**Description of the status of the project in terms of what deliverables are completed and operational**

The original deliverables for my assigned task in the project were:

**Programmer 1: Login, authenticate, persistence**

* password authentication
* two classes of privilege: student and instructor
* new account configuration
* storage and retrieval for other module

All these tasks are full functional in my project. Password authentication, two class of privilege and new account configuration are handled in the user.py class. They also a relevant flask front end along that uses html templates to provide a user friendly interface to the principal client.

The persistence, which is the storage and retrieval is handled by the persist.py class. It provides an interface for the storage and retrieval of data for the entire program.

Once at the main dashboard, the user can register their credentials and select if they are a instructor or not. To login they must use the same credentials they entered during registration other they are not allowed to login. Depending on the type of user they are directed to their respective dashboards of either an instructor or student.

**After leaving my group I was assigned additional tasks to show that my persistence modules was indeed working. For that I had to provide functionality for creating a quiz for an instructor and taking a quiz for the student.**

**Programmer 2: Create Quiz**

* instructor only
* manage question bank
  + MCQ's questions, choices and answers
  + modify,copy existing questions
  + MCQ can have multiple correct answers
* manage quiz content
  + instructor access limited to owner of quiz
  + add/remove quiz questions
  + start and end time
  + students taking the quiz
  + modify and copy quiz

**Programmer 3: Take Quiz**

* check access
  + student and time
* navigate questions

Both these functionalities are complete. These functionalities are handled by the user and quiz classes, along with the app.py providing the flask front end.

An instructor alone can create a quiz. At first an instructor creates a question bank by going into the manage question bank tab on its dashboard and adding new questions to the bank. These questions are stored in the data base using the persist module. Additionally these questions can be removed from the question bank as well. Then it can go to the manage quiz tab and create a quiz and add questions from the question bank and set the start and end time. The quiz is stored in the database using the persist module. Additionally that quiz is retrieved from the database using persist. The quiz can be modified or deleted from the database.

Only registered students are allowed to login and take the quiz by going to the take quiz tab and from the listed quizzes. There is an option to create a quiz on the student dashboard but it is misleading as you cannot actually create a quiz. After navigating through the quiz and choosing the answers, the student can submit the quiz. Their quiz attempt is stored using persist and can retrieved using persist by choosing the show attempt option.

These modules clearly exemplify a working persist module that efficiently stores and retrieves data for the entire program without making other potential programmers working on the project deal with the database.