Software Engineering
CPMA 536
Spring 2006

Meeting Information

College Hall 225
Monday 6:00–8:40 p.m.
Final exam: In-class Monday May 1
Course web page: `http://www.blackboard.duq.edu`

Instructor

Dr. Jeffrey Jackson
Office: 433 College Hall
Office Hours: MWF 3 p.m.–4 p.m., WF 10 a.m.–noon, drop-in, and by appointment
Phone: 412-396-6466
Email: `jackson@mathcs.duq.edu`

Textbook

Software Engineering, A Practitioner's Approach, 6th Edition, Pressman, McGraw-Hill.
Readings will be assigned as the course progresses.

Purpose of the Course

The primary goal of this course is to introduce you to many aspects of "real world" software development.
Some specific objectives are that you learn about:

- Developing software "in the large" as part of long-term team project as opposed to performing short-term individual development

- Dealing with uncertainty about software requirements

- Conducting research, including prototype development and testing, in order to address key technical questions

- Utilizing metrics to understand and improve the software development process

- Interacting with customers, colleagues, management, and employees

- Producing quality software

Expected learning outcomes include that at the completion of this course, you will be able to:

- Choose an appropriate software development process for a small- to medium-size project;

- Use context-free questions to effectively initiate a project;

- Create well-defined use cases;

- Develop a comprehensive set of requirements models;

- Choose an appropriate software design architecture;

- Perform component-level design that conforms with established design criteria;

- Document software requirements, design, and deployment choices using various Unified Modeling Language (UML) diagrams;

- Use basic configuration management tools properly; and

- Design effective test strategies and apply appropriate test techniques.

Grading

To provide an opportunity for putting the concepts covered into immediate practice, the class will work together throughout the course on a sizable software engineering project. Students will have individualized assignments which I will attempt to tailor to their backgrounds and interests. These assignments may include presenting project-related material in class. While there will not be any formal papers assigned, one or more students will be responsible for producing software documentation.

Grading will be based on:

- Project participation and contribution 50%

- Final exam 50%

The project participation and contribution grade will be based on my sense of each student's contribution to the overall success of the project. This will be based both on my observation of in-class team meetings and work sessions as well as feedback from other students. I will have individual "performance reviews" with each student near midway through the course that will give you feedback on your performance in each of several areas, such as technical skill, idea generation, knowledge sharing, and contributions to team cohesion. The

combination of these scores at the end of the semester will determine the project participation and contribution portion of your grade. Note that this is a large contributor to your overall course grade, as your project work will directly reflect how well you have learned and can apply the principles taught in the course.

The final exam will be comprehensive and similar in nature to homework problems in the textbook. The focus of the exam will be on seeing that you understand terminology and can apply concepts taught to new (relatively simple) problems, not on memorizing laundry lists. The exam will be conducted in-class and will be closed-book.

The final grade will be assigned using plus/minus grading as follows:

100–93 = A, 92–90 = A-, 89–87 = B+, 86–83 = B, 82–80 = B-, 79–77 = C+, 76–70 = C, 69–60 = D, below 60 = F.

Schedule

We are scheduled to have 7 class meetings. Meetings will include project team meetings as well as lecture portions. The tentative schedule is:

1. Project: initial meeting with customer, Eclipse setup. Lecture: Overview of software engineering, software processes, and requirements engineering.

2. Project: use case review, UML analysis model development. Lecture: Analysis modeling.

3. Project: analysis model review, architecture and component design, JavaDoc, project planning. Lecture: Design.

4. Project: design reviews, plan review, coding standards, JUnit, test planning. Lecture: Planning.

5. Project: plan review, test reviews, code reviews, JavaHelp. Lecture: Testing.

6. Project: plan review, test reviews, code reviews. Lecture: User interface design, software metrics.

7. Project: code reviews, acceptance testing. Lecture: Selected topics in software engineering.

Note

Students with documented disabilities are entitled to reasonable accomodations if needed. If you need accomodations, please contact the Office of Freshman Development and Special

Student Services in 309 Duquesne Union (412-396-6657) as soon as possible. Accomodations will not be granted retrospectively.

The information in this syllabus is subject to change at the instructor's discretion as circumstances dictate.