

Grading Tool for Web-based Reports

CS4991 Capstone Report, 2021

Peter Chen

Computer Science

University of Virginia

School of Engineering and Applied Science

Charlottesville, Virginia USA

pc5kjv@virginia.edu

ABSTRACT

A team of two other UVA Computer Science students and I are developing a tool that allows for grading of web-based reports. Although word processors like Microsoft Word have features that allow for graders to provide commented feedback, there is no support for grading features such as distribution of point values or display of a rubric. The alternative of providing the grade and feedback on a separate document compromises communication, obscuring exactly what part of the report a comment may be referring to. Thus, the aim of this project is to create a tool that allows for the annotation of websites with point-values/feedback alongside a customized rubric and that can provide different privilege levels for students, teaching assistants, and professors.

The tool, still a work in progress, currently exists as a web application that allows for submission of a rubric and a website. It then renders the site in the web application alongside the rubric and allows for the user to make annotations on the site as well as indicate point values for each rubric category. Some limitations remain, however. The tool is only able to parse and render websites built using Google Sites. Additionally, the approach of parsing and rendering the site in an iframe creates problems with displaying embedded images. Despite limitations, we hope that this grading tool will ease grading of web-based reports for teaching staff. For future work, we look to adapt the tool to a browser bookmarklet or extension that allows for grading/annotation directly on any site.

1 Introduction

Malcolm Forbes, publisher of *Forbes* magazine, once said, "Failure is success if we learn from it." Few, if any, students will ever say their goal in school is to make mistakes on an assignment. However, many likely would

share the sentiment that a large portion of learning actually comes from failures rather than successes. A core part of a high-quality education, thus, may be having high-quality feedback which allows students to learn from their mistakes.

At UVA, the Computer Science course *HCI In Software Development* teaches about human-computer interaction (HCI) and user-centered design. The current instructor of the course, utilizes designing a web-based report as an assignment to teach these concepts. The problem with grading these types of assignments is that it's difficult for graders to provide clear feedback to students since they do not have the ability to make marks on the assignment itself. Previously, graders instead made comments on a separate document containing the rubric. However, it may be confusing to students what specific parts of the report specific comments refer to. Removing the need to switch between two documents can also expedite the grading process.

Our client is the instructor of the *HCI In Software Development* course. The goal of this project is to develop a tool which will allow for graders of the course to directly annotate a webpage with comments and point deductions, showing students clearly where and why they lost points. As an additional benefit, the online tool should provide an avenue of communication between students, teaching assistants (TAs), and instructors, allowing parties to comment on the grading.

2 Background

In HCI, students are divided into teams to work on a web-based report that teaches concepts such as requirements gathering, prototyping, and usability. The web-based reports take on the form of a Google Site that is iterated

upon in multiple milestones, known as phases, throughout the semester. Each phase has specific sections and thus may consist of multiple pages. For example, phase 1 includes research and design sections, while phase 2 involves specifying a concept statement, requirements, and designing flow and usage models.

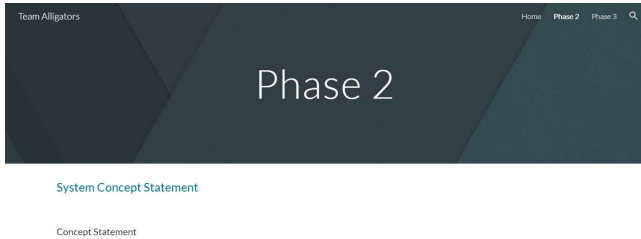


Figure 1: Models Section of Phase 2 Report

[Be sure to refer to figures somewhere in your text if you include them in your report.]

While the immediate goal of this project is to develop software for use in the HCI course, we hope that it will eventually serve as a general-purpose tool for grading these types of assignments.

3 Related Work

The two tools most similar to the one that we are developing are Hypothesis and Gradescope. We plan to integrate some of the features of these into our tool later in development.

Hypothesis is an open-source software that enables “sentence-level note taking or critique” [1]. It can be installed and used as a Chrome extension. The current advantage of Hypothesis to our grading tool is that it allows for users to annotate directly on a great number of sites as well as make comments on those annotations. Additionally, even users that don’t possess the extension on their browser can view the annotations via a proxy. There is also a Groups function, which allows for annotations to only be visible to users belonging to the group which created them. However, this tool does not possess any of the grading capabilities desired, including assignment of point deductions alongside the comments and rubric. The Groups function also does not meet the desired privilege level structure. For instance, a professor would not be able to make a comment to a TA on an annotation without it also being seen by students.

Gradescope offers grading tools for homework assignments, exams, and code. It began in the Computer Science department of UC Berkeley [2], but is now being

used in a number of schools, including UVA. Gradescope allows instructors to create a custom rubric through a web interface and then grade assignments using this rubric. It has a lot of the same features as our desired software such as rubric creation, graded commenting, and association of comments to specific text and rubric criteria. One limitation of Gradescope, however, is that it can only grade reports in the form of PDFs. While websites can be converted to a PDF form, this conversion potentially compromises the formatting and functionality of the site. Gradescope also only allows for points to be deducted, removing the possibility of rewarding bonus points.

4 System Design

Before testing the tool in UVA’s HCI course, basic requirements must first be met. The core requirements for this system are as follows:

Graders should be able to:

- Create a custom rubric for each phase
- Provide graded annotations on the report
- Categorize annotations according to rubric criteria shown alongside the report

Students should be able to:

- View graded annotations on the report
- View rubric and points per category

The current version of our grading tool is a web application built on ReactJS for the frontend and Django for the backend. Before grading a submission, the professor uploads a rubric file (interface shown in Figure 2) showing how the phase will be graded. This rubric file should take the form of a JSON file matching a specific template which specifies information including section names, criteria, and points possible.

Figure 2: Submit a Rubric Interface

On the tool page, graders can view and grade submitted reports. To see the desired report, the semester, team, and phase should be specified. This information will be used to

form the URL of the appropriate Google Site. Information on this site is parsed and re-rendered in an iframe in the tool.

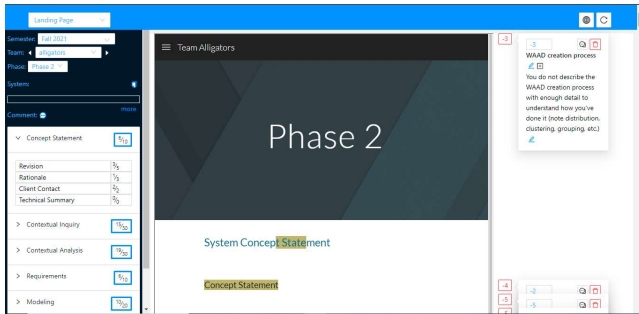


Figure 3: Tool Interface

Graders can make graded annotations on the report through the iframe by highlighting text and then clicking the rubric section that they want it to fall under. They then can edit the annotation’s text and point value, adding or deducting points.

Once grading has been completed, graders can export the page, creating a new page with all the finalized comments and point values (see Figure 4). This page can then be shared with the students.

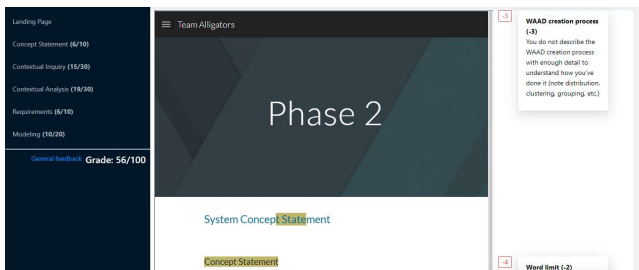


Figure 4: Export Interface

5 Results

The grading tool meets the core requirements specified. Comments about the report are clearly associated with specific rubric criteria as well as specific sections of the report. Graders also can easily make changes to their annotations and point assignments as well as navigate between submitted reports with only a few clicks. The tool will be tested in the HCI course soon.

Despite its useful features, there remains a number of notable restrictions that reduce its usability. First, the tool is currently restricted to only work on pages built with Google Sites. This restriction severely limits the broader applicability of the tool. Second, the format of the site is altered. In Figure 1, there exists a navigation bar containing

tabs for each of the different phases. This format is slightly altered when the page is rendered in the grading tool’s iframe, as seen in Figure 3 (the navigation bar is no longer present). This change was implemented intentionally in order to allow graders to navigate using the tool’s built-in dropdown menu for phases. The solution works in the short-term, but changing the format of the submission is less than ideal as the site’s design could impact students’ grades.

6 Conclusions

The developed grading tool allows graders to create custom rubrics, make graded annotations, and categorize these remarks according to the created rubric. Despite current limitations, the grading tool developed has potential to increase the clarity of feedback for students as well as ease the grading process for teaching staff.

7 Future Work

We hope to eventually make the grading tool more broadly applicable instead of being restricted to Google Sites webpages. This functionality has already been implemented successfully in Hypothesis, which allows for direct annotation on a great many sites. Since Hypothesis is an open-source project, we hope to be able to learn from the work they’ve done and incorporate a similar functionality to our tool. The final version of our project thus may take on the form of a browser extension rather than a web application.

We also hope to increase the tool’s usability. Taking inspiration from Gradescope, we’d like to provide a web interface for graders to create a rubric from, rather than uploading a JSON file, which requires some technical knowledge and is more error-prone. Another desired feature to be implemented is the ability to comment on annotations, especially the ability to make comments with different access levels so that instructors may make remarks on teaching assistants’ grading without this interaction being visible to students.

8 UVA Computer Science Program Evaluation

There were a few Computer Science courses that stand out as relevant to the development of this tool. Programming Languages for Web Applications served as a foundation for knowledge regarding web applications, providing a basic understanding of HTML, CSS, and JavaScript. Mobile Application Development taught JavaScript and React Native, which differs in some ways from ReactJS but was nonetheless useful in development as it covers key concepts such as arrow functions and components. Some members of the team have also stated that Advanced Software

Development Techniques would be useful in understanding Django, though much of this backend work was done by a former team member.

As we look to improve our tool by integrating features from open-source projects such as Hypothesis, however, the task of understanding large code bases written by other individuals can be daunting and challenging. A course that teaches students how to familiarize themselves with large code bases and/or provides them experience working with open-source projects would be greatly beneficial.

ACKNOWLEDGEMENTS

I thank Dr. Panagiotis Apostolellis for providing me the opportunity to work on this project. I also thank fellow UVA students Andrew Song and Andrew Wang for their contributions to the project.

REFERENCES

- [1] Hypothesis. 2020. About Us. (March 2020). Retrieved October 26, 2021 from <https://web.hypothes.is/about/>
- [2] Gradescope. 2021. About Us | Gradescope. Retrieved October 26, 2021 from <https://gradescope.com/about>