Pragmatic Programmer Tip: Care about Your Craft

Why spend your life developing software unless you care about doing it well?
About me

- **Researcher - Software eng and parallel systems**
- **Software Engineer – libraries and debugger**
- **Project lead/Software engineer - libraries and tools**
- **Manager - Programming Environments (PE)**
- **Senior Manager - OS and PE components**
- **Technical Task Manager for the VP of Engineering**

CSE 403, Spring 2007, Alverson
My last project: Red Storm system

- Massively parallel processing supercomputer system used for analysis and stewardship of nuclear weapons - for Sandia National Lab $93M
From XT3 to XT4

- Red Storm was made into a product, Cray XT3
- Full delivery was 3 ½+ years, but got something to the customer in 3
  It was a sprint the whole way, and the team felt it
- Software effort was much more complex than expected
- Rearchitected at least two major SW components after getting
  experience with them

XT3 product was successful and evolved into a product line, the
XT series. Yah!
About the TA’s

- Rosalia Tungaraza
- Alex Zheng
Today’s outline

- Course overview
- Class website

Week readings

- Pitch: Pragmatic Programmer p18-21
- Lifecycle: Survival Guide 54-59, Computerworld article
So what IS software engineering?

From 403 students:

• avoiding Vista fiasco (years late, cut features, morale)
• schedule, milestones, timelines
• design before implementation
• group oriented, teamwork
• managing feature set
• customer involvement
So what IS software engineering?

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 customer
- Managing risk
- Teamwork and communication
What is a software project?

Projects are a balance of three dimensions, with the goal of producing a successful deliverable.
A typical 403 week

1. **Class sessions** to discuss best practices
2. **Sections** to dig deeper and/or discuss pragmatics and tools
3. **Readings and homeworks** 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
1 and 2. Class and Sections

• We’ll present new material, discuss the readings, learn new tools and reflect on experiences
• We’ll have project presentations and demos
• We’ll have multiple industry guest speakers

Interaction and discussion is encouraged!
3. Readings and summaries

Each week:

- Will have a set of required readings associated with it
- Will have a reading summary of one topic assigned

Proposal or Lifecycle due Fri in class

Reading summary assignments:

- topic sentence (s) that states the name, author, article
- at least 3 s containing the main idea/main points
- at least 1 s about how this material relates to 403
- at least 2 s of your own analysis of the paper

Grading is 0-4 scale
4. The Project

- You make project proposals
  - And then vote on which projects to “fund”
- You’re divided into project teams of 6-8 students
  - We choose the teams, to mimic the real world
  - Larger teams, larger projects, like industry
- You develop your deliverable in stages
  - Reflects modern methodologies for effective software project development
- Another team will act as your customer
  - Ultimately, a project will survive only if it satisfies its customer
Project culture

- This is a real project
  - We expect you to work to build a real system
  - To be used by real people

- Take responsibility
  - Take initiative
  - Find and solve problems yourselves
  - Coding is only part of the job
  - Good planning and design, hitting your market, and working well with your team, are all needed for success
Sustainability Project Example

## What is the grading scheme?

<table>
<thead>
<tr>
<th>Percentage</th>
<th>Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>60%</td>
<td><strong>Project:</strong></td>
</tr>
<tr>
<td></td>
<td>Proposal (10)</td>
</tr>
<tr>
<td></td>
<td>Requirements (5)</td>
</tr>
<tr>
<td></td>
<td>Software Design and Planning (10)</td>
</tr>
<tr>
<td></td>
<td>Beta Release (15)</td>
</tr>
<tr>
<td></td>
<td>Final Release (20)</td>
</tr>
<tr>
<td>10%</td>
<td>Project feedback: Design Review, Use and Test Report</td>
</tr>
<tr>
<td>15%</td>
<td>Reading Summaries and Assignments</td>
</tr>
<tr>
<td>15%</td>
<td>Final Exam</td>
</tr>
</tbody>
</table>

Your scores on group related work may be adjusted, based on your contribution.
Input break

- What do you believe will be the biggest challenge of the course?
- How might you (we) address it?
Time to explore the class website

www.cs.washington.edu/education/courses/403/07sp/index.html

- The “Calendar” link will be especially useful to you
Goals of 403:

From students:

- Make something more than toy – something really useful
- Make a successful project that we could commercialize and make a fortune with
- Learn new tools: C#, python
Goals 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