Online Pre-Purchasing

Adam Getchell
acgetchell@ucdavis.edu
https://github.com/acgetchell
http://www.linkedin.com/in/adamgetchell
ucdotnet@ucdavis.edu

TIF

March 28, 2012

University of California Davis

Online Pre-Purchasing in 7 months

- Agile development using Trello task management, GitHub for source code management/collaboration, and TeamCity for Continuous Integration (and eventually, deployment!)
- Battle-tested architecture using JavaScript front end (jQuery, knockout,) UCDArch (MVC, Nhibernate, CAS, etc) middle layer, SQL server backend, SOAP APIs
- Cloud infrastructure using Ganeti, managed by CSE (Bill Broadley) and running in the DataCenter
- UserVoice for feedback
- User training for limited beta commences next week!

Online Pre-Purchasing in 7 months

- (Show pretty graphics on GitHub)

A collaboration

- The Administrative Application Development Initiative led by Tom Kaiser, with heavy input from ADMAN
- The Pre-Purchasing Steering Committee led by Pat Conners
- DaFIS integration, Michael Kuhner and Jonathon Keller
- Training labs, Kelly Crabtree

Programmers

- Trish Ang, designer
- James Cubbage, programmer
- Scott Kirkland, lead programmer
- Alan Lai, project lead and programmer
- Tyler Randles, designer
- Jason Sylvester, programmer and tester
- Ken Taylor, database programmer
- Christopher Thielen, programmer
- Jean Zhou, DaFIS/Kuali API

(see humans.txt)

Systems

- Bill Broadley, architect for cloud services
- Tom Pomroy, systems administrator

What we're doing Right

1. We are starting to listen to the demands of our customers – the "virtuous feedback loop"
2. We are starting to leverage the power of the web
3. We are starting to work together
4. We are starting to use known/good software engineering principles
5. "Central" systems are starting to understand that they provide consumable services to a wide range of applications and end users, and that they need to be flexible (= APIs and documentation)

What we’re doing Right

1. Agile development, Pre-Purchasing Steering Committee, UserVoice
2. HTML5, CSS, Javascript, RESTful web services
3. GitHub, code sprints, frequent meetings with clients (agile development)
4. Solid architecture, code review, test-driven development, continuous integration, instrumentation
5. Application Programming Interfaces, documentation

What we could improve

(Topic for TIF-Ignite presentation)