Job Zone Reporting

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

Sr. Project Team

J. Daniel Behun
Chris Nurre
Juan Obes
Michael Yatzkanic

Advisor: Dan Pedersen

Customers:
Manny Contomanolis
Jim Bondi
Project Overview

• Office of Cooperative Education and Career Services

• Data reporting and analysis system
  • Tracks student’s progress
  • Maintains employer listings
  • Provides job related statistics
Customer Problem

- Current system has been in production use since 1999
- Two-tier architecture limited extensibility
- Code bloat negatively affected performance
- The system is not easily maintained
Customer Needs

- Extensible & maintainable system
- Data available to outside clients
  - Housing
  - Financial Aid
  - Companies
- Security to protect sensitive data
- Performance increase over current system
Satisfying Needs

• Created 3-tier architecture
• Extensibility and maintainability
• Web Services
  • SOAP allows for wide access to data
  • Interceptor secures all services
  • CAS used for authentication
• Reduced bloat of two-tier Architecture
Extensibility

- **OCECS Server (ASP pages)**
- **Outside Client**
- **CAS Authentication (webeapps.rit.edu)**
- **OCECS Web Services**

**OCECS Authentication**
- **Basic Setup Service**
- **Placement Service**
- **Student Service**
- **Activity Service**
- **Employer Service**

**Database Connection Abstraction**
- **OCECS Datasource**

**Alert Service**
- **Portal Service**

**Future Services**

---

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 that require authentication.
Multi-client Support

- SOAP allows for access to data for a wide range of possible clients
- Client must conform to SOAP & security model

DEMO
Security

- Use of SSL Certificates
- Make use of existing DCE authentication using Central Authentication System
- Hold session data in server memory to prevent injections
Performance

• Performance measurements were based on load times of equivalent requests
• Over all requests there was a 41% decrease in load time
Tradeoffs

- Using web services does not necessarily mean performance increases
- Performance increase was more a result of implementing cleaner and more modular code
- Security forced web services to be stateful
Requirements

- Implemented all requirements in SRS
- New features
  - Gender searching
  - Searching for international companies with international students as Co-ops
Status

• Acceptance testing is in progress as the OCECS staff makes prototypes of existing functionality

• Full unit/functional test suite

• Deployment code delivered Tuesday, May 5th

• API documentation done
To-Dos

- Clean up code
- Commenting
- Debugging relics
- More extensive load tests
- Technical document
Deployment

- IIS configuration settings
- SSL
- .NET configuration
- Application connection Strings
- SQL server
Future work / goals

• OCECS will take over development

• Additional services to be added
  • Alert & Portal Services

• Finish the presentation tier for current services

• Work with outside clients
  • Financial Aid

• Housing
What went well

• Frequent communication with customer
• Established architecture design first
  • Drove the requirements
  • Reinforced our understanding of system
• Extra time spent to finalize SRS
  • This prevented scope creep and changing requirements
Challenges

- Steep learning curve
- Consumed majority of winter quarter
- Challenging requirements elicitation
- Defining an appropriate scope for the time allotted for the project
- Securing the system