**Software Requirements Specification**

Version 1.0

March 3, 2015

**Student Question Bank System**

Gustavo Tavares

Jack Kitchener

Connie Qi

Contents

[Table of Contents2 2](#_Toc421801881)

[List of Figures 8](#_Toc421801882)

[1.0. Introduction 9](#_Toc421801883)

[1.1. Purpose 9](#_Toc421801884)

[1.2. Scope of Project 9](#_Toc421801885)

[1.3. Glossary 9](#_Toc421801886)

[1.4. References 10](#_Toc421801887)

[1.5. Overview of Document 10](#_Toc421801888)

[2.0. Overall Description 11](#_Toc421801889)

[2.1 System Environment 11](#_Toc421801890)

[2.2 Functional Requirements Specification 11](#_Toc421801891)

[**2.2.1 Use Cases** 12](#_Toc421801892)

[**2.2.1.1 Use Case 01 - Login** 12](#_Toc421801893)

[**2.2.1.2 Use Case 02 - Initial Infomation** 12](#_Toc421801894)

[2.3 User Characteristics 25](#_Toc421801895)

[2.4 Non-Functional Requirements 25](#_Toc421801896)

[3.1 External Interface Requirements 26](#_Toc421801897)

[3.2 Detailed Non-Functional Requirements 26](#_Toc421801898)

[3.2.1 Logical Structure of the Data 26](#_Toc421801899)

[3.3.2 Security 28](#_Toc421801900)

# List of Figures

[Figure 1 - System Environment](#h.35nkun2)

[Figure 2 - Article Submission Process](#h.z337ya)

[Figure 3 - Editor Use Cases](#h.3whwml4)

[Figure 4 - Logical Structure of the Article Manager Data](#h.4bvk7pj)

# 1.0. Introduction

## 1.1. Purpose

The purpose of this document is to present a detailed description of the Student Question Bank System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system.

## 1.2. Scope of Project

This software system will be a web testing system for the students in Cert IV & Diploma Web/programming of Central TAFE. This system will be designed to put questions online for students to answer about their course that they have enrolled. This system also designed for lectures to access as administrator to put in questions about particular units/clusters.

## 1.3. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Author | Person submitting an article to be reviewed. |
| Database | Collection of all the information monitored by this system. |
| Member | A member of the team. |
| Software Requirements Specification | A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document. |
| Stakeholder | Any person with an interest in the project who is not a developer. |
| User | All developer. |
| Students | Person who study Certificated IV or Diploma in Central Tafe. |

## 1.4. References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

## 1.5. Overview of Document

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

# 2.0. Overall Description

## 2.1 System Environment

**Figure 1 - System Environment**

The Web Quiz System has two active actors and two cooperating systems. Both lectures and students can access the online Website on the Home Page to find the basic information and FAQs. Lectures can log in the Website as administrators to arrange quiz for students and check the students’ results. There is a link in the Website for students to log in the Quiz App to finish the quiz assessment. Administrator initialises the data for lectures and course.

## 2.2 Functional Requirements Specification

This section outlines the use cases for each of the active actors separately.

### **2.2.1 Use Cases**

### **2.2.1.1 Use Case 01 - Login**

|  |  |
| --- | --- |
| **Use Case** | UC01 - Login |
| **Goal** | This use case has the objective authenticate the entry of the actor in the system. |
| **Preconditions** | The actor must be registered in the database system. |
| **Success End Condition** | After executing this use case, the actor can take advantage of their permissions. |
| **Actors** | Student, Lecture, Administrator |
| **Description / Main Success Scenario** | 1.The system provides a screen for the author inform the login data.  2.The author informs the login data.  3.The author click on the button “Login”.  4.The system validates the data.  5.The system processes the information.  6.The system displays the main screen.  End of the use case |

### **2.2.1.2 Use Case 02 - Initial Infomation**

|  |  |
| --- | --- |
| Use Case | UC02 -Initial Information |
| Goal | This use case has the objective initial database. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can import students,lectures and units information. |
| Actors | Administrator |
| Description / Main Success Scenario | 1.The author click on the button “Import”.  2.The system shows a file selecting dialog.  3.The author choose one “.txt” file to upload.  4. The author click “Finish”.  5. The system process the information. End of the use case |

#### 2.2.2 Student Use Case:

**2.2.2.1 Use Case 03 - List Students**

|  |  |
| --- | --- |
| Use Case | UC03- List Students Information |
| Goal | This use case has the objective listing all students information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add, delete or update student information. |
| Actors | Lecture and Administrator |
| Description / Main Success Scenario | 1.The author click the menu on left “Student”.  2.The system shows all students enrolled in. End of the use case |

**2.2.2.2 Use Case 04 - View Student**

|  |  |
| --- | --- |
| Use Case | UC04- View a Student Information |
| Goal | This use case has the objective showing one student information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can back to the main page. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The system shows the student information who was clicked.  2.The author can click the button “Back” back to main page. |

**2.2.2.3 Use Case 05 - Edit Student**

|  |  |
| --- | --- |
| Use Case | UC05 - Edit Student Information |
| Goal | This use case has the objective updating student information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add unit information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Edit Student”.  2.The system shows the student information and the units enrolled in.  3. The author add student name, and email.  4. The author can also click the button “Add Unit” to add a unit for the student.  5. They system pop up “Unit” window.  6. The author select on unit to add.  7.The author click “Save Student”.  8. The system process the information.  End of the use case |

**2.2.2.4 Use Case 06 - Delete Student**

|  |  |
| --- | --- |
| Use Case | UC06- Delete Student Information |
| Goal | This use case has the objective deleting a student information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add, view or update student information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Delete Student”.  2.The system shows a pop up window to confirm.  3. The author click “Sure”.  4. The system delete the student information. End of the use case |

**2.2.2.5 Use Case 07 - Add Student**

|  |  |
| --- | --- |
| Use Case | UC07- Add Student Information |
| Goal | This use case has the objective adding a new student information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add unit information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Add Student”.  2.The system shows New Student screen.  3. The author add student name, and email.  4. The author can also click the button “Add Unit” to add a unit for the student.  5. They system pop up “Unit” window.  6. The author select on unit to add.  7.The author click “Save Student”.  8. The system process the information.  End of the use case |

#### 2.2.3 Lecture Use Case:

**2.2.3.1 Use Case 08 - List Lecture**

|  |  |
| --- | --- |
| Use Case | UC08- List Lecture Information |
| Goal | This use case has the objective listing all Lecture information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add, delete or update Lecture information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click the menu on left “Lecture”.  2.The system shows all Lecture enrolled in. End of the use case |

**2.2.3.2 Use Case 09 - View Lecture**

|  |  |
| --- | --- |
| Use Case | UC09- View a Lecture Information |
| Goal | This use case has the objective showing one Lecture information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can back to the main page. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The system shows the Lecture information who was clicked.  2.The author can click the button “Back” back to main page. |

**2.2.3.3 Use Case 10- Edit Lecture**

|  |  |
| --- | --- |
| Use Case | UC10- Edit Lecture Information |
| Goal | This use case has the objective updating Lecture information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add unit information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Edit Lecture”.  2.The system shows the Lecture information and the relative units.  3. The author add Lecture name, and email.  4. The author can also click the button “Add Unit” to add a unit for the Lecture.  5. They system pop up “Unit” window.  6. The author select on unit to add.  7.The author click “Save”.  8. The system process the information.  End of the use case |

**2.2.3.4 Use Case 11 - Delete Lecture**

|  |  |
| --- | --- |
| Use Case | UC11- Delete Lecture Information |
| Goal | This use case has the objective deleting a Lecture information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add, view or update Lecture information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Delete Lecture”.  2.The system shows a pop up window to confirm.  3. The author click “Sure”.  4. The system delete the Lecture information. End of the use case |

**2.2.3.5 Use Case 12 - Add Lecture**

|  |  |
| --- | --- |
| Use Case | UC12- Add LectureInformation |
| Goal | This use case has the objective adding a new Lecture information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add unit information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Add Lecture”.  2.The system shows New Lecture screen.  3. The author add Lecturename, and email.  4. The author can also click the button “Add Unit” to add a unit for the student.  5. They system pop up “Unit” window.  6. The author select on unit to add.  7.The author click “Save Lecture”.  8. The system process the information.  End of the use case |

#### 2.2.4 Unit Use Case:

**2.2.4.1 Use Case 13 - List Units**

|  |  |
| --- | --- |
| Use Case | UC13- List Units Information |
| Goal | This use case has the objective listing all Units information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add, delete or update Unit information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click the menu on left “ Unit ”.  2.The system shows all Units. End of the use case |

**2.2.4.2 Use Case 14 - View**  **Unit**

|  |  |
| --- | --- |
| Use Case | UC14- View a Unit Information |
| Goal | This use case has the objective showing one Unit information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can back to the main page. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The system shows the Unit information which was clicked.  2.The author can click the button “Back” back to main page. |

**2.2.4.3 Use Case 15 - Edit Unit**

|  |  |
| --- | --- |
| Use Case | UC15- Edit Unit Information |
| Goal | This use case has the objective updating Unit information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can go back to main page. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Edit Unit ”.  2.The system shows the Unit information.  3. The author add Unit title and Course name.  4.The author click “Save Unit ”.  8. The system process the information.  End of the use case |

**2.2.4.4 Use Case 16 - Delete Unit**

|  |  |
| --- | --- |
| Use Case | UC16- Delete Unit Information |
| Goal | This use case has the objective deleting a Unit information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add, view or update Unit information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Delete Unit ”.  2.The system shows a pop up window to confirm.  3. The author click “Sure”.  4. The system delete the Unit information. End of the use case |

**2.2.4.5 Use Case 17 - Add Unit**

|  |  |
| --- | --- |
| Use Case | UC17- Add UnitInformation |
| Goal | This use case has the objective adding a new Unit information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can go back to the main page.. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Add Unit”.  2.The system shows New Unit screen.  3. The author add the unit title and course name  4.The author click “Save”.  5. The system process the information.  End of the use case |

#### 2.2.5 Quiz Use Case:

**2.2.5.1 Use Case 18 - List Quizzes**

|  |  |
| --- | --- |
| Use Case | UC18- List Quizzes Information |
| Goal | This use case has the objective listing all Quizzes information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add, delete or update Quizze information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click the menu on left “Quizzes”.  2.The system shows all Quizzes. End of the use case |

**2.2.5.2 Use Case 19 - View Quizzes**

|  |  |
| --- | --- |
| Use Case | UC19- View Quizzes Information |
| Goal | This use case has the objective showing one Quizze information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can back to the main page. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The system shows the Quizzse information which was clicked.  2.The author can click the button “Back” back to main page. |

**2.2.5.3 Use Case 20 - Edit Quiz**

|  |  |
| --- | --- |
| Use Case | UC20- Edit Quizze Information |
| Goal | This use case has the objective updating Quiz information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add Question information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Edit Quiz”.  2.The system shows the Quiz Edit screen  3. The author select question type from a drop down list and then click “Add Question” to add a new question from selecting questions in the questions list.  4. The author can also click the button “Delete” which in the question row list to delete a question..  5. The system pop up “Unit” window.  6.The author click “Save”.  7. The author click “Back” to go back to the main page.  8. The system process the information.  End of the use case |

**2.2.5.4 Use Case 21 - Add Quiz**

|  |  |
| --- | --- |
| Use Case | UC21- Add Quiz |
| Goal | This use case has the objective adding a new Quiz information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can go back to the main page.. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Add Quiz”.2.The system shows New Quiz screen.3. The author add the Quiz title and Unit name4.The author click “Add Question” button.5.The author system shows Select Questions screen.6. The author can select all the questions which he wants to be shown in the Quiz. 7. The author click “Save”.  8. The system process the information. End of the use case |

**2.2.5.5 Use Case 22 - Delete Quiz**

|  |  |
| --- | --- |
| Use Case | UC22- Delete Quiz Information |
| Goal | This use case has the objective deleting a Quiz information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add, view or update Quiz information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Delete Quiz ”.  2.The system shows a pop up window to confirm.  3. The author click “Sure”.  4. The system delete the Quiz information. End of the use case |

#### 2.2.6 Question Use Case:

**2.2.6.1 Use Case 23 - List Questions**

|  |  |
| --- | --- |
| Use Case | UC23- List Questions Information |
| Goal | This use case has the objective listing all Questions information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add, delete or update Question information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click the menu on left “Question”.  2.The system shows all Questions. End of the use case |

**2.2.6.2 Use Case 24 - View Questions**

|  |  |
| --- | --- |
| Use Case | UC24- View a Question Information |
| Goal | This use case has the objective showing one Question information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can back to the main page. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The system shows the Question information which was clicked.  2.The author can click the button “Back” back to main page. |

**2.2.6.3 Use Case 25 - Edit Question**

|  |  |
| --- | --- |
| Use Case | UC25- Edit Question Information |
| Goal | This use case has the objective updating Question information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add Question information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Edit Question”.  2.The system shows the Question Edit screen  3. The author enters the information about the question.  4.The author click “Save”.  5. The author click “Back” to go back to the main page.  6. The system process the information.  End of the use case |

**2.2.6.4 Use Case 26- Add Question**

|  |  |
| --- | --- |
| Use Case | UC26- Add Question |
| Goal | This use case has the objective adding a new Question information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can go back to the main page.. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Add Question”.2.Th author add the question information like title, unit, type and options 3. The author click “Save”.  4. The system process the information. End of the use case |

**2.2.6.5 Use Case 27 - Delete Question**

|  |  |
| --- | --- |
| Use Case | UC27- Delete Question Information |
| Goal | This use case has the objective deleting a Question information. |
| Preconditions | The actor must have logged in the system. |
| Success End Condition | After executing this use case, the actor can add, view or update Question information. |
| Actors | Lecture |
| Description / Main Success Scenario | 1.The author click on the button “Delete Question ”.  2.The system shows a pop up window to confirm.  3. The author click “Sure”.  4. The system delete the Question information.  5. The system process the information.     End of the use case |

## 2.3 User Characteristics

The Users are expected to be the team developer and to be able to use email with attachments.

The detailed look of these pages is discussed in section 3.2 below.

## 2.4 Non-Functional Requirements

The Online Quiz Bank System will be on a server with high speed Internet capability. The physical machine to be used will be determined by the Central Tafe. The software developed here assumes the use of a tool such as Ruby on Rails and C# for connection between the Web pages and the database. The speed of the students connection will depend on the hardware used rather than characteristics of this system.

**3.0. Requirements Specification**

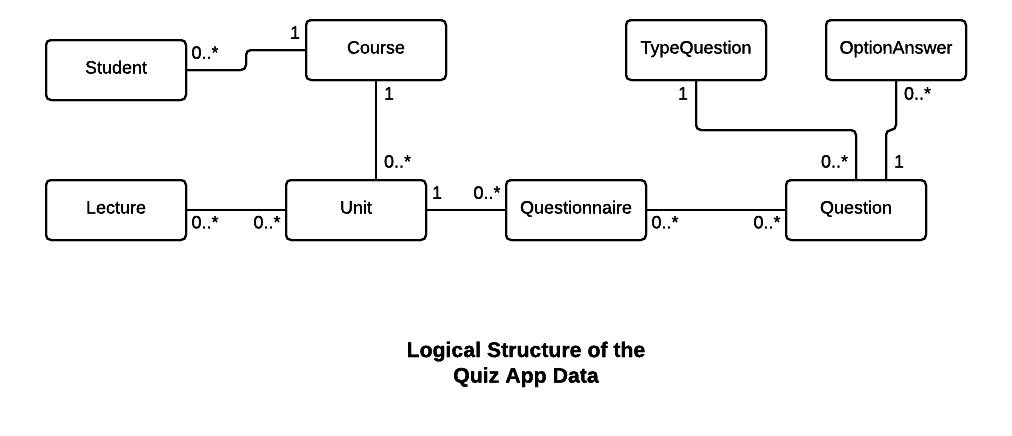
## 3.1 External Interface Requirements

The only link to an external system is the link to the Central Tafe server to share a database with the quiz website which is written by C#.

## 3.2 Detailed Non-Functional Requirements

### 3.2.1 Logical Structure of the Data

The logical structure of the data to be stored in the Central Tafe server database is given below.



The data descriptions of each of these data entities is as follows:

**Student Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| StudentNumber | Integer | Student number |  |
| FirstName | string | First name of student |  |
| LastName | string | Last name of student |  |
| Email | string | Email of student |  |
| Password | string | Password for login |  |
| CourseID | integer | The course ID which student attend |  |

**Course Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Name | string | Name of course |  |
| ID | Integer | ID of course | Primary key |

**Unit Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Name | string | Name of unit |  |
| CourseID | Integer | ID of course |  |

**Question Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Title | string | Name of question |  |
| CorrectAnswer | string | The correct answer |  |
| TypeQuestionID | Integer | The ID of question type |  |

**TypeQuestion Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| ID | Integer | Id of type of question |  |
| Type | string | Type of question |  |

**OptionAnswer Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Option | string | The option answer |  |
| QuestionID | integer | ID of Question |  |

**Questionnair Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| QuestionID | Integer | Id of type of question |  |
| QuestionnaireID | Integer | ID of Questionnaire |  |

**Lecture Data Entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| LecturerNumber | Integer | Number of lecture |  |
| FirstName | string | First name of lecture |  |
| LastName | string | Last name of lecture |  |
| Email | string | Email of lecture |  |
| Password | string | Password for login |  |
| IsAdm | boolean | If the lecture is admin |  |

### 3.3.2 Security

The server on which the Online Quiz Bank System resides will have its own security to prevent unauthorized *write*/*delete* access. There is no restriction on *read* access. The Quiz Test system is on the client systems and thus is external to the system.

The PC on which the Quiz Test system resides will have its own security. Only the Students will have physical access to the machine and use their own user name and password to log in, then answer it. There is no special protection built into this system.