Engineering 7893
Software Engineering
Introduction

Dennis Peters

Fall 2009
Lectures: Tuesday, Thursday 1200-1250 EN-1000, Friday 1300–1350, EN-4020.

Lab slot: Wednesday, 1400-1650 EN-2048

Office hours: Tuesday 1300–1400, Thursday 1400–1500, or by appointment (or not). Jayde (edmundsj@mun.ca) can make appointments for me.

Website: http://www.engr.mun.ca/~dpeters/7893/
## Dates & Evaluation

<table>
<thead>
<tr>
<th>Date</th>
<th>Item</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fri., September 11</td>
<td>Team selection</td>
<td></td>
</tr>
<tr>
<td>Wed., October 7</td>
<td>Simulator Increment 1</td>
<td>15</td>
</tr>
<tr>
<td>October 27–30</td>
<td>Presentations (individual)</td>
<td>10</td>
</tr>
<tr>
<td>Wed., October 21</td>
<td>Simulator increment 2</td>
<td>15</td>
</tr>
<tr>
<td>Wed., November 4</td>
<td>Controller Layer 1</td>
<td>15</td>
</tr>
<tr>
<td>Wed., November 18</td>
<td>Controller Layer 2 (individual)</td>
<td>10</td>
</tr>
<tr>
<td>Wed., November 25</td>
<td>Competition</td>
<td>15</td>
</tr>
<tr>
<td>Mon., November 30</td>
<td>Final implementation &amp; doc.</td>
<td>20</td>
</tr>
<tr>
<td>Fri., December 4</td>
<td>Peer evaluation</td>
<td></td>
</tr>
</tbody>
</table>
Project Teams

- Four or five members
- Choosers chosen by me:
  - Chad Levesque
  - Shawn Josey
  - Adam Young
- Draft selection process (i.e., choosers take turns) Friday Sept. 11 @ 1330 in EN4020.
- You may form pairs yourself — I’ll try not to break them.
Capture the Flag

**Simulator** — Displays state of game field.
- Communicates with controllers via a well defined protocol.
- Behaviour is carefully constrained.

**Controller** — Directs the movement of one team of players.
- Queries the simulator about the state of the game.
- Sends requests to the simulator to manipulate players.
- Built in two layers — each student will make their own ‘top’ layer.

Each team builds one of each.
All simulator/controller pairs must be interoperable.
• We will use an incremental process.
• Testing and documentation are part of each increment.
• Getting the code working is worth less than half of the marks.
• Design and documentation is worth more.
• Individual student marks will be adjusted from the team mark based on peer-evaluation.
Only a couple of weeks.

- Software engineering principles.
- Software production process.
- Test Driven Development
- Requirements Analysis and Specification
• Each student will give a presentation (20 min) on a topic chosen from a list.
• Presentation notes, with references will be distributed (and graded).
• Goal is that you each learn at least one of the relevant topics in some depth.
• Many people build fences/houses/etc. Is that Civil Engineering?
• Many people build boats. Does that make them Naval Architecture Engineers?
• Many people write programs . . .
Characterizing Craft vs Engineering

- The problems are more complex.
- The problems are bigger.
- The problems haven’t been solved before.
- It’s important that the design be correct (safety).
- Accurate and precise design documents are produced.
- The engineer takes legal responsibility for the design.
This course does **not** teach you to be a Software Engineer any more than any one course in a Mech. Eng. programme teaches them to be mechanical engineers.

- Practice specifying and designing reasonably large systems.
- Exposure to team software development.
- Understanding of the phases of the software development process.
- Exposure to some of the things that make large software projects different from small software projects.
- Exposure to some of the things that makes software engineering different from other engineering.