What We Wish We Had Known (And Other Random Tidbits)

CJ Bell
Peter-Michael Osera

Testing IS Important

- You can't tack it on at the end!
- You have to design around it.
  - i.e., build infrastructure that facilitates testing, rather than impedes it
  - ...in the code (test hooks)
  - ...and out of the code (bugzilla, test group)

Incremental Development Preserves Sanity

- You should wear out the compile key.
- How many lines of code do you write before you hit “compile”?
- How many modules do you integrate before you hit “compile”?
  - Hint: it shouldn't be too much...

Meticulous Planning Is Good

- You can build mountains if you plan it
- Our approach: define the program in terms of interacting components
  - Codify those interactions into Interfaces
  - Research before making decisions
  - e.g., TCP v. UDP

Aim High, Shoot Low

- You will drop features, and it’s ok!
- Keep the features that:
  - Are necessary for your audience
  - Motivate you to finish the project
- Little things make magic happen...

Here’s a Boatload of Magic
Motivate via Communication

- You need to keep talking to each other
  - Silence is death.
- Encourage participation and interaction
  - Ask people what they’re working on
  - Meet as often as possible
  - Use email lists, IRC, IM, whatever...

One-Click Builds are Godly

- You do so much with a one-click build:
  - Facilitate testing
  - Encourage the group
  - Implicitly create infrastructure
  - Not have to worry about $*$!^@*$ compile/packaging @#!& before turn-in
  - Show off 133t scripting skills
  - And so forth...

Writing is a Good Thing™

- You need to learn how to express yourself.
  - We write so little in this department
  - Challenge yourself to write coherently and meaningfully

You Get As Much From This Class As You Put In

- You need to be an active participant
  - It’s easy to think “this is all common sense”
  - ...but can you:
    - Synthesize it all into a coherent whole?
    - Put it into consistent practice?
- It’s tougher than it sounds; start now

The Journey Is More Important than the Result

- You shouldn’t feel (too) bad when
  - You don’t get all the features in
  - Your final build isn’t working quite right
  - Your product looks/sounds/plays crappy

- You should feel bad only when
  - You realize that you haven’t been putting the concepts from class into practice