

# Graduate Student Information System (gSIMS) Walkthrough

Kartik Thakore<sup>1</sup>

<sup>1</sup>Department of Software Engineering  
University of Western Ontario

23 Nov 2010

# Outline

- 1 Introduction
  - Project Details
- 2 Requirements
  - Technical Requirements
- 3 Architectural Analysis
  - Analysis
  - Architecture
- 4 Iterative Design
  - Iteration 1

# Outline

- 1 Introduction
  - Project Details
- 2 Requirements
  - Technical Requirements
- 3 Architectural Analysis
  - Analysis
  - Architecture
- 4 Iterative Design
  - Iteration 1

# Project Inception

- Advisor: Dr. Hanif Ladak
- Concerned with managing students in the BioMedical Physics Graduate Student at UWO.
- Current system have the following problem:
  - Calculations and updates can only be done by one person
  - Need to keep the paper copies of meetings
  - Takes lots of time to create reports
  - Hard to track when a student must have a requirement done

# Current System

Demo of the Current System.

- Benefits:
  - Faster, less error prone way of tracking graduate program
  - Possibility of expanding to other department
  - Credibility to the graduate program

# Project Organization

Two components of the project:

- (ECE4416) Business rules:
  - Graduate program milestones
  - Interface requirements with the user
- (SE4450) Technical requirements:
  - Provide the functionality for the User Interfaces.
  - Adhere to required constraints.
  - Make a pilot for these Business Requirements with expansion in mind.

# Proposal

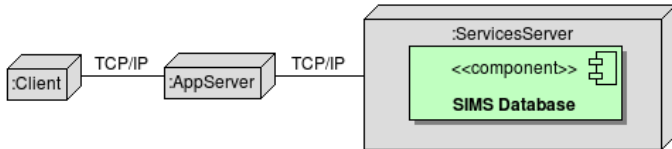


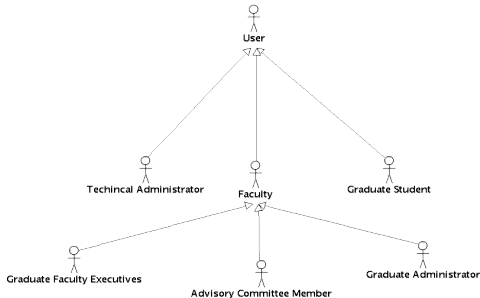
Figure: The proposed system

# Outline

- 1 Introduction
  - Project Details
- 2 Requirements
  - Technical Requirements
- 3 Architectural Analysis
  - Analysis
  - Architecture
- 4 Iterative Design
  - Iteration 1



# System Roles



**Figure:** Roles of the SIMS system

*School of Graduate and Postdoctoral Studies*

*The University of  
Western Ontario*

# Interfaces

- Graphical User Interface:
  - The implementation of the Business Rules defined as HTML pages.
- Electrical User Interface:
  - Collect signatures from a Wacom ©Tablet and store securely in the DataBase.

# Graphical User Interface

Specific requirements for the view of the Web Pages:

- Set of HTML pages that are to be the template of the system.



Figure: Sample GUI provided

# Electrical Device Interface

- Provide an interface for the User to sign on the screen.
- On the client side acquire a bitmap of the signature and encrypt the bitmap data.
- The image should be viewable only by the user who signed and the graduate admin.

# System Features

- User Administration
- Tracking Data
  - Student Data
  - Student Term and Funding Data
  - Student Program Data
  - Student Advisory Committee Meeting
- Reporting
  - Customized Queries
  - Student Output Reports
- Triggering System

# Constraints

- Agile
- Flexibility
- Easy to use
- Roles and Operational Access
- Protocol Security

# Outline

- 1 Introduction
  - Project Details
- 2 Requirements
  - Technical Requirements
- 3 Architectural Analysis
  - Analysis
  - Architecture
- 4 Iterative Design
  - Iteration 1

# Organizing Data

- Separate Authentication Data from Critical Data



Figure: 3 general data use cases

*School of Graduate and Postdoctoral Studies*

*The University of  
Western Ontario*



# Conceptual Model of the Student

- Student can be treated as a ticket, which needs to go through steps to be completed.

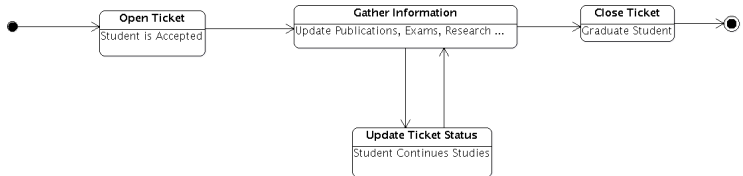


Figure: The simple steps of Grad School

*School of Graduate and Postdoctoral Studies*

*The University of  
Western Ontario*

# Critical Assumptions

- Separation of Data
  - Student Data is kept around even after student has graduated.
- Student as a Bug Ticket
  - Student will be responsible for their own data.
  - Student can only be a student if they have a funded term.
- Data entry will be done manually at this point.

# Outline

- 1 Introduction
  - Project Details
- 2 Requirements
  - Technical Requirements
- 3 Architectural Analysis
  - Analysis
  - **Architecture**
- 4 Iterative Design
  - Iteration 1

# Hardware

- Clients
- Application Server
- Services Server

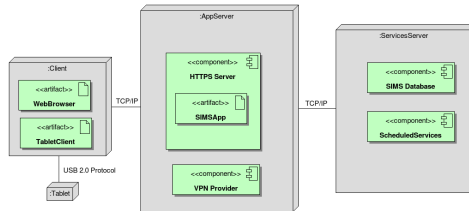


Figure: The System Overview

*School of Graduate and Postdoctoral Studies*

*The University of  
Western Ontario*

# Software

- E-Signature Capture Client
- OpenVPN Server
- Production HTTP server
- Database Server
- Perl Modules

# Outline

- 1 Introduction
  - Project Details
- 2 Requirements
  - Technical Requirements
- 3 Architectural Analysis
  - Analysis
  - Architecture
- 4 **Iterative Design**
  - **Iteration 1**

# Rapid Prototyping

- Perl Framework
  - A glue that abstracts the database, and templates the view
  - Controllers are defined as routes
  - Representational State Transfer (REST)
- Database Schema
  - Intrinsic Student Data
  - Role Based Authentication
- E-Signature client

# Test Plans

- Unit Tests
  - Demo
- Integration Testing
- System Integration Testing



# Summary

- Requirements and Analysis has received direct user feedback.
- Architecture based of the Analysis has been clarified and prototyped.
- The iterative Software Life Cycle has produced useful work quickly and with less effort.
- An emphasis on testing levels is present from the starting.