Assessing the Impact of Specifications Grading on a Data Visualization Course

Alark Joshi
Department of Computer Science
University of San Francisco
San Francisco, CA, USA
apjoshi@usfca.edu

***Abstract*—In this Research-to-practice paper, we share our experience with implementing Specifications Grading in two separate offerings of a Data Visualization course. The course is taught in a Computer Science department every semester. Based on the analysis of the student assignments, final project, and the final grade received by the students, the students in the offerings with Specifications Grading performed better or as well as without it. We found that Specifications Grading led to students taking control of their own learning and restored rigor to the course. Student feedback for Specifications Grading was mostly positive, and multiple students encouraged the instructor to keep using it in future offerings of the course.**

***Index Terms*—pedagogy, student motivation, learning**

I. Introduction

Instructors for upper-division courses are faced with the challenge of engaging and motivating the entire class and having them care as much about the topic as they do. This situation is particularly exacerbated in the case of electives, where there is a mix of students who are extremely passionate and motivated about the topic and students who just want to get it done to fulfill the requirements for their degree. Keeping the motivated students engaged and challenged and providing a minimum standard for the rest of the students, who may not be as interested in the topic, is particularly challenging.

Specifications Grading provides an excellent solution to this problem of varied motivation. Specifications grading draws principles from popular pedagogical strategies such as pass/fail grading, contract grading, and competency-based education. Students know exactly what they need to do to get a specific grade in each assignment and the entire course. Additionally, students get a choice to determine their grade depending on their motivation, time commitments, or other challenges. Students ask for feedback *before* the deadline rather than ignoring the comments provided with the grades.

We employed Specifications Grading in a cross listed upper-division undergraduate and graduate-level Data Visualization course for two semesters (Spring 2022 and Fall 2022) and compared it with two prior semesters (Spring 2021 and Fall 2021) where the course was taught without Specification Grading. We used a combination of student performance on programming assignments, assignment quality, final project performance, and student feedback across the four offerings of the course to compare the impact of Specifications Grading.

In this paper, we present the various assignments, Specifications Grading prompts provided for the assignments, the preparation required to make Specifications Grading work, and the practical benefits and challenges associated with deploying it in the classroom.

II. Related Work

Student learning and motivation has been actively researched in higher education. Mastery Learning [1] is one such pedagogical strategy that emphasizes student learning by providing students the opportunity to take into account the provided feedback and resubmitting their work for evaluation. Students need to demonstrate a high level of mastery on a specific topic before they can continue learning about subsequent topics in the course. Even if the students do not learn all the topics in the course due to some stumbling blocks along the way, the content that they have learned is well understood.

Cohen [2] mentions that feedback "is one of the more instructionally powerful and least understood features in instructional design." Formative feedback [3] is common in fields such as design, where it is common to get feedback from the instructor and peers too, in some cases, as you make progress on your assignment. In Computer Science, personalized formative feedback is not very common, sometimes due to the large enrollment in introductory courses and sometimes due to the fact that it is very time-consuming. Benotti et al. [4] developed an automated tool to provide formative feedback to students in beginners and found it to be quite useful in providing immediate feedback and increased engagement. Providing an automated test framework for students to test their work on is more common than personalized formative feedback. Paiva et al. [5] provide a detailed overview of the various automated testing approaches used by instructors in Computer Science courses to evaluate assignments as well as to provide feedback to students as they work on completing the assignment.

While most instructors want to help students improve and spend a significant amount of time grading and providing meaningful feedback. The challenge with feedback is that students generally receive it with a grade after the assignment has been completed and while they are working on the next assignment. There frequently isn't a chance to do anything meaningful with the feedback received from the instructor.

Butler [6] conducted a study where they found that among the group of students who received only feedback, feedback with grades, and only grades, the students in the only feedback group showed most improvement in the course throughout the course than students in the feedback with grades and only grades groups.

Specifications grading [7] allows students to take into account instructor feedback and resubmit their assignment once to get full credit for their work. It does not allow multiple resubmissions like with Mastery Learning, but it does allow students to incorporate the feedback and resubmit once for improved learning.

III. Methodology

We implemented Specifications Grading in a Data Visualization Course that is taught in the Computer Science Department at our university. The course introduces data visualization principles as well as provides hands-on experience with data visualization techniques through the use of Tableau, P5.js, and D3.js. Students learn about data visualization through weekly programming assignments, reading book chapters and research papers, weekly multiple choice quizzes, and a final summative project that demonstrates their understanding of data visualization principles using d3.js.

The course was taught in Spring 2021, Fall 2021, Spring 2022, and the Fall 2022 semester by the same instructor. The first two offerings (Spring 2021 and Fall 2021) of the course were taught using a lecture-based format with assignments (without any Specifications Grading components), weekly quizzes, and a final project, whereas the last two offerings (Spring 2022 and Fall 2022) of the course were taught using Specifications Grading for the programming assignments only (not the weekly quizzes or the final project).

Specifications grading was used for nine individual assignments (A2-A10) that the students completed throughout the semester. For each assignment, students were given detailed specifications for obtaining a C/B/A/A+ on each assignment. To get a B, student work had to meet the specifications for a C and their assignment was required to meet extra criteria in terms of features, number of techniques implemented, and so on. Similarly, to get an A, student work had to meet specifications for a B and the extra requirements to get an A for that assignment, and so on. For example, Assignment 3 required students to implement visualization techniques to represent distributions in data. The most straightforward visualization for visualizing distributions is a histogram. Students who were aiming for a C were required to write code in P5.js [8] to draw histogram for any one variable in a dataset of their choice from the CORGIS dataset repository [9]. If a student used a dataset about cars for their assignment, then they may show the distribution of the horsepower variable of all the cars in the dataset.

The specifications for a C-grade (75/100) were as follows:

- The histogram must have labeled axes (x- and y-).
- The histogram must have a faint background grid to help the viewer read the chart.
- The data must be loaded in from a CSV file, and it must work on Vizhub [10].

We used Vizhub.com [10] as a website for students to host their code online to create a portfolio for the semester and for the grader and the instructor to access their code.

For the same assignment, the specifications to get a B grade (85/100) required students to implement a box plot for a different variable in the dataset using P5.js. The specifications to get a B were as follows:

- Complete the requirements to get a C *and*
- Implement a box plot for a different variable from the same CSV file
- The box plot must display the minimum value, 1st quartile, median, 3rd quartile, and maximum value
- The data must be loaded in from a CSV file, and it must work on Vizhub.

For the same assignment, the specifications to get an A grade (95/100) required students to implement a strip chart [11] for another variable from the same dataset using P5.js. The specifications to get an A were as follows:

- Complete the requirements to get a B *and*
- Implement a strip chart of another variable from the CSV file to display the distribution along the y-axis
- The strip chart must contain jittering to reduce overlap. Hint: Consider using the random() function in P5 for this.
- The data must be loaded in from a CSV file, and it must work on Vizhub.

The students also had an opportunity to be challenged to aim for an A+ grade (100/100). To get that grade, their assignment needed to meet the following specifications:

- Complete the requirement to get an A *and*
- Identify and highlight all outliers for the box plot
- Implement interactive tooltips for the histogram. Provide tooltips to the user, so they can examine the values under the mouse pointer.
- The data must be loaded in from a CSV file, and it must work on Vizhub.

According to the Specifications Grading methodology, if a student's assignment does not meet the specifications for the grade they were aiming for, they **receive a zero** on the assignment. Students get two *tokens* per semester to resubmit an assignment and get credit for that assignment. This ensures that students take the assignment seriously and ask for help (formative feedback) and confirmation that the assignment meets all the specifications *before* the due date, rather than waiting for an assignment to be graded to see what grade they received. As can be seen from student feedback later, some students found it stressful, but students also commented on how the feedback and help they received from the TA and the instructor helped them with their assignments.

A. Assignments Overview

Here is an overview of the assignments in the course. Due to the fact that the students enrolled in the course had limited exposure to HTML, CSS, and JavaScript, the assignments are structured to give them practice with those aspects as well as build their visualization expertise as the semester progresses. We start with Tableau, then transition to P5.js [12] due to its ease of use and gentle learning curve before transitioning to D3.js [13]. D3.js is the de facto library for interactive data visualization on the web, but is known to have a steep learning curve. We use P5 for the first part of the semester due to its gentle learning curve and its comprehensive library of helper functions. Through the use of P5, we find that students are able to transition to D3 better and appreciate the impressive power of D3.

The students worked on ten assignments throughout the semester. Nine of them had aspects of Specifications Grading incorporated into them, with specifications for a C/B/A/A+ grade. Here is the list of the assignments that the students worked on during the semester:

1) **Data Exploration with Tableau** - This assignment provides a chance for students to get familiarized and learn about data exploration using Tableau Software [14]. Tableau facilitates a quick understanding of the variables in the data and their characteristics (min/max, distributions, and so on). *Note:* This assignment did not have any specifications that the students had to meet.
2) **Visualizing Amounts using P5.js** - This assignment requires students to gain hands-on experience with implementing visualization techniques for visualizing one-dimensional values using techniques such as bar charts, dot plots, and heat maps using P5.js.
3) **Visualizing Distributions using P5.js** - This assignment requires students to implement techniques to show distributions in a single variable in a dataset using a histogram, box plot, and a dot plot using P5.js. This assignment and the related specifications were described earlier in Section III.
4) **Introduction to D3** - This assignment requires students to implement elementary visualization techniques such as a bar chart, a scatter plot, and a heat map using D3.js [13].
5) **Multidimensional visualization in D3** - This assignment requires students to implement a bubble chart, a line chart, and a scatterplot matrix. Students could get extra credit by displaying a legend using the d3-legend library [15] for the bubble chart, by displaying tooltips to the viewer for the line chart, or highlighting a cell in a scatterplot matrix based on the mouse position.
6) **Color Scales in D3** - This assignment requires students to get familiarized with the various color scales available in D3 by implementing a bar chart with a sequential color scale, a diverging bar chart with a diverging color scale, and a qualitative color scale for a scatterplot with two quantitative and two qualitative variables.
7) **Interaction in D3** - This assignment requires students to implement familiar interaction techniques such as filtering, brushing, zooming, and panning using D3. For this assignment, they were given starter code for a multi-series line chart that showed the temperature of three cities over time.
8) **Graph Visualization in D3** - This assignment requires students to get practice with graph visualization techniques by implementing a node-link representation with force directed layout, an adjacency matrix representation, and edge bundling for a dense node-link representation.
9) **Geographical Data Visualization in D3** - This assignment requires students to learn about geographic visualization techniques by implementing an interactive choropleth map and a proportional symbols map.
10) **White Hat Black Hat** - The terms Black Hat and White Hat originate from old cowboy movies, where the "bad cowboys" would always wear black hats, whereas the "good cowboys" would wear white hats. The words Black Hat/White Hat have become more synonymous with computer security now, where hackers are labeled black or white depending on their intent. Students were required to *intentionally* create a "Black Hat" visualization and a "White Hat" visualization. A "Black Hat" Visualization could have one or several of the following problems with it:
    - Intentionally complex and cluttered visualization that is hard to understand and read.
    - Labels, axes, legends are intentionally misleading.
    - Bad/Wrong color scales are used.
    - Title of the visualization is intentionally skewed to bias the viewer's opinion of the visualization
    - The data has been transformed and processed in a weird and/or misleading manner
    - There is no evidence about the source of the data and how it was collected / where it was found

    A "White Hat" Visualization would have all the properties that we look for in a good data visualization:
    - The visual representation is clear and easy to understand for the intended audience
    - Appropriate color scales are used, and the legend is clearly labeled and visible with an appropriately sized font
    - Any data transformations are clearly mentioned
    - Gestalt principles [16] are followed.
    - Annotations are clearly visible and draw viewer's attention appropriately. Students were recommended to use d3-annotation [17] to add annotations to their visualization.
    - The sources of data are stated for viewers to see and verify

B. Statistical Analysis

We analyzed the student performance on the various assignments, as well as the quality of the various artifacts generated by the students in both the offerings (with and without Specifications Grading). We evaluated the individual assignments, the grade received by the students on the final

	A2		A3		A4		A5		A6	
	Mean	Median	Mean	Median	Mean	Median	Mean	Median	Mean	Median
Spring 2021	87.286	**96**	**98.107**	**100.0**	**93.268**	**96.0**	87.500	**99**	**83.071**	**92.0**
Fall 2021	88.333	95	84.167	85.0	77.917	85.0	75.833	85	76.250	90.0
Spring 2022	**90.607**	**95**	88.464	87.5	**88.393**	**92.5**	**88.893**	86	**80.357**	82.5
Fall 2022	**93.0**	95	**95.100**	**95.0**	72.7	90.0	**96.3**	**97**	76.250	**95.0**

	A7		A8		A9		A10	
	Mean	Median	Mean	Median	Mean	Median	Mean	Median
Spring 2021	**98.571**	**100**	**87.286**	94	**90.214**	**100**	90.214	94
Fall 2021	82.083	82.5	**90.455**	**95**	76.667	85	83.333	92.5
Spring 2022	79.464	77.5	85.179	**95**	76.429	85	**93.75**	**95**
Fall 2022	**82.7**	**90**	83	90	**86.1**	**91**	**94**	**95**

TABLE I
MEAN AND MEDIAN OF GRADES RECEIVED BY STUDENTS IN ASSIGNMENTS 2-10. ASSIGNMENT 1 DID NOT HAVE ANY COMPONENTS OF SPECIFICATIONS GRADING ASSOCIATED WITH IT. IN **8** OUT OF THE 9 ASSIGNMENTS, STUDENTS SCORED HIGHER ON ASSIGNMENTS IN SEMESTERS (SPRING 2022 OR FALL 2022) WHERE SPECIFICATIONS GRADING WAS IMPLEMENTED.

	Final Grade		Final Project	
	Mean	Median	Mean	Median
Spring 2021	86.259	**92.130**	89.566	94.0
Fall 2021	76.294	84.510	80.446	86.6
Spring 2022	**88.763**	90.765	**95.543**	**96.1**
Fall 2022	**89.068**	**91.550**	**92.420**	**97.1**

TABLE II
MEAN AND MEDIAN OF THE FINAL GRADE OBTAINED BY STUDENTS IN THE VARIOUS SEMESTERS AND THEIR RESPECTIVE SCORES ON THE FINAL PROJECTS IN THOSE SEMESTERS.

project, and their final overall grades to examine the impact of Specifications grading.

Compared to previous iterations of the same course (Spring 2021 and Fall 2021), the students were more engaged and motivated to meet the specifications for the various grade levels (A-C), and received equal or higher grades on average at the end of the semester. As can be seen in Table I, students received higher scores in 8 of the 9 assignments in either one of the semesters where Specifications Grading was implemented (Spring 2022 and Fall 2022).

As can be seen in Table II, students performed better on the Final Project in both the offerings of the course with Specifications Grading (Spring 2022 and Fall 2022). Figure 1 shows a graphical representation of the grade received by the students on the Final Project. The Final Project allows students to showcase their understanding of the material and D3.js skills. Students had higher quality final projects that demonstrated higher levels of understanding and technical skills in the Spring 2022 and Fall 2022 semester. That was reflected in the higher grades for the Final Project for those two semesters.

Figure 2 shows a graphical representation of the data from Table II. As can be seen from the data and the graph, students received a *higher overall grade* in the course in the semesters where Specifications Grading was implemented.

Based on a one-way ANOVA of the final grades of the students at the end of the semester, we found that the final grades for the semesters with specifications grading were statistically significant (F=3.85, p-value=0.006).

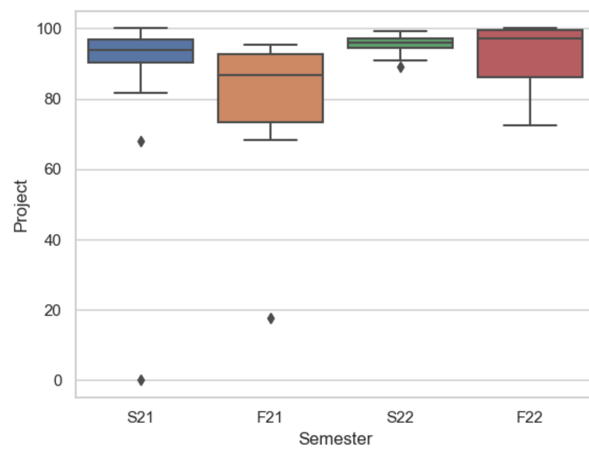

Fig. 1. Students produced higher quality projects in the Spring 2022 and Fall 2022 semesters, when Specifications Grading was used. The higher grades received by the students in those two semesters on their final project reflect their better understanding and d3.js skills.

C. *Student Feedback*

Students also seemed to like Specifications Grading, as evidenced by some of the feedback received at the end of the semester - "Fair assessment of the amount of effort put into the work. Definitely would recommend for future use." and "The grading specifications were really unique and very effective."

When asked whether the instructor should use Specifications Grading in a future offering of the course, students were enthusiastic and said, "I liked that the specification grading made it very clear what I needed to accomplish to receive a certain grade. I would recommend using it again." and "I liked that it allowed me to understand what I needed to do and be able to take advantage of it in case of a large workload during the week of the assignment. I do recommend it for future courses." Students seemed to enjoy it and one

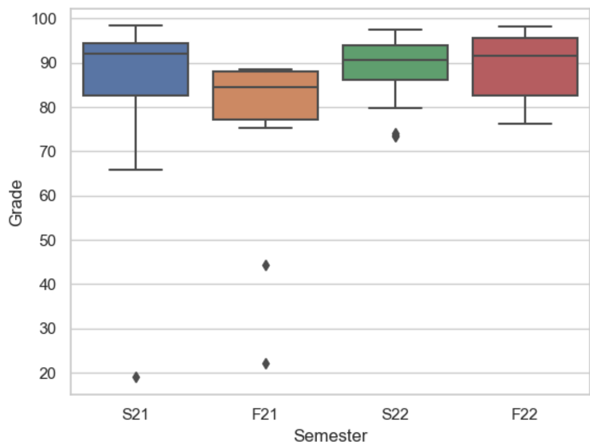

Fig. 2. In this graph, we see the overall distribution of the final grade received by the students in each offering of the course. While the median grade for the course offered in Spring 2021 is high, the overall distribution of the grade is quite spread out. The spread of the grades received by the students in Spring 2022 and Fall 2022 is narrower.

of them said, "I enjoyed the specifications of grading. I recommend keeping the Letter scale specifications." Students also commented on the feedback they received from the TA and the instructor *before* the assignment was due - "What I liked about specifications grading is that I learned all the feedback and comments that the Professor and TA gave me, and I would use it again for the next iteration of the course."

A few students commented that the specifications for getting an A/A+ were too difficult at times. One student said "I disliked the specification grading for the most part due to the difficulty level to get to an A so most of the times, I would get stuck on the C and could barely make it to a B grade."

IV. Discussion

Specifications grading definitely helped with increased student motivation. Students wanted to make sure they received a good grade on the assignment and asked the TA or the instructor for help, multiple times, *before* the assignment was due. This is an ideal outcome as the students can get formative feedback as they work on their assignment, rather than summative feedback day/weeks after the assignment has been submitted. Specifications Grading claims to restore rigor back into the course and that was definitely found to be true as the instructor could set high expectations for the 'C' level work that was required to complete an assignment.

Another claim of Specifications Grading is that it helps with the grading of assignments. This claim was also found to be true, as the assignments either met the specifications or did not. You had to either give the student a C/B/A/A+ or a zero (if the assignment did not meet the required specifications).

Students were sufficiently challenged to incrementally work on increasingly difficult aspects of an assignment. If they ran out of time or had other commitments (other courses or work), they could turn in what they had completed and be confident about the grade that they would receive in the course.

In terms of preparation, Specifications grading does require the instructor to clearly provide the specifications for each assignment. That can be somewhat time-consuming and needs to be taken into account when considering Specifications Grading.

V. Conclusion and Future Work

In this paper, we present our experience with incorporating Specifications Grading into a Data Visualization offered in a Computer Science department. Students performed equally well or better in most assignments in the semesters that had Specifications Grading as compared to the semesters without it.

A large majority of the students liked the clarity and the flexibility that it provides to the students as they navigate their coursework. It also challenged motivated students to aim for higher difficult work, while maintaining rigor for the lowest requirement of the assignment.

In the future, we plan to incorporate more elements from Specifications Grading into the weekly quizzes and the final project.

References

[1] B. S. Bloom, "Recent developments in mastery learning," *Educational Psychologist*, vol. 10, no. 2, pp. 53–57, 1973.

[2] V. B. Cohen, "A reexamination of feedback in computer-based instruction: Implications for instructional design," *Educational Technology*, vol. 25, no. 1, pp. 33–37, 1985.

[3] D. Wiliam, *Embedded formative assessment*. Solution tree press, 2011.

[4] L. Benotti, M. C. Martnez, and F. Schapachnik, "A tool for introducing computer science with automatic formative assessment," *IEEE Transactions on Learning Technologies*, vol. 11, no. 2, pp. 179–192, 2017.

[5] J. C. Paiva, J. P. Leal, and Á. Figueira, "Automated assessment in computer science education: A state-of-the-art review," *ACM Transactions on Computing Education (TOCE)*, vol. 22, no. 3, pp. 1–40, 2022.

[6] R. Butler, "Task-involving and ego-involving properties of evaluation: Effects of different feedback conditions on motivational perceptions, interest, and performance.," *Journal of educational psychology*, vol. 79, no. 4, p. 474, 1987.

[7] L. B. Nilson, *Specifications grading: Restoring rigor, motivating students, and saving faculty time*. Stylus Publishing, LLC, 2015.

[8] L. McCarthy, C. Reas, and B. Fry, *Getting started with P5. js: Making interactive graphics in JavaScript and processing*. Maker Media, Inc., 2015.

[9] A. C. Bart, R. Whitcomb, D. Kafura, C. A. Shaffer, and E. Tilevich, "Computing with corgis: Diverse, real-world datasets for introductory computing," *ACM Inroads*, vol. 8, no. 2, pp. 66–72, 2017.

[10] C. Kelleher, "VizHub." http://vizhub.com/, 2022. [Online; accessed 21-May-2023].

[11] C. O. Wilke, *Fundamentals of data visualization: a primer on making informative and compelling figures*. O'Reilly Media, 2019.

[12] Processing Foundation, "P5.js." http://p5js.org/, 2023. [Online; accessed 21-May-2023].

[13] M. Bostock, V. Ogievetsky, and J. Heer, "D^3 data-driven documents," *IEEE transactions on visualization and computer graphics*, vol. 17, no. 12, pp. 2301–2309, 2011.

[14] C. Stolte, D. Tang, and P. Hanrahan, "Polaris: A system for query, analysis, and visualization of multidimensional relational databases," *IEEE Transactions on Visualization and Computer Graphics*, vol. 8, no. 1, pp. 52–65, 2002.

[15] S. Lu, "d3-legend." https://d3-legend.susielu.com/, 2022. [Online; accessed 21-May-2023].

[16] C. Ware, *Information visualization: perception for design*. Morgan Kaufmann, 2019.

[17] S. Lu, “d3-annotation.” https://d3-annotation.susielu.com/, 2022. [Online; accessed 21-May-2023].