EECS 394
SOFTWARE DEVELOPMENT

Chris Riesbeck

Finishing
**LAST ITERATION**

<table>
<thead>
<tr>
<th>Monday</th>
<th>Tuesday</th>
<th>Wednesday</th>
<th>Thursday</th>
<th>Friday</th>
<th>Saturday</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVP Final</td>
<td>Demo paths</td>
<td>Product Boxes</td>
<td>5-minute Demos</td>
<td>Meet your clients</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Monday: Feature freeze!**
  - The final iteration is to polish the UI, clean up the code, re-test everything
- **Wednesday: two Bboogle docs by class**
  - Spreadsheet: Demo paths
  - Presentation: Product boxes
- **Friday: In-class demos**
User scenarios (not user stories) that walk through your app's selling points.

- "Joe is a hungry NU junior in a rush to meet his 394 team for a code swarm. He needs to quickly find something on his meal plan that's open and near. [Show the app, do the steps to find the nearest open dining hall.]

- Two short paths is better than one or three.
- If your app isn't that smooth on the scenario, fix the app now!
- If your demo path isn't that exciting or convincing, reflect on what you did wrong in developing your product.
- Do NOT just list features.
PRODUCT BOXES

- See *The Agile Samurai*
- **One** slide with picture
- Who it's for, what it does, why it's great
  - Avoid empty phrases like fun, easy to use, seamless, modern, ...
- Goal: Someone can tell if they want it or not
  - Manage expectations: should be obvious if the answer is "not"
Introduce yourselves, hand out a few hardcopies of the Product Box, then demo.
  - Don't explain. If the demo needs explanation, it's a bad demo.

Main goal: identify first iteration deliverable for client's product
  - Use the Inception Deck to identify key concerns.
  - Get a scenario (demo path) to shoot for.
  - Identify the key value part of the path.

In your team, define
  - a **notetaker** (could be a **product owner**)
  - a **contact person** who sends all emails
At most a few MPD^2 students will know agile.
- Those that do probably know Scrum practices.
- None know how to be an agile client.
- A primary cause of project failure is the client.
  - Weak vision, weak grasp on user value
  - Too dependent on developers
- Teach them to be agile clients through your demo, your product box, and your consistent emphasis on early delivery of value.
The contact person on each team sends all inter-team emails
- CCs everyone on both teams
- gently points out when the other contact person did not CC the teams
- Makes it easy to track communications
- Reduces mixed messages
- Team discusses before contact replies