Software Development at KPR

Heidi Koester, Ph.D.

hhk@kpronline.com
Koester Performance Research
Ann Arbor, MI

www.kpronline.com

Overview

- What we do and why

- How we do it
  - Overall approach
  - Development process
  - Project management

- Challenges & Opportunities

KPR Mission

- "…to enhance computer access, particularly for people who are not well-served by the typical keyboard and mouse."

- Help people with physical impairments access computers more easily and efficiently

- Leverage existing solutions more effectively

KPR Software Products

Name	Purpose
Compass	Support evidence-based decisions in computer access
Keyboard Wizard	Adjust Windows keyboard settings to user's needs
Pointing Wizard	Adjust Windows mouse settings to user's needs

Business Model

- A bit unusual…

- Grant funding ~ 90% of revenue
- We define our own projects, then seek funding for them

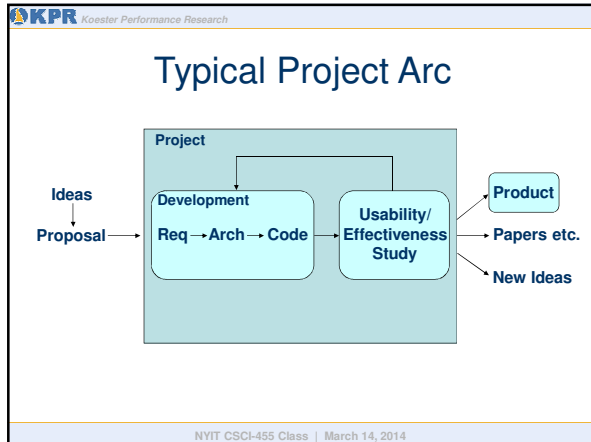- Typically SBIR (Small Business Innovation Research) awards from the National Institutes of Health

Organization

- Try to keep it as simple as possible…

- Build teams with external collaborators on a per-project basis (2 to 8-person teams)

- My own role is to do a bit of everything
- Ultimately responsible for everything that goes out the door: software, user support, research papers, presentations, grant proposals, etc.

1

Typical Project Arc

Project

Ideas

Proposal → Development: Req → Arch → Code → Usability/Effectiveness Study → Product → Papers etc. → New Ideas

Development

- Requirements

- Architecture

- Coding

- Recommend the Pragmatic Programmer series of books

Requirements

- Semi-formal, iterative process
- Document in something like http://www.volere.co.uk/template.htm

- User-centered
- At the very least:
  - Identify key users and use cases
  - Get them interacting with wireframe prototypes

Architecture

- Challenging area – simple yet elegant object-oriented architecture can be tough for a complex application
- Often use consultants for input to this

- Resource like *Head First Design Patterns* is invaluable to build craft in this area

Coding

- Java
- Eclipse IDE
- Unit testing (JUnit)
  - Didn't really "get" that at first, but now it's absolutely crucial
- Integrated testing
  - Written scripts for manual tests prior to release

Project Management

Tool	Purpose
trac	Organize to-do items and project documentation
Tortoise SVN	Version control, code management, backup
Ant	Automate test, build, package steps prior to release
GoToMeeting	Remote collaboration, desktop sharing, user testing, user support

- Necessary even if working solo
- wush.net provides our service

2

Challenges…

- Choosing the most valuable process steps for your project

- Commercial "hardening" is usually tougher than expected (especially for a complicated application like Compass)

- Applying structured approach to the Wild West of web apps (HTML5/CSS3/JavaScript stack)

Opportunities…

- Flexibility to ply your craft on your own terms

- Continuous learning

Final Words

- KPR research & development is supported by:
  - National Institutes of Health
  - U.S. Department of Education
  - Paralyzed Veterans of America Research Foundation
  - TREAT

- Thanks for listening!

- Contact: Heidi Koester
          hhk@kpronline.com