

CMSI 698/598

OPEN SOURCE SOFTWARE DEVELOPMENT WORKSHOP

<http://myweb.lmu.edu/dondi/summer2006-1/cmsi698>

Special Studies, Summer 2006 Session I — Doolan 222

T 7:10–10:10pm, 3 semester hours

Office Hours: T 4:30–6pm or by appointment

John David N. Dionisio, PhD

e-mail: dondi@lmu.edu, AIM: dondi2LMU

Doolan 106; (310) 338-5782

Course Objectives

The primary objective of this course is to gain real-world experience with open source software development, including but not limited to: (a) learning key open source development concepts, (b) gaining proficiency with tools that are frequently used in the open source community (many of which are used in effective software development in general), and (c) making a concrete contribution to an open source software project. The course uses a mentored workshop format that emphasizes individual work and discussion over classroom lectures.

Course Requirements

Proficiency in one or more current programming languages and technologies; ability to learn new ones in a short period of time.

Materials and Texts

The following items comprise the recommended reading for the course. They include technical articles, news items, and other documents such as open source licenses. Most of them are online entities that undergo continuing revision:

- Kent Beck, Erich Gamma, David Saff. “JUnit Test Infected: Programmers Love Writing Tests.” <http://junit.sourceforge.net/doc/testinfected/testing.htm>
- Kent Beck, Erich Gamma, David Saff. “JUnit: A Cook’s Tour.” <http://junit.sourceforge.net/doc/cookstour/cookstour.htm>
- Martin Fowler. “Continuous Integration.” <http://martinfowler.com/articles/continuousIntegration.html>
- Jennifer LeClaire. “Richard Stallman Sets the Free Software Record Straight.” *LinuxInsider*, May 2006. <http://www.linuxinsider.com/story/exclusives/50122.html>
- Open Source Initiative. “The Open Source Definition.” Version 1.9. <http://www.opensource.org/docs/definition.php>

- Eric S. Raymond. “The Cathedral and the Bazaar.” <http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar>
- Eric S. Raymond. “The Magic Cauldron.” <http://www.catb.org/~esr/writings/cathedral-bazaar/magic-cauldron>
- Eric S. Raymond. “The Art of Unix Programming.” <http://www.catb.org/~esr/writings/taoup/html>
- Gary Rivlin. “Open Wallets for Open-Source Software.” *New York Times*, April 27, 2005. <http://www.nytimes.com/2005/04/27/technology/27open.html> (registration required for original; try Google to see if mirrors are available)
- Various authors. Approved open source licenses. Compiled by the Open Source Initiative. <http://www.opensource.org/licenses/index.php>

You will also likely do a lot of reading for specific tools required for your open source contribution.

Course Work and Grading

Graded coursework consists of 1 open source contribution (35%), 5 weekly reports (7% each or 35% of the total grade), and 1 term paper (30%). Letter grades are determined as follows: $\geq 90\%$ gets an A– or better; $\geq 80\%$ gets a B– or better; $\geq 70\%$ gets a C– or better. Grades may be curved upward based on qualitative considerations such as degree of difficulty, effort, class participation, time constraints, and overall attitude throughout the course. Grades are never curved downward.

Open Source Contribution

The focus of this course is a concrete contribution that you shall make to an existing open source project. The specific open source project and contribution must be determined by May 22 at the latest.

Contributions take many forms, such as bug fixes, new features, test cases, and documentation. You may choose anything that fits your interests and/or expertise, as long as you can finish and report on it by the last meeting on June 22.

Your deliverable for the open source contribution is a final report to be submitted on June 22. This report is not as much a paper as it is a factual account of what took place during the course. Key information to include in your report:

- The chosen project and why you chose it
- Technologies that you had to learn
- Problems/challenges you encountered
- Artifacts produced by your work: bug reports, contacts made, forum posts, patches, source code
- Final result of your contribution work: was the patch accepted, were files committed, were documents uploaded to the community, etc.
- Potential follow-up work to the contribution(s) that you made

Late reports will not be accepted.

Weekly Reports

Every Monday before class, you are asked to submit a report to me by e-mail, stating what has been accomplished in the prior week and what remains to be done. The idea behind these reports is to facilitate a “transparent” process — that is, anyone reading your reports should have a complete understanding of what work has been done and what problems or situations have been encountered. We can then use the Tuesday meeting to discuss your work. Late reports get half credit, but really, you don’t want to be late — at 35% of the total, lateness can cost you an entire letter grade (or more).

Term Paper

To supplement the experience of making your open source contribution, you are asked to submit a short paper (3–5 pages) on one or more of the recommended readings. If you choose to write your paper about an open source license, choose at least two and perform a comparison. The paper should consist of (1) a summary of the reading material, and (2) your own discussion or response to the ideas in the material; it will be graded along the following criteria:

1. *Content (40%)*: Does the paper show sufficient understanding of the reading material? Does the discussion show objective and critical thinking?
2. *Organization (30%)*: Is the text structured well? Are its ideas and flow easy to follow? Are distinct sections or topics clearly identified?

3. *Writing (20%)*: Are statements clear and easy to follow? Is the language precise and grammatically correct? Is the paper’s tone appropriate?
4. *Polish (10%)*: Is the content properly proofread? Are there any misspellings, typos, or other formatting faux pas?

The paper is due on June 22. Late papers will not be accepted.

Attendance

I am not a stickler for attendance, but I do like having a full class. Remember that the university add/drop with refund deadline is May 19. The deadline for withdrawal or credit/no-credit status is June 9.

University Policy on Academic Honesty

Loyola Marymount University expects high standards of honesty and integrity from all members of its community. Applied to the arena of academic performance, these standards preclude all acts of cheating on assignments or examinations, plagiarism, forgery of signatures or falsification of data, unauthorized access to University computer accounts or files, and removal, mutilation, or deliberate concealment of materials belonging to the University Library.

Course Schedule

Due to the workshop nature of the class, the course schedule consists primarily of deadlines:

- May 19* University add/drop with refund deadline
- May 22* Weekly report #1 due — contributions chosen
- May 29* Weekly report #2 due
- June 5* Weekly report #3 due
- June 9* University withdraw/credit/no-credit deadline
- June 12* Weekly report #4 due
- June 19* Weekly report #5 due
- June 22* Term paper and project report due; final “de-briefing” meeting

You can view the class calendar on the Web at <http://ical.mac.com/dondi/LMU>. If you have an iCalendar-savvy client (i.e., Mozilla Calendar, Ximian Evolution, KOrganizer, Apple iCal, etc.), you can subscribe to the class calendar at <webcal://ical.mac.com/dondi/LMU.ics>. On-the-fly updates and adjustments to the class schedule will be reflected in this calendar.