For our assignment 3, we successfully implemented several key features that were required by the project description. All the students are able to register and log in to the website securely using bcrypt-hashed passwords. There are two views which depend on the user type, students and instructors. Students can view their grades for all of the assignments, labs, midterm, and final exam. They can also submit a remark request if they have any questions by simply entering a reason beside a specific assignment. These requests are stored in the database and instructors can see from their views. Students can also submit feedback for instructors and it is anonymous so that instructors only see the feedback .

Instructors are able to view the grades of all students which is shown in several tables categorized by the assignment type. They can also update the mark by simply clicking on the button on top of the page and updating marks using a form, where they select or input a student, choose an assignment or exam, and input the mark. Moreover, instructors can view all the remark requests submitted by students and update the request status as “approved”, “pending”, or “rejected”. Instructors can see all the feedback written by students and can mark as “Reviewed” or “Open” just simply toggle it. If it is reviewed it will show a color of green, and if it is open meaning not yet reviewed, it will show a color of red.

There are some struggles while working on the assignment as we are splitting work to three sections: frontend, backend, and integration/testing. Everyone did their jobs well but integrating the frontend and backend together is kinda a disaster as there are lots of inconsistencies for the names, and we spend lots of time debugging. Also, we struggled with including correctly rendering dynamic content using Jinja2, specifically when displaying flash messages since in the beginning, it shows nonsense that the flash message does not appear in the right page or even ont displaying.

The work for this project was divided into three parts, so one person is responsible for the frontend stuff, one person is responsible for the backend, and the last person is going to integrate the frontend and backend and test the whole website. We use several bonus features, such as including JQuery for dynamic updating, and also use a datalist inputs for instructors to select student usernames when entering marks more intuitively. In conclusion, our webpage meets all the functional requirements and we collaborate well to provide a user-friendly interface for both students and instructors.