Functional Requirements duggaedservice

Functionality Requirement No	Function Requirement Description
FR1	Manage quizzes
FR2	Manage quiz variants
FR3	User authentication and authorization

FN: No:	Test Proposal:
FR1	Create a new quiz with valid input such as a course with ID, quiz, start date, deadline etc. and verify that the quiz is updated in the database with the correct info.
FR2	Update an existing quiz variant and check if the quiz is updated in the database with correct info.
FR3	Try to attempt to access a quiz without logging in and verify that it's not accessible for the user.

FN No:	FR1
Shall	Users with write-access or is a super user **shall** be able to manage quizzes.
While	**While** the users have a valid session and correct access level.
When	**When** a request is sent with a specific option.
Where	**Where** the users, with permission, can manage quizzes and the "opt" parameter is set.
If Then	**If** the user can't manage the quizzes, **then** nothing will happen.

FN No:	FR2
Shall	Users with correct access level **shall** be able to update, delete and add quiz variants.
While	**While** the user is a superuser or have right access level on their account.
When	**When** the user makes a request to the server.
Where	**Where** the "opt" parameter is set and the user has permission to manage quizzes.
If Then	**If** a user, with no write-access, tries to manage a quiz-variant the system will prevent them from doing so.

FN No:	FR3
Shall	**Shall** be used to control access to quiz
While	**While** a user is authentication and is authorized, they can manage quiz's.
When	**When** a user tries to access protected resources.
Where	Users with right credentials can access **where** the "auth.py" file is located.
If Then	**If** user is authenticated but not authorized, **then** certain parts of the website is not accessible.