Software Engineering

• The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.

CS 314 Software Engineering

• Methods used to develop large-scale software projects in industry emphasizing design, implementation, and testing.
• Semester “internship”
  – Teamwork
  – Processes
  – Tools
• http://www.cs.colostate.edu/~cs314/

Industry Panel Topics

• Teamwork
  – What teamwork skills do you expect?

• Processes
  – What processes does your team use?

• Tools
  – What tools does your team employ?
Industry Panel Participants

- Heather Ellsworth, Secure64
- Nathan Nuber, HP
- Ron Vaughn, NVIDIA

Similarities and Differences

- Teamwork
- Processes
- Tools
CMMI for Development Model

<table>
<thead>
<tr>
<th>Maturity</th>
<th>Process</th>
<th>Project</th>
<th>Engineering</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>• Organizational Performance Management</td>
<td>• Quantitative Project Management</td>
<td>• Requirements Development</td>
<td>• Causal Analysis and Resolution</td>
</tr>
<tr>
<td>4</td>
<td>• Organizational Process Performance</td>
<td>• Integrated Project Management</td>
<td>• Technical Solution</td>
<td>• Decision Analysis and Resolution</td>
</tr>
<tr>
<td>3</td>
<td>• Organizational Process Definition • Organizational Process Focus • Organizational Training</td>
<td>• Risk Management</td>
<td>• Product Integration</td>
<td>• Configuration Management • Measurement and Analysis • Process and Product Quality Assurance</td>
</tr>
<tr>
<td>2</td>
<td>• Requirements Management • Project Planning • Project Monitoring and Control • Supplier Agreement Management</td>
<td></td>
<td>• Verification • Validation</td>
<td></td>
</tr>
</tbody>
</table>


Topics

<table>
<thead>
<tr>
<th>CMMI</th>
<th>Processes</th>
<th>Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Planning</td>
<td>Scrum</td>
<td>GitHub</td>
</tr>
<tr>
<td>Project Monitoring and Control</td>
<td>Test Driven Development</td>
<td></td>
</tr>
<tr>
<td>Requirements Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integrated Project Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Product &amp; Process Quality Assurance</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Configuration Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Requirements Development</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Technical Solution</td>
<td>Design Patterns • User Experience • Refactoring</td>
<td>UML • Eclipse • Swing / JavaFX / Android</td>
</tr>
<tr>
<td>Verification</td>
<td>Inspections • Black box testing • White box testing</td>
<td>JUnit • EclEmma</td>
</tr>
</tbody>
</table>

Assumptions

• Students meet prerequisites
  – CS 253
  – Implies CS 160, CS 161, and CS 200
• Students know how to
  – Solve problems
  – Design object oriented solutions
  – Program and debug in Java

Expectations

• Lecture preparation and attendance is important.
  – No textbook.
  – Team activities to turn in after each class.
• Projects will be completed incrementally.
  – Ask questions early.
  – Don’t inconvenience your teammates.
  – Don’t wait till the last minute.
  – 20% of your grade is for following the process.
Grading

• Letter grades (A, B, C, D, F)
  – 40% Scrum Projects
  – 20% Scrum Activities
  – 40% Midterm Exams

• No late submissions
  – a few caveats

Scrum Activities – Due This Week

• Submit development team members and GitHub id

• Prerequisite quizzes
  – Object Oriented Programming
  – Data structures
  – Recursion