CS 691 Syllabus

22 January 2015

Welcome to CS 691, the software development project.

Where: LLC 206, chair's office
Credits: 3

Contact Info

Instructor: Prof. Christopher League, Ph.D.
Email: christopher.league@liu.edu — please include the course number (CS691) in the subject. I have several email addresses, but all messages end up in the same place, so use only one.
Google Hangout: cleague@gmail.com
AIM: chrysleague
Office hours: Monday, Wednesday 2–2:50 or make an appointment at https://liucs.net/bookme
Office phone: +1 718 488 1274
Office location: LLC 206, LIU Brooklyn

Resources

Web sites: https://liucs.net/cs691s15/
https://github.com/liubrooklyn/cs691s15
https://piazza.com/liu/spring2015/cs691/
Text: No required textbook
Library: Campus library resources tailored for computer science are available at https://liucs.net/u1

Requirements

There are a total of 1,000 points available, broken down as follows:

- There will be **8 project milestones** scheduled throughout the semester. The exact requirements and expectations for each will be posted to the course web site. Your contribution will be worth **125 points each,** but I will drop the lowest, so that only 7 milestones count, for a total of **875 points. Warning:** the *last* milestone cannot be dropped.
- There is no midterm exam, but there will be a final exam, worth **125 points.**

On the 1,000-point scale, you can expect the following letter grades:

		≥ 870:	**B+**	≥ 770:	**C+**	≥ 670:	**D+**
≥ 930:	**A**	≥ 830:	**B**	≥ 730:	**C**	≥ 600:	**D**
≥ 900:	**A–**	≥ 800:	**B–**	≥ 700:	**C–**	else:	**F**

In the end, I may choose to adjust the scale slightly to compensate for assignments or questions that turned out to be trickier than I intended. Such adjustments would never *lower* your grade from what is designated in the above table; if you achieve 930 points, you are guaranteed an **A**.

Goals and objectives

Upon completion of the course, students should be able to…

- demonstrate proficiency in basic algorithms and data structures
- understand the mathematical and logical foundations of computing
- master the fundamental facilities of various programming languages and software architectures
- effectively use tools for software development
- develop a data modeling design for a proposed database application
- communicate technical ideas and specifications in writing
- give an effective oral presentation on some technical subject area
- exhibit awareness of professional organizations and technical opportunities
- productively attend seminars and workshops outside of class work

Schedule