Interactive Graduate Student Information Database

Midterm Report

Kartik Thakore (250313003)   *Parth Champaneri (250367669)*

kthakore@uwo.ca   *pchampan@uwo.ca*

Abstract

This report aims to provide preliminary elicited requirements for the Interactive Graduate Student Information System (SIMS). Additionally a walk through of the first iteration is included. We started work on the first iteration which includes two 2 critical features of SIMS. Data schemas and logic for Authentication and Student Funding were accomplished. Using the analysis of elicited requirements, data flow requirements, and verified assumptions a database schema was designed. After normalizing the database schema, a Representational State Transfer (REST) framework was implemented to combine components. With a preliminary implementation in place unit and integration testing were performed. A sign-off from end users and clients is required to prepare from the next iteration. However the components for the next iteration have been decided to include the E-signature component for the advisory meeting tracking feature. In preparation for iteration 2 we have begun rapid prototyping of the E-signature component.

1   Specific requirements

1.1   External interface requirements

1.1.1   Hardware interfaces

Electronic Signature Pad

> A hardware interface will be required to record signatures from several roles for electronic records of forms. The data from the signature pad must be encrypted as it is read, and saved in an appropriate protocol on the DB.

1.1.2   Software interfaces

Web Interface

> The web interface will be responsible for providing users with access, and the overall functionality of the systems. That is to say users will only be able to access the interface via the web interface. In addition, the web interface will require a specific web browser to be used. The constraint on the browser will prevent any problems with compatibility issues across the various devices used by the client.

1.1.3 Communications interfaces

Network

The network interface will provide security to the database, so as to prevent access from unknown devices. Additionally any transfer of data between the user and the system must be placed in a secure channel/protocol.

1.2 System features

1.2.1 Graphical User Interface

1. Purpose: Provide a dossier format interface for users to see all relevant data of the student in a centralized location.

2. Response sequence: User will login, and be able to search a student or go through a table of all available users. Selecting a student will bring the user onto this interface.

3. Associated Functional Requirements:

   (a) Drill down: Functional Requirement 1

       i. Sections: each student will have multiple sections that can be disabled or enabled.
       ii. Expansion: sections will expand to show summary of addition information.
       iii. Link: sections will link to sections pages for more detailed information.

1.2.2 Term Calculations

1. Purpose: Calculate dates and event times for graduate student programs

2. Response sequence: When a student profile and program is created the system will make triggers for relevant events to each milestone.

3. Associated Functional Requirements:

   (a) Milestones: Functional Requirement 1

       i. Calculate start - end Semester dates for Graduate Students
       ii. Calculate due dates and triggers for milestones

   (b) Funding Calculations: Functional Requirement 2

       i. Track funding availability for each semester and the source of funding
       ii. Show next date for major funding applications

1.2.3 Tracking

1. Purpose: Track Major Milestones (grants, publications, exams etc) for graduate students through the program.

2. Response sequence: Data is updated according to business rules and workflow of the program and the student's progress.

3. Associated Functional Requirements:

    (a) Publications and Grants: Functional Requirement 1

        i. Track student publications that have been published

        ii. Track grants that have been received by the student

    (b) Advisory Committee Members: Functional Requirement 2

        i. Send trigger to student to form an Advisory Committee

        ii. Allow students, and advisory committees to store and track comments and discussions

        iii. Show calendar view of all meetings and results of the meetings

        iv. Allow for single or joint supervisors

        v. Track electronic submissions of advisory meeting form

    (c) Manage Milestones: Functional Requirement 3

        i. Handle and process milestones for the Masters program in BioMedical Physics at UWO

            A. Form advisory committee by end of 1st term

            B. Annual seminars

            C. Low-level exams for new students

            D. Exams are organized by department

            E. Exams usually in late June; informed in early May

            F. Possible MSc to PhD reclassification

            G. Discuss reclassifications with supervisor and advisory committee first

            H. Reclassification must be completed before end of 5th semester

            I. Submit and defend MSc thesis if not reclassified

            J. `http://www.uwo.ca/biophysics/grad_program_policies/guidelines_intro.htm`

    (d) Send Triggers and Receive Responses: Functional Requirement 4

        i. Process conditional and requested triggers

        ii. Allow Faculty Advisor to create and view all triggers

        iii. Conditional triggers are event based automatic or triggered conditions

        iv. Requested Triggers are created by users and their activities on the system

        v. The system should allow responses to each Triggers be collected and stored

        vi. Responses should be accessible by relevant users only

1.2.4   User Layers and Collaboration

1. Purpose: Ensure ad-hoc access for multiple users to facilitate realtime collaboration and ensure up-to date information in the database. Permissions map to prevent unauthorized access and control the scope of data.

2. Response Sequence: Multiple users will be able to log in simultaneously and information will update in realtime.

3. Associated Functional Requirements:

    (a) User Group Permission Map

        i. Graduate Students: This group of users will be able to log in and able to edit, update and save their demographic information and other program information including Advisory Committee members, Publications, Thesis etc.

        ii. Advisory Committee Members: This group will be able to comment and provide feedback on a student's advisory committee meeting output. Ideally, other student information will be restricted for changes.

        iii. Graduate Executives: This group of users will primarily utilize the generated reports for planning and information purposes. Access will be restricted to viewing information. Will be taken to a Project dashboard where they will have a bird's eye view of reports and information statistics. Read-Only Access.

        iv. Graduate Affairs Assistant: Key stakeholder for the system. WIll be able to manage, administer and access all informational program data. Access to change log. Generate reports in Excel, PDF etc.

        v. Technical Administrator: Primarily responsible for system administration, periodic maintenance schedule and providing technical assistance to users. Ability to reset system passwords and create users.

1.2.5 Security

1. Purpose: To build a secure system that adheres to local and federal privacy laws (FIPPA)

2. Response Sequence: Any data inputted into the system will be encrypted and all passwords will be stored using hash.

3. Associated Functional Requirements:

    (a) To be defined

1.2.6 Database design

1. Purpose: Improve Error detection and stability of the system.

2. Response Sequence: The information will be normalized and data redundancy will be introduced

3. Associated Functional Requirements:

    (a) Data Normalization: Systematic way to ensure that the design is free from any undesirable characteristic - insertion, update, and deletion anomalies that could lead to the loss of data integrity.

    (b) Data Redundancy to improve error detection

1.3 Performance requirements

1.4 Design constraints

1.5 Software system attributes

1.6 Other requirement

2 Iteration 1

2.1 Features

2.1.1 Authentication

Multiple user access to the systems from an organization role aspect.

2.1.2 Student Funding

Feature to archive student funding data, calculate ending term and provide reports.

2.2 Analysis

2.2.1 Roles and Operation Analysis

Organizational roles and what need

2.2.2 Data Flow Analysis

What are the criteria for our DFD components. Concerns ..

2.2.3 Relationship Analysis

Critical Assumptions, Scope checking

2.3 Design

2.3.1 Use Class Diagrams

What can a user do

2.3.2 Access Control List

Who can use what

2.3.3 Data Flow Diagrams

Sources, processes, Sinks

2.3.4 Class Diagrams

Auth.PNG and StudentFunding.PNG

2.4 Implementation

2.4.1 Catalyst Framework

Chained Operations

2.4.2 Rapid Database Prototyping

SQLite database and rapid schema changing

2.5 Testing

2.5.1 Regression Testing

Unit Tests

2.5.2 Acceptance Testing

User Tests and feedback

3 Iteration 2

Add Student milestone tracking.

3.1 Rapid Prototyping

For signature Pad

4 Progress

4.1 Updated Gantt Chart

4.2 Changes

References

[1] Joe Lin, Charley Ho, Wasim Sadiq, and Maria E. Orlowska, "On workflow enabled e-learning services", *Advanced Learning Technologies, IEEE International Conference on*, vol. 0, pp. 0349, 2001.

[2] "Council highlights", `http://www.cou.on.ca/News/News---Views/Newsletters/PDFs/Council-Highlights-2005May.aspx`, 2005, This is an electronic document. Date of publication: May 1, 2005. Date retrieved: Sept 29th, 2010.

[3] I.P.W. Fung, "On monitoring study progress with time-based course planning", *Advanced Learning Technologies, 2001. Proceedings. IEEE International Conference on*, pp. 361 –364, 2001.

[4] Office of the Privacy Commissioner of Canada, "Canada's personal information protection and electronic documents act", `http://www.priv.gc.ca/information/guide_e.cfm`, 2003, This is an electronic document. Date of publication: November 20, 2003. Date retrieved: October 4, 2010. Date last modified: [Date unavailable].