CSE 403
Software Engineering

Spring 2006

Pragmatic Programmer Tip: Care about Your Craft
Why spend your life developing software unless you care about doing it well?
Who am I?

- Gail Alverson
- Computer Science Professor
- Senior Engineering Manager at Cray Inc.
- Your teacher and entertainer for the next 10 weeks from 10:30-11:20 and beyond!
My last project: Red Storm system

- Massively parallel processing supercomputer system used for analysis and stewardship of nuclear weapons - for Sandia National Lab $93M
Who are you?

Help me out …

- Senior students about ready to start your career outside of school
- Talented, skilled, highly capable individuals
- Lifelong learners
- My captive audience for the next 10 weeks!
Why are we here?

Me:
To share my knowledge about the software development process and what it takes to both deliver a successful product AND be a successful engineer

You:
To learn the facets involved in software engineering, in order to be prepared for real world projects and a successful career

Both:
To have fun while learning skills critical to successful engineering!
Today’s outline

- Course overview
- Class web page

Readings

- “Rapid Development”, Steve McConnell, Ch 1, 2
- “Things they would not teach me in college”, Brechner
So what IS software engineering?

• Let’s hear your ideas!
• Here are some of mine:

**Software engineering** involves:

• Processes necessary to turn a *concept* into a *robust deliverable* that can *evolve* over time
• Working with *limited* time and resources
• Satisfying a *client*
• Managing *risk*
• *Teamwork* and communication
How will the material be taught?

1. **Class sessions** to discuss best practices
2. **Lab sessions** to dig deeper and/or discuss pragmatics
3. **Readings and essays** to reinforce the information
4. **Group project** to enable you to have direct experience with the material we’re covering
   - You’ll meet *technical challenges* given the larger project
   - You’ll meet *social challenges* given the team effort

*Your satisfaction in the class will directly relate to how well you engage in and conquer these challenges!*
1 and 2. Class and Lab Sections

• We’ll present new material and reflect on experiences
• We’ll have project presentations and demos
• We have 4 industry guest speakers lined up!
• We have a personality profile for you individually and your group planned

Interaction and discussion is encouraged!
3. Readings and essays

Readings will be identified each class:

- “Rapid Development” by Steve McConnell *(required)*
- “The Pragmatic Programmer” by Andrew Hunt and David Thomas *(recommended)*
- Handouts, distributed and/or online *(403 web site)*

A number of books are on reserve for 403 in the Engineering Library!

Essay assignments:

- Ask you questions or to read an article, and reflect on its content in light of your experiences
- Emphasis is on your “reasoning” and not whether you have the “right” answer
4. The Project

Here’s a high level view:

- **You make project proposals**
  - And then vote on which projects to keep
  - Start thinking about ideas today (teams of 2-3)!
  - Sample project ideas are linked from the course web

- **You’re divided into project teams of about 8 students**
  - Otherwise it'd be toy development, and you'd miss on some of the most important experiences

- **You develop your deliverable in stages (SE process!)**
  - Reflects modern methodologies for effective software project development
  - You get feedback from us and each other after each stage

By “living” the class material, you truly learn its meaning and can build on your experiences with it.
What is the general flow of topics?

Part 1: Introduction – quick run through SE basics
  o Software development lifecycle
  o Project ideas, architecture, plans, best practices

Part 2: Visit SE components in depth
  o Requirements, specifications, architecture, design, plan and risk assessment, quality assurance, release
  o Team dynamics

Part 3: Special topics
  o Project recovery, maintainability, intellectual property, guest speakers, agile methods
What is the grading scheme?

<table>
<thead>
<tr>
<th>Percentage</th>
<th>Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>10%</td>
<td>Lifecycle Objective (project idea)</td>
</tr>
<tr>
<td>10%</td>
<td>Lifecycle Architecture (project arch and plan)</td>
</tr>
<tr>
<td>20%</td>
<td>Reflective essays (2)</td>
</tr>
<tr>
<td>15%</td>
<td>Beta Release</td>
</tr>
<tr>
<td>20%</td>
<td>Final Release</td>
</tr>
<tr>
<td>20%</td>
<td>Final Exam</td>
</tr>
<tr>
<td>5%</td>
<td>Participation</td>
</tr>
</tbody>
</table>

Assignment 1 – Lifecycle Objective – Available now!
Input break!

- What do you most want to learn from the course?

  OR

- What concerns you the most about the course?
Time to explore the class website!

www.cs.washington.edu/education/courses/403/06sp/index.html

- The “Calendar” link will be especially useful to you
What can we expect by the end of 403?

To have:
- Been exposed to some of the best software development practices in use today
- Learned how to more effectively collaborate with others toward a common goal
- Understood how software is produced – from conception to shipping and subsequent maintenance
- Developed skills to articulate your ideas and progress
- Understood the issues and tradeoffs involved in making decisions as software engineers and project managers

Welcome to 403!