We don't need source code control.
3. Use a development tool that is in beta.
4. Implement a perfect democracy on your project; a benevolent dictatorship works much better.
5. Decide this is the term you are going to bathe less frequently.
6. Take four other heavy programming classes the same quarter as senior project.
7. Try build a fully functional first person shooter game from scratch, including the engine and graphics, in ten weeks.
8. Not show up for your weekly meetings and annoy the instructor.
9. Pick team members that you really dislike and annoy them.
10. Miss the major field test by going snowboarding for two weeks.
Some (Lame) Senior Project Suggestions

Spellbound
Based on the US National Spelling Bee, this tool will train spellers to compete in the regional and national competitions by tracking past performance in training and focusing on improving the accuracy of currently known words and adding new words to the spellers vocabulary.

Backgammon
A graphical, easy to use backgammon game. You can play against the computer or across the network. There is also a general server that allows an arbitrary number of connections.

Animal Shelter Manager
Animal Shelter Manager is a complete computer solution for animal sanctuaries and shelters. Features complete animal management, document generation, full reporting, charts, internet publishing, pet search engine integration, web interface and more.

Issue Tracker
Issue-Tracker is a customer/technical support system. The system is designed to be user-friendly and easy to customize. Features include things like email parsing, file uploads, email alerts, workflow, etc.

A Note
Software that let you put post-it like notes onto your desktop. When the computer shuts down the notes will be uploaded to a database so you can share your notes between your home and office computer.

FreeGuide TV Guide
FreeGuide is a TV guide. Download TV listings from the Internet, view them offline, create a personalized TV guide and choose favorites. It uses the XMLTV tools to grab listings for lots of different countries.

Run
Run is an enhanced interactive training diary for runners. With Run you can log every detail of your running activities, measure and track your progress and effort, and recognize fitness patterns. Use Run to maximize efficiency and avoid injuries.

Classical Music Collection
Track a collection of classical music files, including composers, performers and instruments, related tracks and discs, etc. Integrate with a media player for playing files with the music.
**Some Additional (Not So Lame) Project Ideas**

**Database of Rescued Dogs**

Rescue Every Dog ([http://www.rescueeverydog.com](http://www.rescueeverydog.com)) is looking for help building a database of dogs. Jordan Andersen is the local contact for this group. Here's some additional info from Jordan:

This organization has a "database" of thousands of animals that they have, are, or will work with. They currently keep track of all of this via a several hundred megabyte word file synchronized via subversion. Obviously, some sort of actual database system would be preferable.

**Logos Software**

Logos Software is a Bellingham company that makes software products for reading, searching, cross-referencing, etc. of religious texts (“Bible software”). They have a large body of data giving simultaneous translations of the Bible into other languages. They would be willing to help share this data with us for senior projects. Here are a couple of potential ideas:

- an automated cross-language query improver (ala Google “Did you mean: correct”. I have a reference to an article on how to do this quickly.
- a flash card system for learning languages
- an aid to memorization of bible verses, poetry, etc.

I have some more information if there is interest.