|  |
| --- |
| Central Queensland University, Sydney |
| Requirement Specification  Rotary Youth Drive Awareness Project |
| COIT20273: Software Design and Development Project |
|  |
| Lecturer/Tutor: Zakiullah Khan Course Coordinator: Dr. Lily Li |
| **Due date: 8/11/2017** |

Project Members:

Bijay Shahi (12021856)

Keshav Khadka (S0280624)

Pratik Shrestha (S0282733)

|  |  |  |  |
| --- | --- | --- | --- |
| **S.N.** | **Document Name** | **Version** | **Revision Date** |
| 1. | Requirement Specification | 1.0 | 2017-08-11 |

Table of Contents

[1. Functional and non-functional requirements 1](#_Toc490229267)

[1.1. Functional requirements 1](#_Toc490229268)

[1.2. Non-functional requirements 1](#_Toc490229269)

[1.3. Requirement Priority 2](#_Toc490229270)

[2. Use cases (UML) 3](#_Toc490229271)

[3. Use cases (text description) 4](#_Toc490229272)

[4. Mapping of requirements to use cases 5](#_Toc490229273)

# 1. Functional and non-functional requirements

## 1.1. Functional requirements

Administrators should be able to:

* Create/read/update/delete quiz questions
* Give an explanation on each question as hints
* View the report of quizzes undertaken by students
* View quiz answers submitted by individual student
* Filter quiz results by date, student name and email

Students should be able to:

* Attempt quiz questions with multiple choices
* View the results of their attempts
* View the right answers for the questions
* View the explanation (hints) of each question provided by administrator
* Review their performances

## 1.2. Non-functional requirements

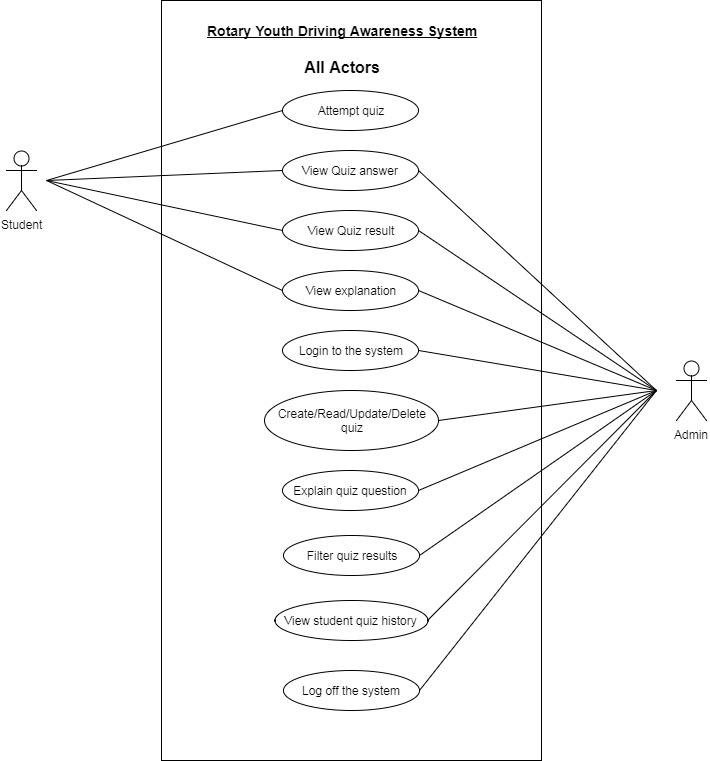
* Strong security, authentication and authorisation mechanism must be implemented.
* Database design must be in third normal form (3NF).
* Coding must be documented, well indented and flexible for enhancement.
* Interactive and responsive user interfaces must be implemented.
* System must perform better in all aspects in terms of speed, reliability and availability.

## 1.3. Requirement Priority

|  |  |
| --- | --- |
| **Requirement** | **Priority** |
| Create/read/update/delete quiz questions | Basic |
| Give an explanation on each question | Basic |
| View the report of quizzes undertaken by students | Basic |
| View quiz answers submitted by individual student | Basic |
| Filter quiz results by date, student name and email | Advanced |
| Attempt quiz questions with multiple choices | Basic |
| View the results of their attempts | Basic |
| View the right answers for the questions | Basic |
| View the explanation of each question provided by administrator | Basic |
| Review their performances | Basic |
| Strong security, authentication and authorisation mechanism must be implemented. | Advanced |
| Database design must be in third normal form (3NF). | Advanced |
| Coding must be documented, well indented and flexible for enhancement. | Basic |
| Interactive and responsive user interfaces must be implemented. | Basic |
| System must perform better in all aspects in terms of speed, reliability and availability. | Basic |

**Table 1. Requirement Priority**

# 2. Use cases (UML)



**Fig. 1. UML Use case diagram**

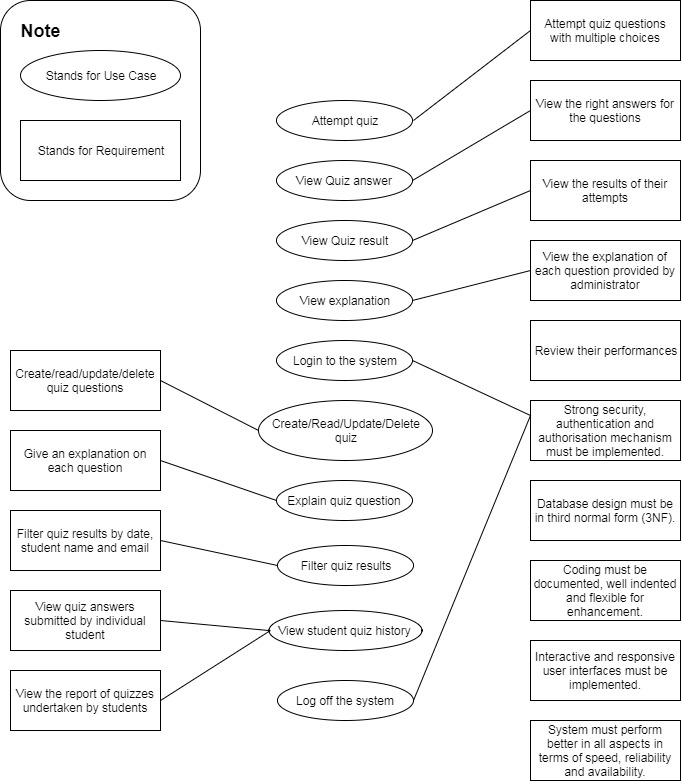
# 3. Use cases (text description)

|  |  |  |
| --- | --- | --- |
| **Use Case** | **Actor** | **Description** |
| Attempt Quiz | Student, Admin | Users can attempt the quiz by entering in the system. |
| View Quiz answers | Student, Admin | After attempting the quiz, users can view the correct answer for each question. |
| View Quiz result | Student, Admin | Users can view their quiz result after attempting the quiz. |
| View explanation | Student, Admin | After attempting the quiz, users can view the correct answer and explanation. |
| Login to the system | Admin | Admin can log into the system with his/her credential. |
| Create/Update/  Read/Delete quiz | Admin | Admin can set up the quiz questions by clicking on “Create question” button and update, read and delete the questions by clicking on the “Update”, “View” and “Delete” link on the action column respectively. |
| Explain quiz question | Admin | Admin can give an explanation to the quiz question while they create a question. |
| Filter quiz results | Admin | Admin can filter quiz results according to date, student name and email address by clicking on the “Filter” button. |
| View student quiz history | Admin | Admin can view quiz history of all students by clicking on the student’s name from the list/grid display. |
| Log off the system | Admin | Admin can log off from the system by clicking on the “Log off” link. |

**Table 2. Use case description**

# 4. Mapping of requirements to use cases

All functional requirements are mapped to the use cases. One of the non-functional requirement has been mapped which is related with the Security and Access Control List (ACL).



**Fig. 2. Use case requirement mapping**