***Note: The use case’s functionality hasn’t been altered since Assignment 5.***

**Title**: View Class Results for an Individual Quiz

**Primary** **Actor**: Instructor

**Stakeholders** **and** **Interests**:

Instructor – Wants to review class results for an individual quiz

Student – Wants correct information to be logged under their ID

**Precondition**: User with Instructor privileges logged in already

**Trigger**: Instructor needs to log results into his own separate file for end of semester review

**Main Scenario**:

1. Instructor selects the “Grades” section of the interface
2. SuD shows the list of quizzes that have been taken
3. Instructor selects a specific quiz (e.g Quiz 1)
4. SuD lists the names of students and the grades they got on the quiz
5. Instructor makes note of the student’s grade in their own file
6. Steps 2-5 are repeated for as many quizzes as needed.
7. Instructor logs out of the system.

**Extensions**:

No extensions have been elaborated on at this time.

**Clear Connection between Module and Functionality**

The basis of the Quiz\_Result implementation is going to be the “\_quiz\_dict” dictionary member variable of the QuizResult object. When called, it will take the dictionary from Persist, supplied by the Take Quiz module, and iterate over it, adding each attempt’s information to the \_quiz\_dict dictionary, with the quiz names as keys and a list with the attempt data (student email, list of questions, list of choices, list of correct answers and list of student responses) for each individual attempt

Per the requirements list (COMP2005 Project Requirements Sheet.html) each of my functions is tailored to satisfy a specific requirement, which should be clear from the naming of the function. For example, the ins\_class\_part function is meant to return the class participation as a percentage value to the instructor. The stu\_quiz\_grades function is meant to return a list of strings representing a summary of the student’s grades for a specific quiz. The naming convention makes the connection clear. The flask front end will then use these functions to display the appropriate data as needed.