Report and Analysis of Data

R·I·T Office of Cooperative Education and Career Services

J. Daniel Behun
Chris Nurre
Juan Obes
Michael Yatzkanic

Sr. Project Team

Advisor: Dan Pedersen

Customers:
Manny Contomanolis
Jim Bondi
Project Description

- Reporting and analysis of co-op data
- Track student’s progress
- Maintain employer listings
- Provide co-op data to other offices
Technologies

- Web Services – C# & .NET
- SQL Server 2000
- ASP.NET
- Utilize RIT CAS authentication system
Current System

• System has been in production use since 1999
• Two-tier web application
• Code has become bloated and parts are obsolete
• Productivity is affected by poor system performance
• Architecture limits extensibility and maintainability
Sample Query

Period: 072 To 072

Job Type: Co-op

Major: ALL Majors
- APPE
- APPP
- APRR

*Press Ctrl Key to select multiple majors

Placement only Active, but not placed All
- International students only
- NTID students only
Context & Deployment

- Users
  - RIT OCECS
  - Internal RIT consumers
  - External RIT consumers
- Deployment
  - Full production use by RIT OCECS
Primary Goals

• Three-tiered architecture using web services
• Must be maintainable and extensible
• Increased speed over existing system
• Implement current system features
Three-Tier Architecture

• Separation of concerns
• Extensible
• Maintainable
• Testable
Satisfying Customer Goals

- Separation of responsibilities
- Easier to extend
- No single library file to modify
- Data is now portable to many platforms
Project Scope

- Scoped to the business tier
  - Developing primary services
  - Maintainability
  - Extensibility
  - Performance
- Presentation tier will be handled by OCECS
- Data tier already in place
- Implementing database optimizations
This connection is the contract of passing the auth ticket# supplied by RIT.

Not actually a service, but uses the Interceptor Pattern to catch all calls to services the require authentication.
SCRUM Process

• Incremental development
• Allows for work when requirements are not set
• Visibility and communication
• Customer feedback
NOBY & SCRUM

- Product backlog
- Burndown chart
- SCRUM meetings
- Sprint planning/review
- Frequent customer meetings
Process

- Implementation is done in parallel
  - Team - web services layer
  - OCECS – Presentation layer
- 7 Sprints
  - 1-3: Design
  - 4-6: Implementation & testing
  - 7: Project delivery
Requirements Phase

- Analysis of high level architecture helped solidify requirements
- It took time to find the scope of our requirements in customers broad vision of final system
Final Requirements

- Elicitation was more a continual growth of understanding rather than iterations
- 4 Primary Web-Services
- 69 functional requirements of the system
- Requirement volatility has been minimized during the requirements phase
Testing

- Automated testing
  - JMeter
- Accuracy
  - Compare result sets from old system
- Parallel development will act as functional testing
Project Status

- Current
  - Finalized requirements
  - Finished design and architecture
  - End of Sprint 3

- Future / spring quarter
  - Implementation
  - Testing
Metrics

- Effort Metrics
  - Estimation accuracy
- Progress Metrics
  - Slippage
    - Plan accordingly to make sure deadlines are met
- Earned Value
  - Make sure the team is progressing forward
Quality Metrics

- Lines per code per module
- Query response time
  - use results from automated testing to compare improvement
Task Estimation

- Project plan & product backlog list all tasks for each sprint
- Task duration is estimated during sprint planning sessions
- On average, each item ran over 30%
Slippage

- Slippage metrics are important to prevent senioritious from setting in and causing a build up of work near graduation.

- Week 4: Sprints were pushed back by 2 days to add the next weekends to the sprints
  - Done to give extra time after meeting with the Co-op office on Thursday

- Sprints overall have not be delayed
  - Items have been shifted around in general between sprints but the planned workload has remained consistent
Earned Value Metrics

- Points are assigned for every item that needs completion
  - Project Plan
  - Architecture Document
  - Requirements Document
  - Diagrams
  - Customer Meetings / Interviews
Risks

- Understanding system
- Feature creep
- Parallel development with presentation layer
- Time management
- Senioritis
Reflection

• System learning curve was steep which slowed requirements elicitation

• Creating the architecture helped to understand the system and drove requirement elicitation

• Frequent meetings with customer helped
Questions?