

COMP354 Course Project: Online Lab Management System Project
Version 1, Dec. 15, 2008
Dr. Yuhong Yan

1. Background

Recent advances in network, eScience and eLearning have lead to growing demand for conducting experiments online. The experiments can be scientific research experiments or experiments for high-education. Online experiments have the advantages of sharing expensive equipments, reducing traveling cost, and accessing experiments around the clock and from anywhere. Online lab management system is a software system promises the functions of 1) managing lab resources online; 2) providing accessibility to lab resources over Internet.

Canarie has given 1.4M budget to UWO, Concordia, CLS and IBM to build a service-oriented framework for online experiment platform within 3 years. Prof. Yuhong Yan is working on the service-oriented architecture for the Canarie Science Studio project.

2. The Objectives of COMP354 Course Project

COMP354 course project uses online experiment application as a background to train the undergraduate students for software engineering skills. The students will go through the full software engineering process, more specifically, following the iterative development process-Unified Process, to build a prototype online lab management system.

Compared to the Canarie Science Studio project, COMP354 course project is a small scale trial project. It mainly serves as a training project for the students. We may not expect scientific value from the results of the project. But from my past observation, our students are too much trained in well managed environment. They are lack of the real world experiences to face uncertainties and challenges in a real world projects. Also, their programming and development skills are not sharpened to deal with complex frameworks and multi-module projects. As COMP354 is the only and the last software engineering course for computer science graduates, I would like to design this project to bring more real world experiences to the students. Therefore, the project is open – the students will meet clients, make decisions for design and development, and implement the platform with real world products. The students will not stay in classroom. It is not a toy project – it is open-end. The students will be evaluated by their performances in different stages of this project.

COMP354 course project will investigate the general user requirements for all engineering and science domains which require experiments for research or teaching, and develop an online lab management system to serve the purpose. The students will be guided through the software engineering process and be evaluated by their performances in different stages of this project.

3. About the User Requirements of the Online Lab Management System

We have worked for several years on online lab management system. Here are user requirements for the online lab management systems in literatures. This just provides background information for the students. The students should generate user requirements from the clients directly.

a) Online lab resources management

Lab resources include all the computation and experiment equipments and lab spaces. For example, computers, instruments, machines are all lab resources. An online lab resources management module will monitor the status of the resources (availability and booking status), manage experiment/resource booking, report the statistic information (e.g. serving hours, availability rate), etc.

b) Training tutorial management

Complex equipments have tutorials for their operation. Online experiments for teaching also have tutorials to tell the students how to do an experiment. The tutorial management module manages the tutorials like an online learning system. An example can be the Moodle used by Concordia for online tutorial/course management.

c) User management

Users can have the roles of clients (who do the experiments), lab manager (who manages the lab), experts (who accesses the system to give advices). Different roles have different authorities to access different functions.

d) Remote operation of equipments

Many equipments have digital interface, i.e. the capacity to connect to network and communicate with software. It is possible to send commands and receive data to/from the equipments. Some research is on this topic.

e) Collaboration tools

Wikis, online chatting, instant messages and online video streaming are used to connect scientists/students to discuss their experiments and exchange ideas in real time.

f) Data management

An experiment usually produces a lot of data in digital format. The issues are how to store the data, how to allow the user access the data after experiment, how to transfer the data to another site.

f) Security

The lab should be accessible to some authorized parties. The communication should be secured.

4. The management of the project

Team organization:

a) Each team is composed by 4-5 students.

b) Roles: one team leader for organizing team meeting and communicating with all team member, one presenter for reporting to the class, one writer to record team meeting and write documents, two advisers to give ideas. All the team members have to share development and design work, i.e. each one plays the role of architect and developer. The workload of each team member has to be agreed and tracked. The report of the actual workload will be signed by all the team members.

c) Re-organizing the team: at each iterative phase, the team will be voluntarily re-dissolved and re-organized. The students should switch roles in each team. Within a phase, the team should keep stable and fulfill its objective.

Evaluation:

Workload share (25%): from project plan and workload sheet

Product quality (25%): from the function demonstration at the end of the project, and the code submitted

Team organization (25%): report, presentation, team meeting

Client feedback (25%): clients will be invited to review the functions of the module developed

5. Project plan and deliverables

5.1 Inception: approximate vision, business case, scope, vague estimates

Tasks:

- a) read the paper on how to conduct the interview (week 2);
- b) draft and submit the questionnaire of interview before meeting the clients (week2);
- c) conduct a 20 minutes interview (Interview 1) with the clients to know the domain and business (week 3);
- d) write requirements document (requirements document 1) (week3);
- e) pick a subset of user requirements to design, build and test (week3); (note: it may not be realistic to build and test the code. More realistic is to think about the design, architecture and make a plan of design and development);
- f) class meeting and presentation;

Deliverables (submissions by group):

- a) questionnaire of Interview 1
- b) requirements document 1
- c) project plan document 1: include the scope of the project, the selected use case for design and development, technique analysis, team plan in Gantt chart.

Time line: precise deadline of deliverables will be decided later.

5.2 Iteration 1 (Elaboration Phase): refined vision, iterative implementation of the core architecture, resolution of high risks, identification of most requirements and scope, more realistic estimates.

Tasks:

- a) Select one use case and interview the client for 20 minutes for business logic of this use case (Interview 2) (week4)
- b) Decision making for architecture, select proper development tools, and set up development environment (week 4);
- c) detailed design and development for the use case (week 5, 6)
- d) report to client (week 8)
- e) class meeting, presentation, and demonstration (week 8)

Deliverables (submissions by group):

- a) interview preparation and report
- b) project plan document: project scope, project plan (including workload report), architecture, development environment, domain models,
- c) domain model
- d) use case model
- e) data model
- f) class diagram
- g) a report to client (simple 1 page)

Time line: precise deadline of deliverables will be decided later.

5.3 Iteration 2 (Construction Phase): iterative implementation of the remaining lower risk and easier elements, and preparation for deployment.

Tasks:

- a) interview the client again for 20 minutes (Interview 3), report the progress, refine the first use case, (select a 2nd use case for implementation) (Week 8)
- c) refine the first use case (add more scenarios) (week 9)
- d) design and development with testing functions (week 10, 11, 12)
- e) deployment (week 12)
- e) report to client (week 13)
- f) class meeting, presentation, and demonstration (week 13)

Deliverables (submissions by group):

- a) project plan document: project plan (including workload report)
- b) domain model
- c) use case model
- d) data model
- e) class diagram
- f) code
- g) a report to client (simple 1 page)

6. Appendix documents

- a) how to conduct interview
- b) Interview 1 sample questions