SYLLABUS

I. Title: CSC493 Software Project

II. Time and Place: Spring Semester 2013,
Friday 2:45-5:15 p.m. (RS 202/Virus lab);
Final Project Demos to the class on Monday, April 29th from 1 to 3 p.m.

III. Credit: Three units

IV. Instructor: Dr. McKinstry, Professor of Computer Science

V. Office Hours: Rohr Science 216, (619) 849-2269; email: jeffmckinstry@pointloma.edu

Monday:	11:00-11:50 a.m., 2:30-3:30 p.m.
Tuesday:	9:30 a.m.- 11:50 a.m.
Wednesday:	11:00-11:50 a.m., 2:30-3:30 p.m.
Thursday:	9:30 a.m.- 11:50 a.m.
Friday:	11:00-11:50 a.m.

VI. Text: Recommended: Bernd Bruegge and Allen Dutoit. Object-Oriented Software Engineering Using UML, Patterns, and Java 3nd Edition, Pearson, Prentice Hall, New Jersey, 2010.
Other resources: wxWidgets tutorial slides can be found on templates on Grumpy under the CSC 494 folder
More in-depth tutorials on wxWidgets can be found at www.wxwidgets.org.

VII. Objectives of the course: "… the CC2001 Task Force recommends that curricula include a capstone project that allows students to bring together all the skills and concepts that they have previously learned during their undergraduate courses. Such a course might include a small amount of additional material, but the major focus must be on the project." – *Computing Curricula 2001, Computer Science* produced by The Joint Task Force on Computing Curricula, IEEE Computer Society, and the Association for Computing Machinery.

VIII. Learning outcomes:
Students will be able to write correct and robust software.
Students will be able to apply their technical knowledge to solve problems.
Students will communicate effectively orally and in writing.
Students will understand the interaction between hardware and software.

IX. Course Organization: The Course Schedule provides an outline with dates for some of the important activities of the course. Class time will be used for:

1. Weekly project status meetings
2. Working on software project in the virus lab.

X. Student Evaluation:

Weekly status meeting participation/attendance	20%
Term project RAD (1 version)	10%
Term project ODD (iteration 1, 2 and 3)	15%
Term project STD (iteration 1, 2 and 3)	15%
Term Project demonstration + source code (iteration 1,2,3)	20%
Customer's grade (iteration 3, final product)	20%

You will be required to demonstrate your programs to the professor and to turn in an electronic copy of your work.

Grades will be determined as follows:

93-100%	A
90-92%	A-
87-89%	B+
83-86%	B
80-82%	B-
77-79%	C+
73-76%	C
70-72%	C-
67-69%	D+
63-66%	D
60-62%	D-
0-59%	F

Group projects are difficult to grade in a fair manner, but most software development projects in the real-world are group projects, and the team will be judged as a whole by the customer. To try to address the fairness issue in a group project, I reserve the right to add or subtract up to 10% (1 letter grade) based on your contribution to the group as determined by your time log and verifiable complaints by group members. Students who work significantly harder than the rest of the group may receive a higher grade, while students who work and/or contribute significantly less may have their grade lowered. Be a team player.

XI. Course Schedule (subject to change).

Week 1, Jan. 11: Pick projects. Project preference ranking due Wednesday, Jan 16th by email. I will assign projects by next Friday.
Week 2, Jan. 18: project status meetings
Week 3, Jan. 25: project status meetings
Week 4, Feb. 1: project status meetings + SRS due
Week 5, Feb. 8: project status meetings + iteration 1 SDD due
Week 6, Feb 15: project status meetings
Week 7, Feb 22: project status meetings + iteration 1 STD and demo due
Week 8, Mar. 1: project status meetings + iteration 2 SDD due
Week 9, Mar. 8: **Spring Break**
Week 10, Mar. 15: project status meetings
Week 11, Mar. 22: project status meetings + iteration 2 STD and demo due
Week 12, Mar. 29: **Easter Recess**
Week 13, April 5 : project status meetings + iteration 3 SDD due
Week 14, April 12: project status meetings
Week 15, April 19: project status meetings
Week 16, April 26: project status meetings
Week 17, Finals week. Iteration 3 STD and demo. **Demonstrate your project to the class during the final exam time, Monday, April 29th, from 1:00 to 3:00 p.m.**

XII. Projects

Projects may be individual projects or group projects. All projects must have a customer who cares about the project and wants to use the software after the course is over.

Project #1: Dr. McKinstry's pet project. Neuroscience research (Applies only to juniors who plan to consider doing summer research with Dr. McKinstry in summer 2013). If interested, discus with Dr. McKinstry after class. One current project involves developing and testing software that can recognize arbitrary motion paths of 3D objects. This may involve learning to program in MATLAB. Obviously the first month will involve background reading.

Project #2: Continuation of Software Engineering course project. Students would work in teams of 2 to complete and perfect the software from last semester. The reminder glasses app was almost complete, so unless there was a huge upgrade, that would not be a good candidate.

Project #3: Robot simulation software. There is a software simulator, called Player Stage, that allows one to simulate various commercial robots in a 3D virtual world. The project would involve installing this system on a Mac or PC, and writing code in C++ to control a virtual robot to attempt to solve one of the tasks from the DARPA ARM project. You can search the web for more info on this, but this would be a lot of fun.

Project #4: Artificial natural selection of a retina neural network. The idea is to write software that simulates natural selection in an artificial environment in order to "evolve" a neural network that resembles animal retinas. The computation of the retina network is well-known, so we have a good idea of what to look for. To my knowledge, no one has done this, and I believe it would be a significant achievement. Furthermore it could be done by a team of undergraduates.

Project #5: Join an Open Source software project team of your choosing and make a contribution. You must suggest which project you will join and describe it if you choose this project.

Project #6: You may find a customer of your own that has a software need and write software for them. This must NOT be a job you are already paid to do, or are doing as part of another course. The customer must contact me, and write a brief project description by the end of the first week.