**Assignment 6**

Changes:

1. Sign-off from team members added.
2. Requirements list referenced added.
3. Description of the module added.
4. Project diagram added
5. Stubs implemented
6. Parameters of methods in pydoc added
7. Failure test and storage in persistence added in Unit test
8. README.txt

Group members:

Olmstead, Kyle

Nuang, Xiaochuan

Shevchenko, Arsenii

Langer, Aidan

Harrhy, Jack

Sign-off from each team members:

A close up of text on a white background

Description automatically generatedA close up of text on a whiteboard

Description automatically generated

A screenshot of text

Description automatically generated

Project diagram

Project quiz has five main modules, flask front end and persistence. Components are presented on diagram below.

A black and silver text on a white surface

Description automatically generated

Project diagram

Module take quiz

According to [Requirements List](http://www.cs.mun.ca/~brown/cs2005/Notes/project/projrequirements.html):

According to 1.3.1 User can take permitted quiz and 1.3.2 Access limitations are checked:

User can take quiz if date permits and number attempts are not exceeded.

Module take quiz has functions to access limitations by a number of attempts and availability.

According to 1.3.4 User can have multiple attempts:

User can take quiz specified number of attempts in specific period of time

According to 1.3.14 User can pause quiz

User can pause quiz, quiz in progress will be stored in persistence without indication complete.

According to 1.3.17.1 All answers are auto-saved upon quiz submission:

Picked answer are saved before submission.

According to 1.3.17.2 A confirmation notification for submission appears:

When user clicks submit button, user is transferred to submission page.

Module take quiz functionality and design

All methods are presented in pydoc documentation.

The module takes quiz receives an object of a class quiz from the persist storage and keeps it as a parameter self.quiz\_in\_progress.

Module take quiz can manipulate data from an object of a class quiz.

Module take quiz creates a question set from an object of and returns question and set of answers for questions. (def setQuestions())

Module take has function getQuestionText(index) that used to present questions text one by one and function getQuestionOptions(index) to present set of answer for each particular question.

Answers that User picks are stored in dict structure where key is id of the question, and value is picked answer, function def setAnswer(quest\_id, answer).

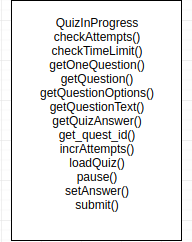
Module take quiz checks number of attempts function def checkAttempts(), to verify that number of attempts are not exceeded.

Module take quiz check if quiz is available, ie time period is correct (def checkTimeLimit())

Module take quiz has to options to pause quiz def pause(), that stores quiz in persistence storage but do not set flag complete to quiz.

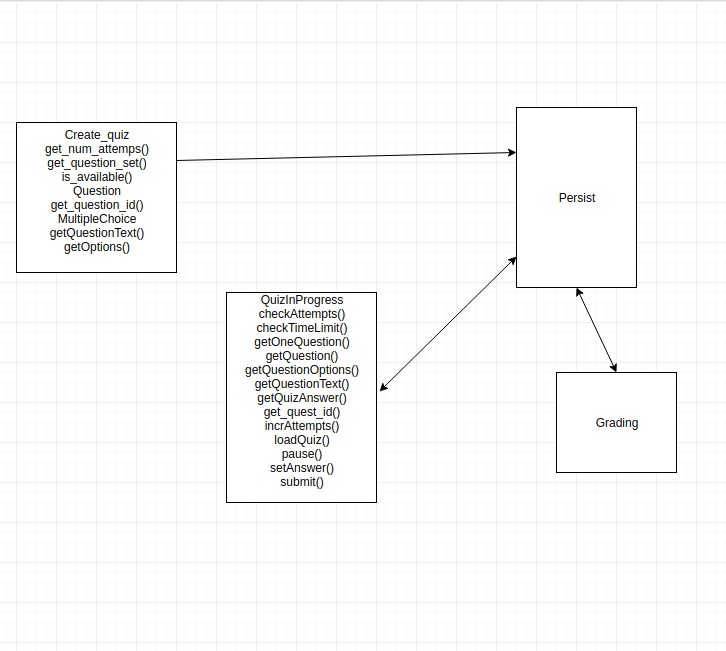
Module take quiz has function def submit() that stores quiz in persistence and module 4, 5 Grading and Viewing Results can have access to it.

Module has design that coupling with module create Quiz because it uses instance of class Quiz as quiz in progress. Main methods are presented in diagram below.



Main methods of module take quiz

Coupling with module create Quiz are presented in diagram below.



Coupling with module create Quiz

User case description:

Student wants to take quiz. User have to log in in the system (Module 1: Authentication), then student selects quiz that was created by professor (Module 2: create quiz). Then student taking quiz (Module 3: Take quiz), students answers questions and submit quiz. Student’s quiz is graded (Module 4: Grading). And student can view results of the quiz (Module 5: Viewing results)

Title: Take Quiz

Primary Actor: Student

Scope: quiz

Level: User goal

Stakeholder and Interests:

Student wants to take a quiz

Preconditions:

Student already logged in

Quiz created

Main success scenario:

1. User open quiz

2. User chooses answers for questions

3. User submit quiz

4. User moved to page of submission

Extension:

1a. Quiz is not available

1b. User spent all attempts

3a. User pauses the quiz