Senior Project Proposal

Department of Computer Science - Calvin University

Title: Online Grader Helper App
Authors: Michael Sisko & Emily Costa
Date: May 20, 2021
Advisor: Victor T. Norman

Background and Problem

Grading is already a time consuming and redundant task, and Moodle's user interface doesn't aid in making the grading process more efficient. When a professor or grader wants to grade an assignment, you can either enter the feedback directly into Moodle or download a CSV file, enter the feedback, and then reupload it. One issue with both of these methods is that the feedback has to be manually entered each time, and the same feedback is often repeated on numerous student submissions. Another issue, when entering feedback into the CSV file, is that you have to fit all the feedback into one line in a single cell. And, if a grader decides to change the number of points taken off for a specific mistake, they have to manually go back and change all of the scores for that specific mistake. This is an important issue to be addressed because the less time professors and graders spend grading, the more time they have available to help students, replying to emails, and other important tasks.

Brief Description of Solution Being Provided

Originally, graders would use the interface provided by Moodle, but since this interface is clunky and does not allow for streamlined feedback; we plan to develop an application to make the process quick and painless. For every homework students are assigned, graders must evaluate their work. This process involves careful review of the submitted work, oftentimes different students will make the same error. Considering this, it would be tedious for a grader to re-enter or modify feedback should they change their mind. Furthermore, we will implement feedback consistency, meaning that feedback value and wording changes will be correspondingly reflected wherever it is called.

With our application, a student grader or professor would download an assignment's CSV file from Moodle, upload it, and then our app will parse through the file to get all of the students' information. The application will display a full chart of the students' full name, their Calvin email, their grade for the assignment, and the feedback. The CSV file does provide an image URL of each student, but we have decided to omit this information from our application to promote objective grading. The grader can proceed through the list of students, entering a string of feedback, and the value of the deduction. All of the feedback strings and their point values will be stored into a saved list. Using this list, a grader can then just check off the feedback that

applies to each student's submission. This eliminates the redundancy of typing in the same feedback for multiple students, through selecting the relevant custom feedback. Our application will also keep track of the feedback given to each student, so if the grader modifies the custom feedback or the deduction value, if saved, the change will be reflected globally. Finally, the application will also display assignment statistics: the frequency of each feedback, the average score, standard deviation, median, and mode. Once an assignment has been graded, the grader can then export the modified CSV file and upload it to Moodle.

In the development of our project, we will be using Angular, Bootstrap, Apex Charts, and GitHub. Throughout the duration of our project, we will be practicing end-to-end development using Cypress.

Research or Development Objective

This is a software development project. With this web application, we aim to streamline the grading process by providing a central location where a grader can give a score and provide feedback. Our goal is to create an application that is simple and efficient to use for graders, and will require no training.

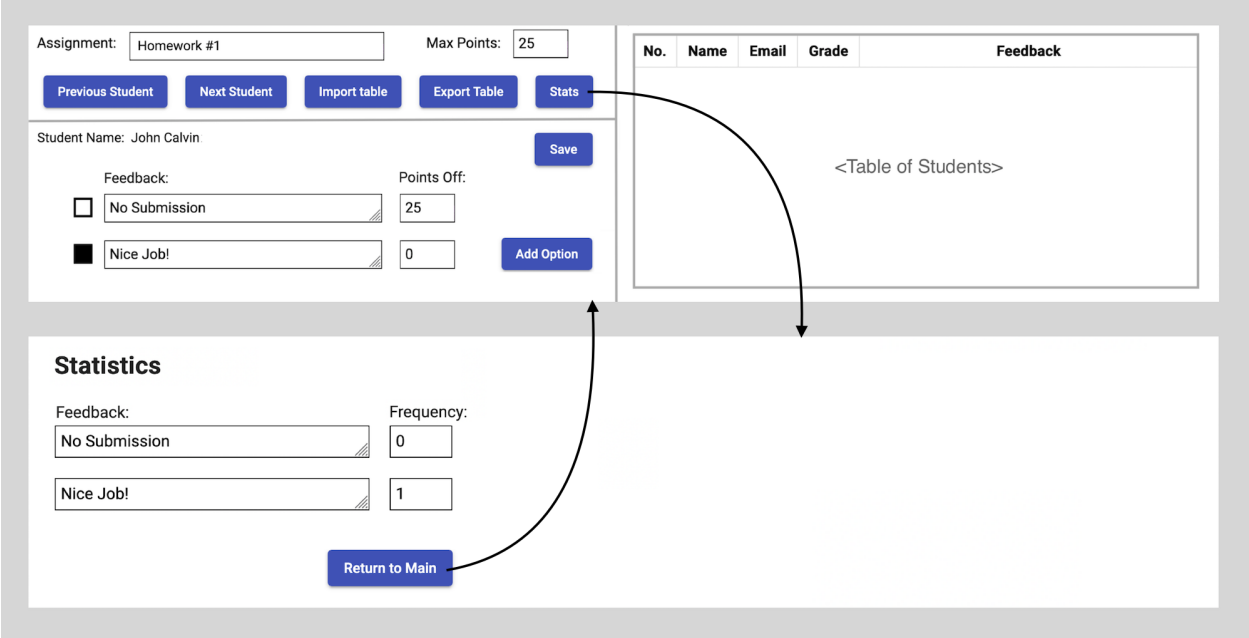
Your Interest and Qualifications

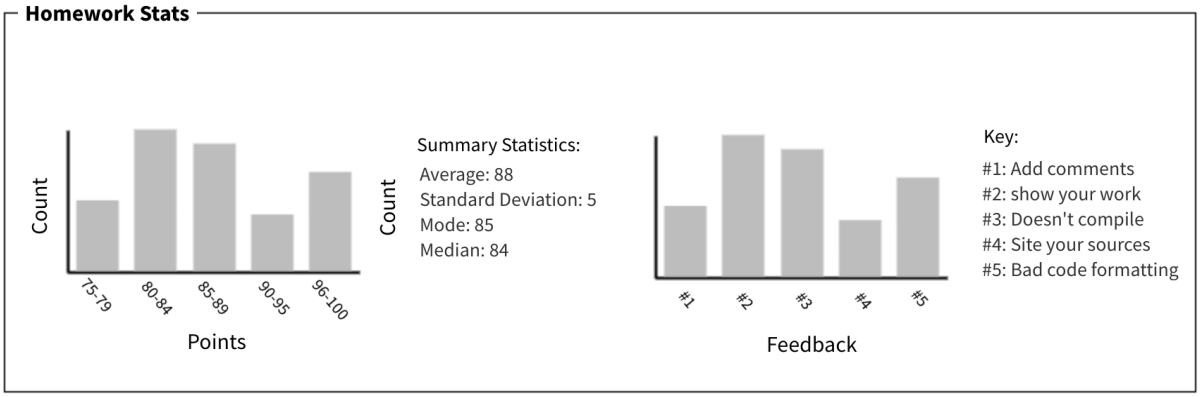
Our team has taken CS-336 Web Development, and we thought that this project would play into our skillset as programmers. Web Development covered angular web applications, and user-interface design at a considerable level, which provides us with confidence that we will have a successful project. We also believe that our summer internships will provide us with new developing skills and experience that would be useful and applicable to this project. We decided to work on this project on Mondays, Tuesdays, and Thursdays from 8pm to 10pm EST.

Collaboration with Advisor, Outside Experts and Users

We plan to meet with Victor Norman on Mondays at 7pm EST. As it stands, Victor Norman will be the sole advisor on this project. For usability testing, our users will be current and past course graders, as well as professors.

UI Design Mock-ups





Detailed Schedule

Order of Implementation:

*While implementing, we will be making new e2e test cases to test each feature

1. Implement UI on front page to be responsive
2. Add a new feedback and deduction score to the list
3. Delete a feedback and deduction score from the list
4. Be able to import CSV file
5. Parse through CSV file to display the student's name and email address
6. Display contents of the CSV file in a nicely formatted chart

7. Be able to navigate through the CSV file by selecting “Previous Student” and “Next Student”
8. Be able to click on the student chart to select the student you’d like to grade
9. Select and apply feedback to various students’ assignment
10. Be able to add feedback/edit the CSV file
11. Keep track of the feedback and score, so when you change the score in the list it changes for everyone
12. Export CSV file
13. Keep tally of the feedback frequency
14. Keep track of the scores to compute the assignments average, mode, standard deviation, and median
15. Format homework statistics into nice charts
16. Clean-up UI

Grab the overall schedule from https://cs.calvin.edu/courses/cs/396/schedule.html, adapt to summer / fall, and for the summer part, add deliverables/checkpoint/deadline every 2 weeks.

Test-driven development: would love to see you do this right from the beginning.

Semester	Due Date	Milestones
Summer (CS 396)	May	May 20: Project proposals should be drafted and delivered to your project advisor. May 31: Submit a project proposal to your advisor and the project course coordinator. The department will review the proposal and respond by
	June	June 7: **Deliverable** Implement UI on front page to be responsive, Add a new feedback and deduction score to the list, Delete a feedback and deduction score from the list June 21: **Deliverable** Be able to import CSV file, Parse through CSV file to display the student’s name and email address, Display contents of the CSV file in a nicely formatted chart A project status session may be scheduled. Be prepared to present an updated version of your project proposal and, if appropriate, to demo it to the project class.
	July	July 12: **Deliverable** Be able to navigate through the CSV file by selecting “Previous Student” and “Next Student” July 26: **Deliverable** Be able to click on the student chart to select the student you’d like to grade

	August	August 9: **Deliverable** Select and apply feedback to various students' assignment, Be able to add feedback/edit the CSV file August 16: Present a project status report in a department seminar. August 16: Submit a draft project website with supporting materials to the project course coordinator. The department will review these materials and respond by the beginning of the CS 398. August 30: Review the project guidelines with your faculty advisor and, if appropriate, attend any scheduled project status sessions.
Fall (CS 398)	September	September 13: **Deliverable** Keep track of the feedback and score, so when you change the score in the list it changes for everyone September 27: **Deliverable** Export CSV file
	October	October 11: **Deliverable** Keep tally of the feedback frequency October 25: **Deliverable** Keep track of the scores to compute the assignments average, lowest, and highest score
	November	November 8: **Deliverable** Format homework statistics into nice charts November 22: **Deliverable** Clean-up UI
	December	December 6: Present a final project presentation in a department seminar. December 6: Submit your final project deliverables to the project course coordinator: ● Final Report ● Website ● Compressed archive (tar or zip) of your final project, including all code and the final report, for archival purposes. The department will review your work and respond by the end of final exams.