CMPSC 290
Principles of Software Development
Fall 2013

Syllabus

Course Instructor

Dr. Gregory M. Kapfhammer
Office Location: Alden Hall 108
Office Phone: +1 814-332-2880
Email: `gkapfham@allegheny.edu`
Twitter: `@GregKapfhammer`
Web Site: `http://www.cs.allegheny.edu/sites/gkapfham/`

Instructor's Office Hours

- Monday: 1:00 pm – 2:30 pm (30 minute time slots)

- Tuesday: 2:30 pm – 4:00 pm (15 minute time slots)

- Wednesday: 4:30 pm – 5:00 pm (15 minute time slots)

- Thursday: 9:00 am – 10:00 am (15 minute time slots) *and*
              2:30 pm – 4:00 pm (15 minute time slots)

- Friday: 1:00 pm – 2:30 pm (10 minute time slots) *and*
           4:30 pm – 5:00 pm (5 minute time slots)

To schedule a meeting with me during my office hours, please visit my Web site and click the "Schedule" link in the top right-hand corner. Now, you can browse my office hours or schedule an appointment by clicking the correct link and then reserving an open time slot.

Course Meeting Schedule

Lecture, Discussion, Presentations, and Group Work: Tuesday and Thursday, 11:00 am – 12:15 pm
Laboratory Session: Wednesday, 2:30 pm – 4:20 pm
Final Examination: Friday, December 13, 2013 at 9:00 am

Course Catalogue Description

A study of the principles and concepts used in the specification, design, implementation, testing, and maintenance of large software systems. Topics include requirements elicitation and analysis, formal specification, software architectures, object-oriented design, software measurement, software testing and analysis, and evolution of a program. Students practice the principles of software development by participating as group members in the creation of a significant software application. One laboratory per week. Prerequisites: Computer Science 210 and 220 or permission of the instructor.
Offered in alternate years.

Course Objectives

The process of developing software involves the application of a number of interesting theories, tools, techniques, and methodologies. In this class we will explore the phases of the major software development life cycles and examine the tools, concepts, challenges, and open questions associated with each phase. Throughout the semester, we will examine the interplay between the theory and practice of software development. We will delve into the details of software specification, design, implementation, testing, and maintenance through a discussion of book chapters and articles from the software engineering and software testing literature. Moreover, students will enhance their ability to write about software in a clear and concise fashion. Finally, students will gain practical software development experience in laboratory sessions and a final software development project.

Performance Objectives

At the completion of this class, a student should be aware of the fundamental challenges associated with software development. Furthermore, students should be comfortable with a wide array of concepts, methodologies, techniques, and tools that they can apply to the problem of developing large software systems. However, a successful student will emerge with more than an understanding of the tools (e.g., text editors, compilers, debuggers, integrated development environments, and version control systems) that a software engineer uses. A student also should have a fundamental understanding of the major software life cycles and the activities that take place in each phase of these life cycles. Finally, a student should have a basic understanding of some of the current research and the open research questions in the field of software engineering. After completing this class, a student should be equipped for further graduate study in the fields of computer science and software engineering. The student should also be able to participate in real-world software development projects by adeptly using modern software tools and working with a team of developers.

Required Textbooks

Software Engineering: Theory and Practice. Shari Lawrence Pfleeger and Joanne M. Atlee. Fourth Edition, ISBN-10: 0136061699, ISBN-13: 978-0136061694, 792 pages, 2009.
(References to the textbook are abbreviated as "SETP" in the syllabus and on the Web site).

The Mythical Man Month. Frederick P. Brooks, Jr. Second Edition, ISBN-10: 0201835959, ISBN-13: 978-0201835953, 336 pages, 1985.
(References to the textbook are abbreviated as "MMM" in the syllabus and on the Web site).

Students who want to improve their technical writing skills may consult the following books.

BUGS in Writing: A Guide to Debugging Your Prose. Lyn Dupré. Second Edition, ISBN-10: 020137921X, ISBN-13: 978-0201379211, 704 pages, 1998.

Writing for Computer Science. Justin Zobel. Second Edition, ISBN-10: 1852338024, ISBN-13: 978-1852338022, 270 pages, 2004.

Along with reading the required books, you will be asked to study many additional articles from a wide variety of conference proceedings, journals, and the popular press.

Class Policies

Grading

The grade that a student receives in this class will be based on the following categories. All percentages are approximate and, if the need to do so presents itself, it is possible for the assigned percentages to change during the academic semester.

Class Participation and Instructor Meetings	5%
First Examination	15%
Second Examination	15%
Final Examination	20%
Laboratory and Homework Assignments	30%
Final Project	15%

These grading categories have the following definitions:

- *Class Participation and Instructor Meetings*: All students are required to actively participate during all of the class sessions. Your participation will take forms such as answering questions about the required reading assignments, asking constructive questions of your group members, giving presentations, and leading a discussion session. Furthermore, all students are required to meet with the course instructor during office hours for a total of sixty minutes during the Fall 2013 semester. These meetings must be scheduled through the course instructor's reservation system and documented on a meeting record that you submit on the day of the final examination. A student will receive an interim and final grade for this category.

- *First and Second Examinations*: The first and second interim examinations will cover all of the material in their associated module(s). While the second examination is not cumulative, it will assume that a student has a basic understanding of the material that was the focus of the first examination. The date for the first and second examinations will be announced at least one week in advance of the scheduled date. Unless prior arrangements are made with the course instructor, all students will be expected to take these examinations on the scheduled date and complete the tests in the stated period of time.

- *Final Examination*: The final examination is a three-hour cumulative test. By enrolling in this course, students agree that, unless there are extenuating circumstances, they will take the final examination at the time stated on the first page of the syllabus.

- *Laboratory and Homework Assignments*: These assignments invite students to explore the concepts, tools, and techniques that are associated with different phases of the software development life cycle. All of the laboratory assignments require the use of the provided tools to design, implement, test, and maintain programs that solve important problems. To ensure that students are ready to develop software in both other classes at Allegheny College and after graduation, the instructor will assign individuals to teams for each of the laboratory assignments. Unless specified otherwise, each laboratory assignment will be due at the beginning of the next laboratory session. Many of the laboratory assignments in this course will expect students to give both a presentation and a demonstration of the software that they specified, designed, implemented, tested, and documented.

- *Final Project*: This project will present you with the description of a problem and ask you to implement a full-featured solution using one or more programming languages and a wide variety of software development tools. The final project in this class will require you to apply all of the knowledge and skills that you have accumulated during the course of the semester to solve a problem and, whenever possible, make your solution publicly available as a free and open-source tool. The project will invite you to draw upon both your problem solving skills and your knowledge of programming languages and software engineering tools. The final project will be completed in groups assigned by the course instructor.

Assignment Submission

All assignments will have a stated due date. The printed version of the assignment is to be turned in at the beginning of the class on that due date; the printed materials must be dated and signed with the Honor Code pledge of all the students in the group. When the printed version is submitted, the electronic version of the assignment also must be made available to the course instructor in a version control repository. Late assignments will be accepted for up to one week past the assigned due date with a 10% penalty. All late assignments must be submitted at the beginning of the session that is scheduled one week after the due date. Unless special arrangements are made with the course instructor, no assignments will be accepted after the late deadline. In addition to submitting the required deliverables for any assignment completed in a group, students must turn in a one-page document that describes each group member's contribution to the submitted deliverables.

Attendance

It is mandatory for all students to attend the class and laboratory sessions. If you will not be able to attend a session, then please see the course instructor at least one week in advance to describe your situation. Students who miss more than five unexcused classes, laboratory sessions, or group project meetings will have their final grade in the course reduced by one letter grade. Students who miss more than ten of the aforementioned events will automatically fail the course.

Use of Laboratory Facilities

Throughout the semester, we will experiment with many different tools that software engineers use during the phases of the software development life cycle. The course instructor and the department's systems administrator have invested a considerable amount of time to ensure that our laboratories support the completion of both the laboratory assignments and the final project. To this end, students are required to complete all assignments and the final project while using the department's laboratory facilities. The course instructor and the systems administrator will only be able to devote a limited amount of time to the configuration of a student's personal computer.

Class Preparation

In order to minimize confusion and maximize learning, students must invest time to prepare for class discussions and lectures. During the class periods, the course instructor will often pose demanding questions that could require group discussion, the creation of a program or test suite, a vote on a thought-provoking issue, or a group presentation. Only students who have prepared for class by reading the assigned material and reviewing the current assignments will be able to effectively

participate in these discussions. More importantly, only prepared students will be able to acquire the knowledge and skills that are needed to be successful in both this course and the field of software development. In order to help students remain organized and effectively prepare for classes, the course instructor will maintain a class schedule with reading assignments and presentation slides. During the class sessions students will also be required to download, use, and modify programs, diagrams, and data sets that are made available through the course Web site. Students who are not comfortable with compiling, editing, and running Java programs should see the course instructor.

Email

Using your Allegheny College email address, I will sometimes send out class announcements about matters such as assignment clarifications or changes in the schedule. It is your responsibility to check your email at least once a day and to ensure that you can reliably send and receive emails. This class policy is based on the following statement in *The Compass*, the college's student handbook.

> "The use of email is a primary method of communication on campus. . . . All students are provided with a campus email account and address while enrolled at Allegheny and are expected to check the account on a regular basis."

Disability Services

The Americans with Disabilities Act (ADA) is a federal anti-discrimination statute that provides comprehensive civil rights protection for persons with disabilities. Among other things, this legislation requires all students with disabilities be guaranteed a learning environment that provides for reasonable accommodation of their disabilities. Students with disabilities who believe they may need accommodations in this class are encouraged to contact Disability Services at 332-2898. Disability Services is part of the Learning Commons and is located in Pelletier Library. Please do this as soon as possible to ensure that approved accommodations are implemented in a timely fashion.

Honor Code

The Academic Honor Program that governs the entire academic program at Allegheny College is described in the Allegheny Course Catalogue. The Honor Program applies to all work that is submitted for academic credit or to meet non-credit requirements for graduation at Allegheny College. This includes all work assigned for this class (e.g., examinations, laboratory assignments, and the final project). All students who have enrolled in the College will work under the Honor Program. Each student who has matriculated at the College has acknowledged the following pledge:

> I hereby recognize and pledge to fulfill my responsibilities, as defined in the Honor Code, and to maintain the integrity of both myself and the College community as a whole.

Welcome to a Software Engineering Adventure

In reference to software, Frederick Brooks, Jr. wrote in Chapter One of MMM, "The magic of myth and legend has come true in our time." Software is a pervasive aspect of our society that changes how we think and act. High quality software also has the potential to positively influence the lives of people. Moreover, the specification, design, implementation, testing, maintenance, and documentation of software are exciting and rewarding activities! At the start of this class, I invite you to pursue this adventure in software engineering with enthusiasm and vigor.