**Requirements Specifications**

**InPress – Group 3**

Liu Wenbo (0970709)

Justin Kan (0843763)

Abdul Hadi Muhammad (0951862)

## 

Table of Contents

Change History 4

Project Drivers 5

THE PURPOSE OF THE PROJECT 5

The User Business or Background of the Project Effort 5

Goals of the project 5

THE CLIENT, THE CUSTOMER AND OTHER STAKEHOLDERS 5

The Client 5

The Customer 5

USERS OF THE PRODUCT 5

The Hands-On Users of the Product 5

Priorities Assigned to Users 5

User Participation 5

Maintenance Users and Service Technicians 5

Project Constraints 6

MANDATED CONSTRAINTS 6

Solution Constraints 6

Implementation Environment of the Current System 6

Partner or Collaborative Applications 6

Off-the-Shelf Software 6

Anticipated Workplace Environment 6

Schedule Constraints 6

Budget Constraints 6

NAMING CONVENTIONS AND DEFINITIONS 6

Definitions of all Terms 6

RELEVANT FACTS AND ASSUMPTIONS 7

Facts 7

Assumptions 7

Functional Requirements 7

THE SCOPE OF THE WORK 7

The Current Situation 7

The Context of the Work 7

Work Partitioning 8

THE SCOPE OF THE PRODUCT 12

Product Boundary 12

Product Use Case List 12

Individual Product Use Cases 13

FUNCTIONAL AND DATA REQUIREMENTS 17

Functional and Data Requirements 17

Nonfunctional Requirements 21

LOOK AND FEEL REQUIREMENTS 21

Appearance Requirements 21

Style Requirements 21

USABILITY AND HUMANITY REQUIREMENTS 21

Ease of Use Requirements 21

Personalization and Internationalization Requirements 21

Learning Requirements 21

Understandability and Politeness Requirements 21

Accessibility Requirements 21

PERFORMANCE REQUIREMENTS 21

Speed and Latency Requirements 21

Safety- Critical Requirements 21

Precision or Accuracy Requirements 22

Reliability and Availability Requirements 22

Robustness or Fault- Tolerance Requirements 22

Capacity Requirements 22

Scalability or Extensibility Requirements 22

Longevity Requirements 22

OPERATIONAL AND ENVIRONMENTAL REQUIREMENTS 22

Expected Physical Environment 22

Requirements for Interfacing with adjacent Systems 22

Productization Requirements 22

Release Requirements 22

MAINTAINABILITY AND SUPPORT REQUIREMENTS 23

Maintenance Requirements 23

Supportability Requirements 23

Adaptability Requirements 23

SECURITY REQUIREMENTS 23

Access Requirements 23

Integrity Requirements 23

Privacy Requirements 23

Audit Requirements 23

Immunity Requirements 23

CULTURAL AND POLITICAL REQUIREMENTS 23

Cultural Requirements 23

Political Requirements 23

LEGAL REQUIREMENTS 23

Compliance Requirements 23

Standards Requirements 24

Project Issues 24

OPEN ISSUES 24

OFF-THE-SHELF SOLUTIONS 24

NEW PROBLEMS 24

TASKS 24

MIGRATION TO THE NEW PRODUCT 24

RISKS 24

COSTS 24

USER DOCUMENTATION AND TRAINING 24

WAITING ROOM 24

IDEAS FOR SOLUTIONS 24

Requirements Timeline 25

Table of Figures

Figure 1: Context of the Work 7

Figure 2: Work Partitioning 11

Figure 3: Product Use Case List 12

Figure 4: Create Instructor's Account Use Case 13

Figure 5: Create a Course Use Case 13

Figure 6: Create Questions Use Case 14

Figure 7: Post and Unpost an Assessment 14

Figure 8: Answer the Question Use Case 15

Figure 9: Check the Results Use Case 15

Figure 10: Question Data Analysis Use Case 16

Figure 11: Student Account Registration Use Case 16

Figure 12: Add and Removing Assessments 17

Figure 13: Volere Cards 20

# Change History

1. The following table shows the change history for this test plan document.

|  |  |  |  |
| --- | --- | --- | --- |
| 1. Version | 1. Date | 1. Author | 1. Comments |
| — | — | AHM, WL, JK | Original content. |
| 0 | October 10, 2013 | AHM, WL, JK | Initial check in |
| 1 | February 25, 2014 | AHM, WL, JK | Revision 0 Changes |
| 2 | April 11, 2014 | AHM, WL, JK | - Modified the structure of the document (added table of contents, change history, figure tables, etc)  - Added some requirements missing from initial check-in  - Figures and Diagrams were redesigned  - Non-Functional and Functional Requirements were revisited |

Table 1: Change History

This document follows the Volere Requirements Specification Template. This template may be found at <http://www.volere.co.uk/pdf%20files/templateArchive.zip>.

# Project Drivers

## THE PURPOSE OF THE PROJECT

### The User Business or Background of the Project Effort

Design a web interface called “InPress”, which allows students in various universities to respond to real-time questions posted by their instructor using any internet-enabled device (i.e. iPads, iPhones, Android Phones/Tablets, PCs, Macs).

### Goals of the project

There are two goals in this project:

- Provide students with a free classroom interaction tool that will allow them to participate in classroom activities

- Allow students to learn more efficiently

## THE CLIENT, THE CUSTOMER AND OTHER STAKEHOLDERS

### The Client

Kevin Dunn, Chemical Engineering

### The Customer

Customers include colleges/universities, instructors and students.

## USERS OF THE PRODUCT

### The Hands-On Users of the Product

- Students and Instructors will be connecting to the web interface in order to interact with the various features.

- System Administrators will need to configure and setup the software environment through a web interface.

### Priorities Assigned to Users

- Key Users: Instructor and Students

- Secondary Users: System Administrators

### User Participation

- Students will answer questions posted by their instructor

- Instructors will be able to populate the system with questions

- Instructors will be able to analyze data received through student interaction

### Maintenance Users and Service Technicians

Maintenance users will be mainly System Administrators at the college/university, and Software Developers at InPress. System Administrators at the college/university will have to deal with any kind of operating discrepancy in the environment where the code is hosted.

Any other problems (i.e. a bug in the code) will first need diagnosing by the System Administrators, and if need-be a service ticket must be opened against the Development team at InPress.

# Project Constraints

## MANDATED CONSTRAINTS

### Solution Constraints

*Description*: InPress is a web-based system

*Rationale:* The InPress software will only be accessible via an Internet enabled device

*Fit Criterion*: Users need to use the Internet to communicate with InPress

### Implementation Environment of the Current System

The environment in which this system is going to be hosted is McMaster University. We will be hosting it on a departmental server (inpress.mcmaster.ca) in the Engineering Faculty. This software will be portable to other universities, but the software will have to be hosted internal in that university.

### Partner or Collaborative Applications

- Kevin Dunn, Chemical Engineering and Client

- Chemical Engineering System Administrators

- Dr. Rong Zheng, CAS Supervisor

- Dr. Spencer Smith, CAS Supervisor

### Off-the-Shelf Software

* **i-Clicker®:** Electronic devices that allows users to interact with the host through five buttons (A, B, C, D, E). The product supports multiple-choice only.
* Similar products already exist in the market (i.e. iClicker®). Most of the products only allow students to interact with the instructor, and do not have much functionality.

### Anticipated Workplace Environment

Colleges and Universities

### Schedule Constraints

All deliverables must be done within the deadline. The expected completed date is April 30/2014.

### Budget Constraints

N/A

## NAMING CONVENTIONS AND DEFINITIONS

### Definitions of all Terms

Students:People currently enrolled in an undergraduate or graduate program at a University/College

Instructor: An individual teaching an undergraduate/graduate course

System Administrator: Technical staff dealing with computers in the University/College

## 

## RELEVANT FACTS AND ASSUMPTIONS

### Facts

- Current popular implementations (i.e. iClicker) allow students to answer multiple-choice questions only

- Current popular implementations (i.e. iClicker) require physical hardware

### Assumptions

- Hardware will be self-provided by the customer, and hardware failure will be addressed by on-site university IT personnel.

- We assume that the users have Internet access whenever they want to use the product

- Assume that students have some minimal knowledge (i.e. know the English language, know their Student No., etc.)

# Functional Requirements

## THE SCOPE OF THE WORK

### The Current Situation

- The software (web application) will require students and instructors to login, so that their responses will be saved

- The saved response can be used for data analysis (i.e. Number of Correct Responses)

- Students can use the web-application to respond to Instructor’s questions

- Easy setup for instructors by entering the questions before the lecture

- Students can review their answers, and the respective solutions after lecture

### The Context of the Work

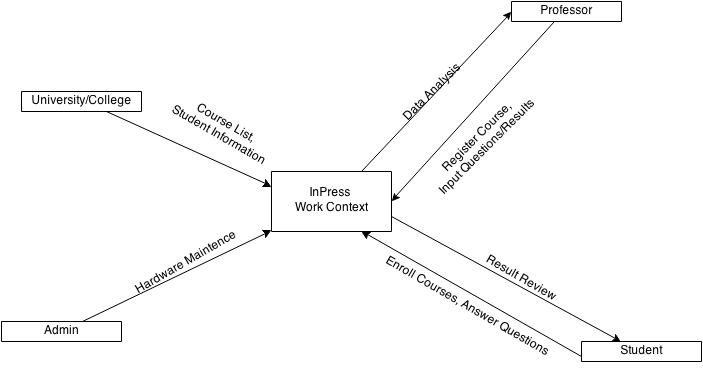


Figure 1: Context of the Work

### Work Partitioning

|  |  |  |  |
| --- | --- | --- | --- |
| **R#** | **Event Name** | **Input and Output** | **Summary** |
| R1 | Create an Instructor Account | Instructor’s Credentials (IN)  Created Account (OUT) | The system administrator will need to create an account for each instructor using InPress. They will need to provide InPress with basic information about the instructor, and will have the opportunity to create a unique username and password. |
| R2 | Login as an Instructor | Instructor’s Credential (IN)  Instructor’s Personalized Webpage (OUT) | Using the username and password provided by the System Administrator, instructors would be able to login and see their personalized page. |
| R3 | Login as a Student | Student’s Credential (IN)  Student’s Personalized Webpage (OUT) | Students will be able to login to InPress via their student number. Upon login, they will see their personalized webpage with courses they are enrolled in. |
| R4 | Create a Course | Instructor’s Credentials (IN)  Course Information (IN)  Created Course (OUT) | Courses can only be created through an instructor’s account. Once logged in, instructors may create a course by selecting the appropriate GUI button. Instructors will need to provide the web interface with basic course information (i.e. course name, course code, class list, etc.). Once created, the course will appear on the instructor’s personalized webpage. |
| R5 | Delete a Course | Instructor’s Credentials (IN)  Course Information (IN) Course Deleted (OUT) | Upon selecting the appropriate GUI button on the instructor’s personalized webpage, instructors are able to manage their courses. Through the “Remove a Course” UI, instructors will be able to delete a course. |
| R6 | Add an Assessment | Instructor’s Credentials (IN)  Course Information (IN) Assessment Information (IN)  Course Populated with assessment data (OUT) | Upon selecting the appropriate GUI button on the instructor’s personalized webpage, instructors are able to manage their courses. By selecting “Add an Assessment”, instructors are able to add an assessment by filling out the necessary information (Name of Assessment, Effective Date of assessment). Once this form is submitted, the assessment is added to the course. |
| R7 | Remove an Assessment | Instructor’s Credentials (IN)  Course Information (IN) Assessment Information (IN) | Upon selecting the appropriate GUI button on the instructor’s personalized webpage, instructors are able to manage their courses. By selecting “Remove an Assessment”, instructors are able to remove an assessment. |
| R8 | Add questions to an assessment | Instructor’s Credentials (IN)  Course Information (IN) List of Questions/Solutions (IN)  Course Populated with Questions/Solutions (OUT) | Upon selecting the appropriate GUI button on the instructor’s personalized webpage, instructors are able to manage their courses. Instructors will be able to populate a specific assessment with various questions. Once populated, instructors will be led back to their personalized webpage with the course now populated. |
| R9 | Post/Unpost an Assessment | Instructor’s Credentials (IN)  Course Information (IN) Posted Course (OUT) | Upon selecting the appropriate GUI button on the instructor’s personalized webpage, instructors will be able to manage their courses. Instructors will be able to post/Unpost an assessment for a course. |
| R10 | Effective Date (Assessment) | Instructor’s Credentials (IN)  Course Information (IN) Posted Course (OUT) | Upon selecting the appropriate GUI button on the instructor’s personalized webpage, instructors will be able to manage their assessments. All assessments have an effective date which restrict the assessment to be available only on that date. Instructors are able to change this date after an assessment has been created. |
| R11 | Data Analysis | Instructor’s Credential (IN)  Course Information (IN)  Data Results (OUT) | Upon selecting the appropriate GUI button on the instructor’s personalized webpage, instructors can navigate to the “Data Analysis” webpage. Each assessment of a course will have its own unique data analysis page. Instructors will be able to see various statistics about the class such as how many students have completed the assessment, and what the answer distribution was like for each question. |
| R12 | Data Analysis for Students (Answer Key) | Student’s Credential (IN)  Course Information (IN)  Data Results (OUT) | Upon selecting the appropriate GUI button on the Student’s UI, students are able to navigate to an assessment, and view their answer along with the correct answer to each question. |
| R13 | Add and Remove Students | Instructor’s Credentials (IN)  Course Information (IN) | Upon selecting the appropriate GUI button on the Instructor’s UI, Instructors will be able to add and remove students by adding/removing their student numbers. |
| R14 | LaTex in Questions/Solutions | Instructor’s Credentials (IN)  Course Information (IN)  LaTex Questions (OUT) | Upon selecting the appropriate GUI button on the Instructor’s UI, navigate to an assessment and add a question. Follow the instructions on how to add latex to your questions and/or solutions. Questions and Answers are outputted in LaTex. |
| R15 | Answer Questions | Student’s Credential (IN)  Course Information (IN) | Upon selecting the appropriate GUI button on the Student’s UI, students are able to navigate to an assessment, and answer the questions posted by their instructor. |

Figure 2: Work Partitioning

## THE SCOPE OF THE PRODUCT

### Product Boundary

In this case the work scope is the product boundary.

### Product Use Case List

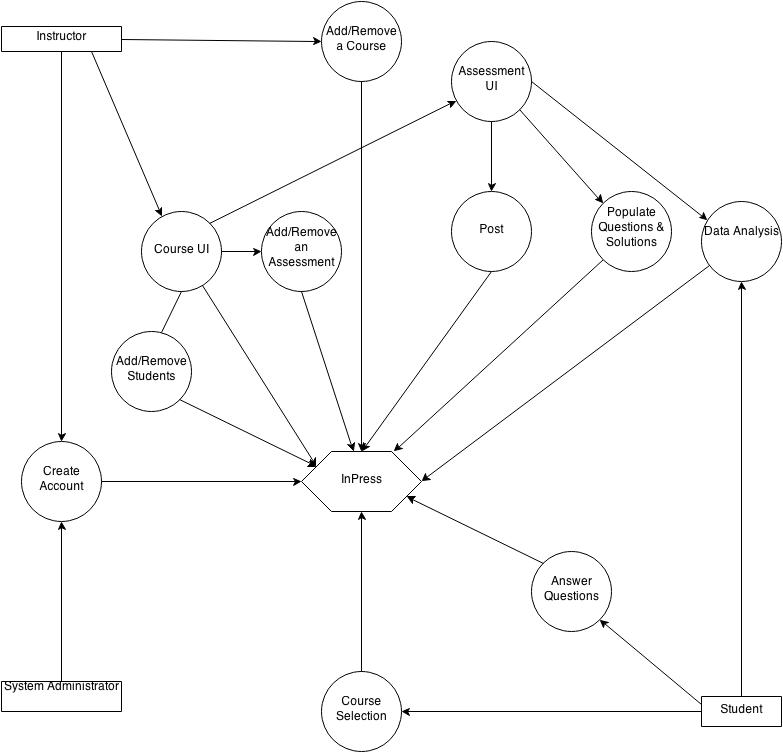


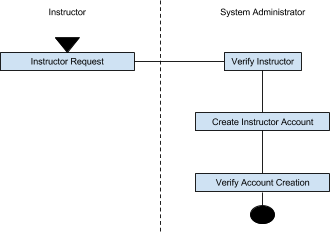
Figure 3: Product Use Case List

### Individual Product Use Cases

1. **Product Use Case Name:** Create Instructor’s Account

**Trigger:** System Administrator Manage Account Page

**Preconditions:** Login as System Administrator

**Interested Stakeholders:** The client, customer

**Actor:** Web browser

**Activity Diagram:**

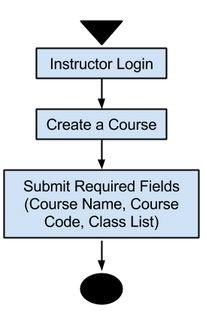
Figure 4: Create Instructor's Account Use Case

1. **Product Use Case Name:** Create a Course

**Trigger:** “Create a Course” on Manage Courses webpage

**Preconditions:** Have to login as an instructor

**Interested Stakeholders:** The client, customer

**Actor:** Web browser

**Activity Diagram:**

Figure 5: Create a Course Use Case

1. **Product Use Case Name:** Create Questions

**Trigger: “**Create Questions” UI Button in the Course’s Personalized Webpage

**Preconditions:** Have to login as an instructor

**Interested Stakeholders:** The client, customer

**Actor:** Web browser

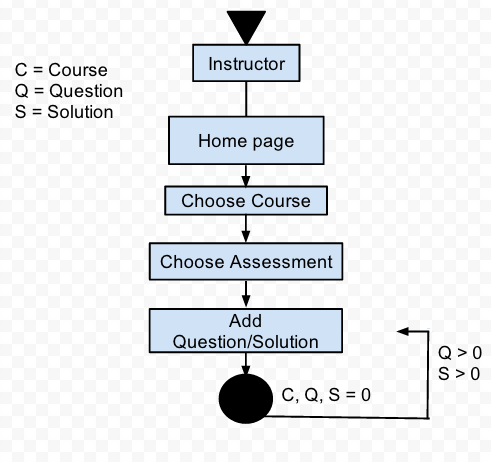
**Activity Diagram:**

Figure 6: Create Questions Use Case

1. **Product Use Case Name:** Post/Unpost an Assessment

**Trigger:** “Post/Unpost” UI Button in the Course’s Personalized Webpage

**Preconditions:** Have to login as an instructor

**Interested Stakeholders:** The client, customer

**Actor:** Web browser

**Activity Diagram:**

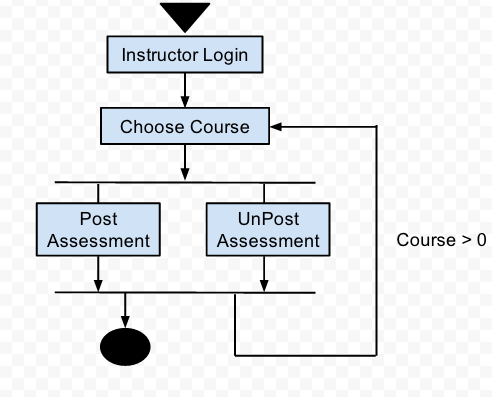


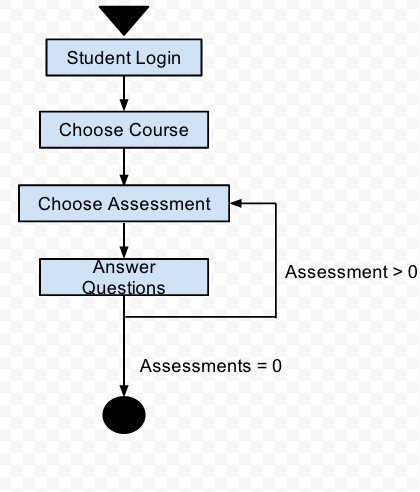
Figure 7: Post and Unpost an Assessment

1. **Product Use Case Name:** Answer the question

**Trigger:** “Start Assessment” UI button in the Course’s personalized webpage

**Preconditions:** Have to login as a Student

**Interested Stakeholders:** The client, customer

**Actor:** Web browser

**Activity Diagram:**

Figure 8: Answer the Question Use Case

1. **Product Use Case Name:** Check the results

**Trigger:** “Results” UI Button on the Course’s Personalized webpage

**Preconditions:** Login as an instructor/student, must be enrolled in the course

**Interested Stakeholders:** The client, customer

**Actor:** Web browser

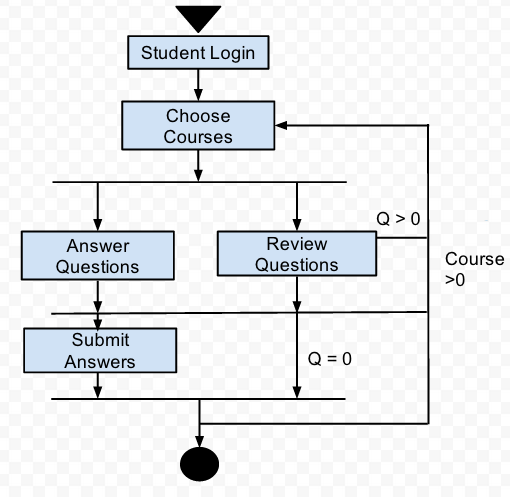
**Activity Diagram:**

Figure 9: Check the Results Use Case

1. **Product Use Case Name:** Question Data Analysis

**Trigger:** “Data Analysis” UI Button in the Course’s personalized webpage

**Preconditions:** Have to login as Instructor and posted this question

**Interested Stakeholders:** The client, customer

**Actor:** Web browser

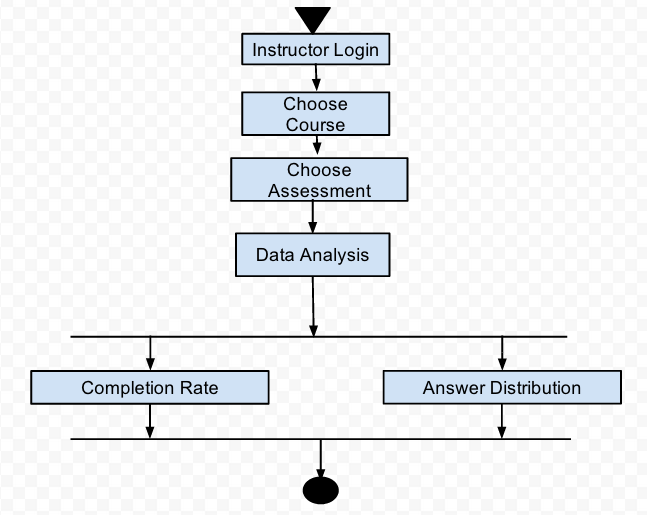
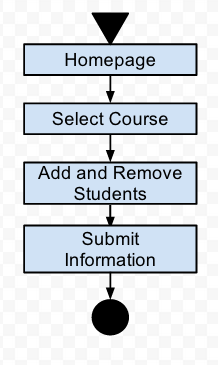
**Activity Diagram:**

Figure 10: Question Data Analysis Use Case

1. **Product Use Case Name:** Add and Remove Students

**Trigger:** Course Homepage

**Preconditions:**

**Interested Stakeholders:** The client, customer

**Actor:** Web browser

**Activity Diagram:**

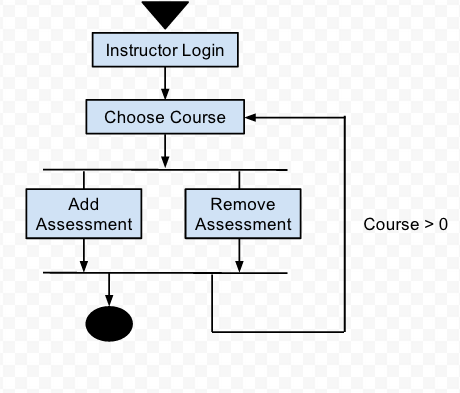
Figure 11: Student Account Registration Use Case

1. **Product Use Case Name:** Add and Remove Assessments

**Trigger:** Course Homepage

**Preconditions:**

**Interested Stakeholders:** The client, customer

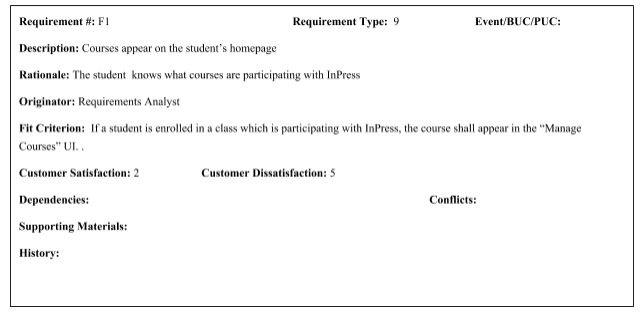
**Actor:** Web browser

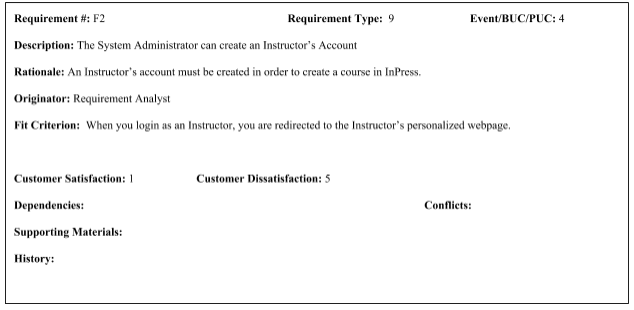
**Activity Diagram:**

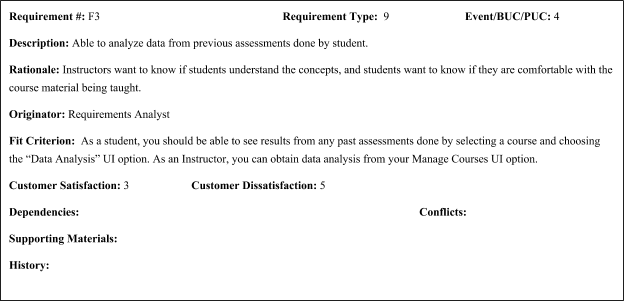
Figure 12: Add and Removing Assessments

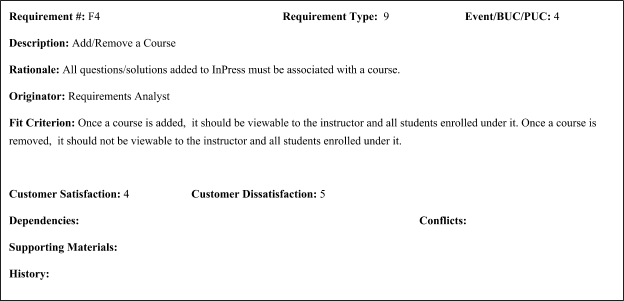
## FUNCTIONAL AND DATA REQUIREMENTS

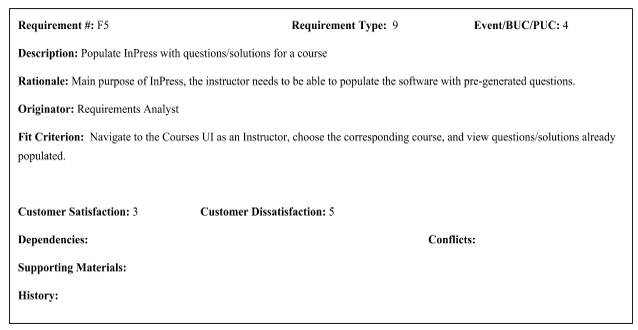
### Functional and Data Requirements











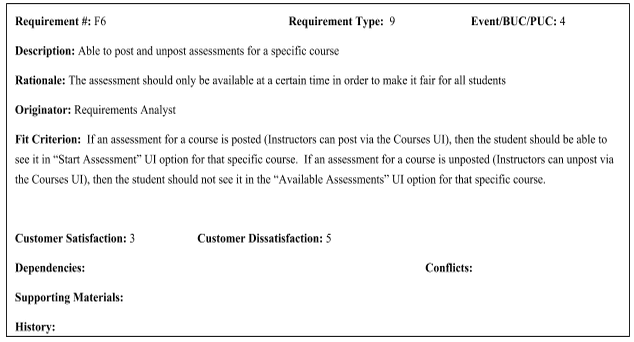


Figure 13: Volere Cards

# Nonfunctional Requirements

## LOOK AND FEEL REQUIREMENTS

### Appearance Requirements

The User Interface for the web-application will be a simple interface with very few buttons.

### Style Requirements

The style will be user friendly, keeping button clicks to a minimal in order to achieve various tasks.

## USABILITY AND HUMANITY REQUIREMENTS

### Ease of Use Requirements

The web-application should be easy to use for every type of users out there, from novice to power-users.

### Personalization and Internationalization Requirements

- Since each student and instructor have different courses, they will have their own webpages

- The web-application will be in English.

### Learning Requirements

- The learning curve will be kept minimal. It should not take them more than five minutes to understand how to use this web-application.

- For the System Administrator or Technicians, it should not be difficult to install the application, maintain and create Instructor ID’s.

### Understandability and Politeness Requirements

- Any type of user can use the web application easily

- Hide functions that will confuse users

- There should be no grammar mistakes and inappropriate language anywhere on the web-application.

### Accessibility Requirements

- For Mobile Apps, it will be on Apple’s App Store, or Google’s Play store

- For Laptops, there will be a link that can access it directly (something similar to Avenue).

- Supported on all Internet Enabled devices (including Apple, Blackberry, Sony, Android devices)

## PERFORMANCE REQUIREMENTS

### Speed and Latency Requirements

- Each webpage shall take no longer than ten seconds to load.

### Safety- Critical Requirements

- Student data should not be accessible by any other student

- Instructors data should only be accessed by the instructor themself or the system administrator.

- Only the System Administrator has access to the administration site.

### Precision or Accuracy Requirements

- There should be no errors when saving the students answers, even after they changed their answers.

- The web application should not be retrieving incorrect student information.

- Students, administrator, and instructors should have their own unique ID’s and should not be mixed up

### Reliability and Availability Requirements

- Once launched to the university, InPress will be available to students/instructors 24 hours a day 7 days a week.

### Robustness or Fault- Tolerance Requirements

- The web-application will be robust enough to recover from errors (such as duplicated login)

- Zero fault-tolerance. This is because if the students’ data are saved incorrectly, data analysis will not be accurate. Hence, defeating the point of the data analysis.

### Capacity Requirements

- The application should be scalable, meaning the max capacity will depend on the hardware (system and storages)

- The application will support at least 200 concurrent user connections with at most 10 seconds latency

### Scalability or Extensibility Requirements

- Infrastructure similar to cloud. An increase of storage space, or an upgrade for the system (CPU / Memory) will be sufficient.

### Longevity Requirements

- It should last until it is no longer used. There will be maintenance patches, and the robustness should recover from failures.

## OPERATIONAL AND ENVIRONMENTAL REQUIREMENTS

### Expected Physical Environment

- It could be anywhere, from classroom to home.

- The hosting server will be in the school’s data center

### Requirements for Interfacing with adjacent Systems

Not Applicable

### Productization Requirements

Not Applicable

### Release Requirements

This product will be opened sourced and will have regular maintenance.

## MAINTAINABILITY AND SUPPORT REQUIREMENTS

### Maintenance Requirements

- Periodic patches and maintenance to ensure the system is up-to-date and bug-free

### Supportability Requirements

- Backup should be made every month for current courses.

### Adaptability Requirements

- The web-application should be able to adapt into any Linux environment

- Migrating into a new box should be very easy

## SECURITY REQUIREMENTS

### Access Requirements

- Only the System Administrator can make changes to the web-application.

- Instructors can only see their own course data along with their enrolled students’ data

- Students should only access their only account

### Integrity Requirements

- Security algorithms will be in place to prevent data corruption

- Minimum of RAID 5 array will be in place to recover from hard disk failures.

### Privacy Requirements

- Only the System Administrator may see everyone’s data

- The instructors will only see their own course data along with their enrolled students' data.

- Students can only see their own data.

- The main implementation will be hidden from everyone.

### Audit Requirements

- The web-application will follow the institute’s data center security compliances.

### Immunity Requirements

Not Applicable as it will depend on the Institute’s Security.

## CULTURAL AND POLITICAL REQUIREMENTS

### Cultural Requirements

Not applicable

### Political Requirements

Not applicable

## LEGAL REQUIREMENTS

### Compliance Requirements

- Web-application is open-sourced, and will follow the relevant licenses.

- Web-application should not involve any patent wars.

### Standards Requirements

**-** Everything is confidential, and should not be released to anyone except authorized individuals

- Will not be affiliate with any Patents.

# Project Issues

## OPEN ISSUES

* We do not know where the server is physically located
* We do not know the specifications of the server

## OFF-THE-SHELF SOLUTIONS

N/A

## NEW PROBLEMS

N/A

## TASKS

* 1. Gather missing requirements
  2. Develop prototype
  3. Get feedback about prototype
  4. Provide arguments why current prototype is ideal by comparing it to current of-the-shelf products.

## MIGRATION TO THE NEW PRODUCT

Not applicable.

## RISKS

There is a risk of being unable to migrate the web-application to a new server if the current server cannot support any more resources (i.e. increase of CPU or memory). Although everything can be made into an image and be migrated to a new server, there is no promise that storage will not have errors.

## COSTS

Not in the requirement

## USER DOCUMENTATION AND TRAINING

There will be documentation for all three groups of people: the administrators (including system administrators / technicians), instructors and students. This documentation is crucial for the administrators because they will be the ones that have to set up the system.

## WAITING ROOM

Not applicable

## IDEAS FOR SOLUTIONS

A number of solutions currently exist. 1) iClicker ® 2) Top Hat Monocle

# Requirements Timeline

|  |  |
| --- | --- |
| **Requirements** | **Projected Month Deadline** |
| R1: Create an Instructor Account | By December |
| R2: Login as an Instructor | By December |
| R3: Login as a Student | By December |
| R4: Create a course | By December |
| R5: Delete a Course | By December |
| R6: Add an Assessment | By January |
| R7: Remove an Assessment | By January |
| R8: Add questions to an assessment | By January |
| R9: Post/Unpost an Assessment | By January |
| R10: Effective Date (Assessment) | By January |
| R11: Data Analysis | By February |
| R12: Data Analysis for Students (Answer Key) | By February |
| R13: Add and Remove Students | By March |
| R14: LaTex in Questions/Solutions | By March |
| R15: Answer Questions | By February |