

# 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 graduate program for BioMedical Physics.
- Current system has lots of problems.
  - Calculations and updates are mostly manual.
  - 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.

# Project Organization

Two components of the problem:

- (ECE4416) Business rules:
  - Graduate program milestones and dataflow.
  - Direct interaction with the User.
- (SE4450) Technical requirements:
  - Provide the functionality for the User Interfaces.
  - Adhere to required constraints.

# Proposal

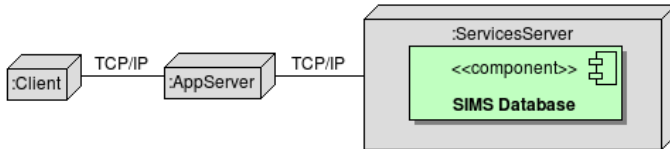


Figure: The proposed system

# Outline

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



# 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

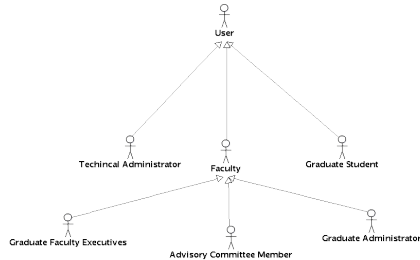
- Provide an interface for the User to sign on the screen.
- On the client side acquire a bitmap of the signature and encrypt the bitmap data.
- The image should be viewable only by the user who signed and the graduate admin.

# 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

- Security
  - System Security
  - Roles and Operational Access



*School of Graduate and Postdoctoral Studies*  
**Figure: Roles of the SIMS system**

# Outline

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

# Organizing Data

- Seperate 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.

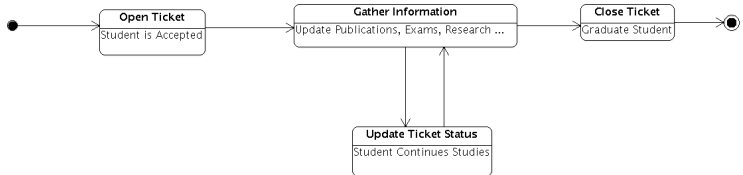


Figure: The simple steps of Grad School

*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

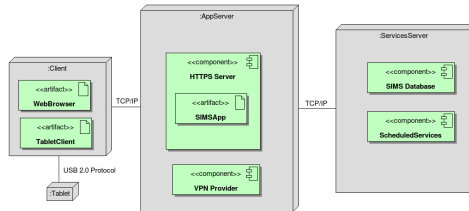


Figure: The System Overview

*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
- Database Schema
- E-Signature client

# Test Plans

- Unit Tests
- 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.
- A strong emphasis on 3 testing levels is present from the starting.