

Visualizing Peer Assessment

Yu Suzuki

UC Berkeley, School of Information

Note: This project was conducted with Markus Krause in UC Berkeley,
International Computer Science Institute

Faculty Advisor: Zachary Pardos

Abstract

Peer assessment in coursework assignments is widely used in both university courses and MOOCs, bringing multiple educational benefits to the students as well as allowing the grading process to scale. However, it is not always easy for instructors to manage peer assessment, as the number of reviews created by the students can be enormous.

In this project, I developed visualization tools for instructors to monitor how students are reviewing each other's work. First, I interviewed multiple instructors to understand their needs. Based on the findings from the interviews, I created prototypes and then implemented them as a part of Moocita, an online platform for peer assessment developed by Markus Krause. Finally, I conducted user testing to validate the visualization tools. Based on the feedback I received, I made modifications to improve the clarity of the visualizations.

1. Introduction

1.1 Peer Assessment and Its Benefits

In peer assessment, students grade and comment on each other's work. Peer assessment is widely used in both large-scale online courses like MOOCs (Massively Open Online Courses) and conventional university classes (de Alfaro and Shavlovsky 2016).

The benefits of peer assessment can be divided into two. First, it lowers the grading burden of instructors and allows classes to scale (Kulkarni et al. 2014; de Alfaro and Shavlovsky 2016). This is especially crucial for online courses like MOOCs with enormous numbers of students. Since it is impossible for an instructor to grade thousands of students' submissions, many of such courses use automated quizzes and homework which precludes open-ended questions (Kulkarni et al. 2014). Peer assessment provides an alternative solution to this problem without limiting the assignments to multiple choice questions.

Second, peer assessment brings multiple educational benefits to students. For example, Kulkarni et al. (2014) listed the following benefits:

- Help students to see work from an assessor's perspective
- Create an audience that provides honest feedback and multiple perspectives
- Expose students to solutions, strategies and insights that they otherwise would not likely to see
- Increase student involvement
- Increase student maturity
- Enhance classroom discussion

1.2 Difficulties of Managing Peer Assessment

Despite all the benefits, it is not always easy to monitor how students are reviewing each other's work in peer assessment. This is related with the number of reviews created by students. If there is a class of 50 students and 5 assignments are given, there would be 250 submissions. If each student has to review 3 peers, there would be 750 reviews created by students.

The numbers would be much greater in MOOCs where peer assessment is one of the critical components. If 10,000 students are enrolled and each of them is reviewing 3 peers, there would be 30,000 reviews submitted for each assignment. In such situations, it is impossible for the instructors to know how students are engaging in peer assessment without an effective tool.

1.3 Goal of This Project

Addressing the above issue, the goal of this project is to build visualization tools for instructors to know how peer assessment is taking place. I aim to provide two insights to the instructors through the visualizations. First, they should be able to see how well the students are performing in the assignments. Second, they should be able to see how the students are reviewing each other's work.

2. Moocita

In this project, I developed the visualizations as a part of an existing platform called Moocita. Moocita is an online platform developed by Markus Krause for coursework assignments. Here, I will briefly explain the concept of Moocita and how it works.

2.1 Principles of Moocita

Moocita is based around the concept of apprenticeship. In apprenticeship, an apprentice learns how to do a job through working closely with her master. She would work on real world challenges with constant feedback from her master. The apprentice would also be compensated for her work, which can be monetary or not.

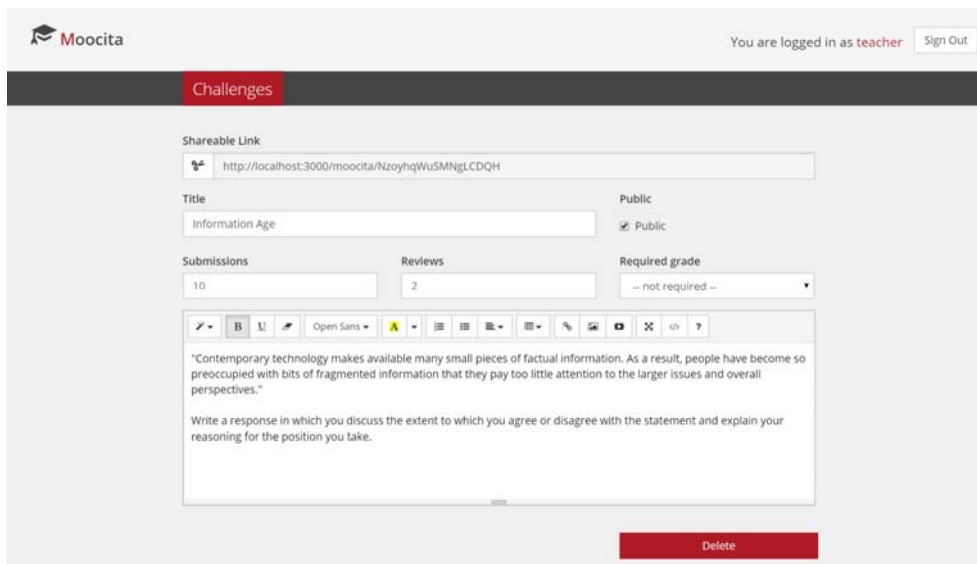
Similarly, assignments on Moocita are connected with real world tasks. The tasks are drawn from online work platforms such as *UpWork* and *witmart*, and students will be paid for completing them. This will provide monetary independence to students, which is especially helpful for people in developing countries who often do not have the financial stability to pursue online education. Income through completing the tasks also provides additional motivation for the students.

Peer assessment is a key component of Moocita. There are two reasons for using peer assessment in Moocita. First, it allows the grading process to scale in online courses, as I explained earlier. Second, it will guarantee the quality of work in the paid tasks.

2.2 How Moocita Works

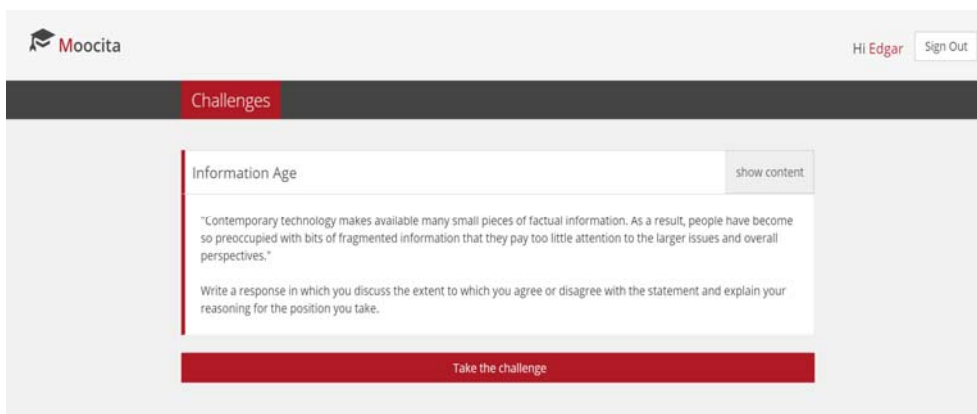
Although Moocita has multiple features, I will only explain the process of peer assessment in this platform.

First, the instructor creates a challenge on the platform. She would then share the link to the challenge with her students.



The screenshot shows the Moocita 'Challenges' page. At the top, the Moocita logo is on the left, and 'You are logged in as teacher' with a 'Sign Out' button is on the right. A red 'Challenges' header is below the navigation bar. The main form includes a 'Shareable Link' field with the URL 'http://localhost:3000/moocita/NzoyhqWuSMNgLCDQH'. Below this is a 'Title' field with 'Information Age'. To the right of the title is a 'Public' checkbox, which is checked. Further right is a 'Required grade' dropdown menu set to 'not required'. Below the title field are two input fields: 'Submissions' with the value '10' and 'Reviews' with the value '2'. Below these is a rich text editor with a toolbar and a text area containing a paragraph about contemporary technology and a prompt to write a response. At the bottom right of the form is a red 'Delete' button.

Once the instructor sends the link to the students, they can take the challenge by accessing the link.



The screenshot shows the Moocita 'Challenges' page from a student's perspective. At the top, the Moocita logo is on the left, and 'Hi Edgar' with a 'Sign Out' button is on the right. A red 'Challenges' header is below the navigation bar. The main content area shows a challenge titled 'Information Age' with a 'show content' button. Below the title is a text area containing a paragraph about contemporary technology and a prompt to write a response. At the bottom of the challenge area is a red 'Take the challenge' button.

They would enter their responses to the textbox and submit it.

Students can also see grades and comments they received from other students. Again, they cannot see who has written the review but the instructor can. Students will be asked to give feedback to the reviews about how useful they were. To avoid bias in the feedback, students can see the grades they received only after submitting their feedback to the comments in the review.

3. User Interview

To understand the needs of instructors, I interviewed multiple people who are currently teaching or have teaching experience in university courses. I also interviewed a graduate student who has worked on a similar topic in the past.

3.1 Participants

Below is the list of people I interviewed:

Professor A – teaching a computer science course in Northeastern Illinois University

Professor B – used to teach English writing in UC Berkeley

Professor C – teaching information visualization in UC Berkeley

Graduate Student D – conducted research in the past on how to present information about courses to instructors teaching MOOCs

3.2 Findings 1 – Benefits of Peer Assessment

From the interviews, I found that there are multiple benefits in peer assessment and instructors use it for different reasons.

Professor C regards peer assessment as a way of active learning. Students will get engaged in the coursework more through grading and commenting on other students' work. She also pointed out that peer assessment can strengthen bond between students. From the learning perspective, *Professor C* uses peer assessment to expose students to other students' solutions. In information visualization she teaches, there are often multiple possible ways to visualize a given dataset.

Professor A uses peer assessment for coding assignments. He uses it as a part of the grading process. In contrast, *Professor B* regards it as an intermediate process to improve students' writings. In her class, students do not grade each other's work – they

only give comments. Then, they would make modifications to their work based on their peers' comments. Their grades are determined by their final submissions graded by the instructor.

Professor B wants to see the difference of students' writings between before and after peer assessment. She also wants to know what comments are useful for improving their work.

3.3 Findings 2 – Bias in Grading

All of the interviewees were aware of the grading bias in peer assessment. *Professor C* told me that instructors have to train students to grade other students' work accurately. We should note that there is only a weak correlation between review accuracy and student proficiency (de Alfaro and Shavlovsky 2016). This means, high performing students do not necessarily be good graders.

One way to deal with this issue is to create a rubric with descriptions of grading criteria and standards. This will help to decrease discrepancies between students and the instructor, and among students. However, she mentioned that students sometimes do not understand the rubrics. Another way suggested by her to improve the quality of the comments is to give students examples of useful comments prior to the peer assessment.

There seem to be some tendencies in grades assigned by students. *Professor A* pointed out that grades given by peers tend to be too high, as students do not want to be mean to others. On the other hand, *Professor B* mentioned that students tend to be too harsh to themselves in self-assessing their writings. She explained that while the instructor takes the context of the writings into consideration, students tend to focus too much on the surface quality of their writings.

Another issue is the discrepancies between students. *Graduate Student D* suggested that one way to deal with this is to have students' submissions graded by two peers, and bring in a third peer if the difference between the first and second peer is greater than a threshold. On the other hand, while *Professor B* admits that disparate feedback may confuse students, she also thinks it as a learning opportunity. Students have to think deeply to synthesize the different comments they get to reflect them to their final submissions.

In many platforms, students can give feedback to the reviews they received based on how useful they are. However, *Professor C* pointed out that students tend to rate the comments higher when they received high grades. Such bias is also reported by de Alfaro and Shavlovsky (2016). As I already mentioned, Moocita addresses this issue by showing the grades only after students have submitted their feedback.

3.4 Findings 3 – Requirements for Visualizations

Graduate Student D argued that visualizations for teachers must be simple. Complex visualizations such as scatter plots need training to interpret. We should be aware that instructors in non-STEM fields may not be familiar with such diagrams.

Professor B argued that dashboards should be actionable. That is, the instructors should be able to know what actions they have to take in order to improve their courses from the visualizations.

The interviewees told me multiple possible actions an instructor can take. If a certain student seems to be struggling, she can speak individually to that student or send an email to him. If many students have performed poorly on a certain assignment, she can cover the topic again in the class. Also, there might be a problem in the question. In this case, she may check the wording of the question to avoid misunderstandings. Finally, she can make modifications to her course offered next year.

Graduate Student D pointed out that instructors value different types of information in conventional university classes and MOOCs. As instructors cannot monitor enormous numbers of individual students, visualizations for MOOCs should be focused on aggregate information and overall trends.

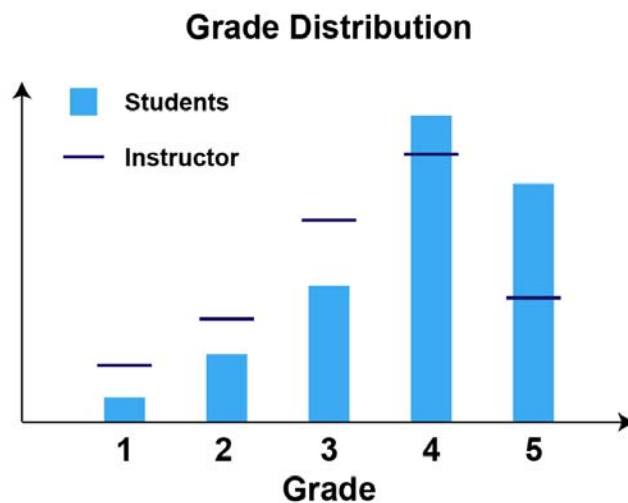
3.5 Summary

From the interviews, I found that peer assessment can be used in many different fields, and instructors adopt it for various reasons. Although peer assessment brings various benefits to instructors and students, all of the interviewees were concerned about the grading bias. There are some known tendencies, such as students overinflating the grades of their peers. From the interviews, I also understood that visualizations for instructors should be simple and actionable.

4. Prototypes

Based on the findings from the user interviews, I created prototypes for the visualizations. Because the grading bias of students was the greatest concern of the interviewees, I decided to develop visualizations focusing on this issue. I created three prototypes of visualizations, each of them summarizing the data from different perspectives.

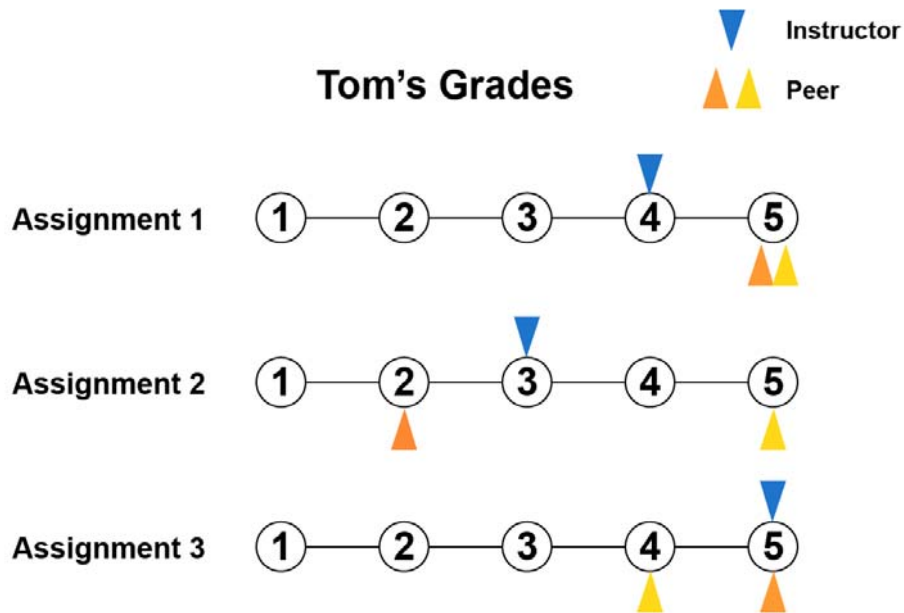
4.1 Visualization for Each Challenge



In this visualization, the grade distribution for each challenge is shown as a histogram. It provides two aspects of information to the instructor. First, the distribution represents how well the entire class has performed in the challenge. From this graph, the instructor can easily see whether the students understood the topic. Second, the instructor can see the grading bias of the students through comparing the grade distributions by the instructor and the students.

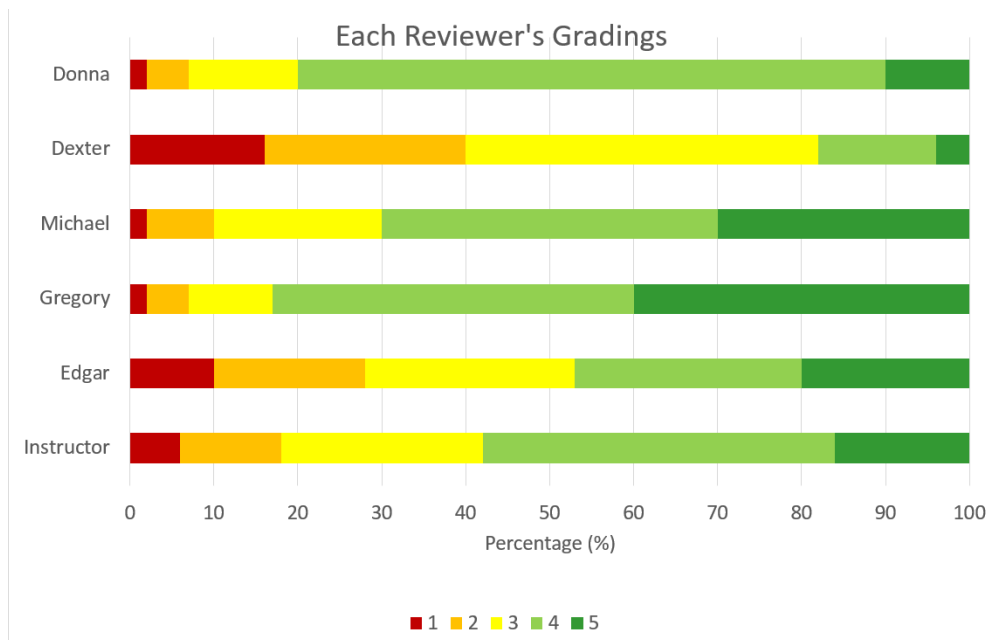
In the prototype, grades given by the students are represented as a bar graph and grades given by the instructor are represented as markers overlapping the bar graph. This treats the instructor's distribution as the standard and emphasizes the difference of the students' distribution. However, since I used an off-the-shelf JavaScript library to implement the visualization, my final visualization represents the two distributions as two bar graphs with different colors. I will explain about this in the later section.

4.2 Visualization for Each Student



This visualization shows the grades a certain student has received during the course. The instructor can get two insights from this visualization. First, it shows how well a certain student is performing. While the visualization of the grade distribution discussed above focuses on the performance of the entire class, this focuses on individual students. Second, it shows how well the grades given by the instructor and the students match. The instructor can immediately know if there is a serious disagreement in a certain submission. For example, there is a great difference in grades assigned by two peers in “Assignment 2” in the prototype.

4.3 Grades Given by Each Student



This visualization is specialized to show the reviewers' bias in peer assessment, representing what grades each student has given to other students during the course as a stacked bar graph. The grades given by the instructor are also shown. The instructor can point out which student is being too lenient or too harsh to other students. In the prototype, Dexter is giving too many low grades and Gregory is giving too many high grades. Also, Donna has a problem too, as too many of the grades she gives are “4.”

5. Technologies Used to Develop the Visualizations

Based on the prototypes, I implemented the visualizations for instructors on Moocita. Here, I will explain what framework, tools and libraries I used for the implementation.

5.1 Meteor

Moocita is developed with a JavaScript framework Meteor that runs on Node.js. Meteor is suitable for rapid prototyping for multiple reasons. First, the front-end and the database are seamlessly integrated in Meteor and the front-end can directly access the database. This speeds up the development process as we do not have to write the back-end code to bridge the front-end and the database.

Meteor also makes the front-end development easier by its template engine. We can embed dynamic data into web pages and the pages are automatically updated when the data changes. Finally, Meteor produces cross-platform code supporting PC browsers, Android and iOS. This frees us from worrying about compatibility across platforms.

5.2 MongoDB

Meteor uses MongoDB as the database. MongoDB is a document-oriented database in which data is stored as JSON objects. In this database, queries are also represented as JSON object. Below is an example of a JSON document stored in the database and a query to retrieve the matching document. The JSON-centered architecture matches well with JavaScript frameworks like Meteor, as we can easily generate queries and use the data retrieved from the database.

```
{
  "_id" : "eU6B8i014dZ1bq8gk6",
  "challenge_id" : "Vdf05KXSrujPySAv7",
  "content" : "Today, the abundance of information is in
essence creating a haze in the minds of people leaving them
more confused as they gather different versions of the same
information. Buying a television or a camera has become a
tough ordeal as every manufacturer tries to create an
impression of superiority.",
  "owner_id" : "y5GQMZ8FTN9Dxc38P",
  "submission_owner" : "Dexter Sheriff",
  "public" : true,
  "createdAt" : "2016-04-15 18:09:56.038Z",
  "updatedAt" : "2016-04-15 18:09:56.038Z"
}
```

Sample of JSON document stored in MongoDB

```
db.getCollection('submissions').find(
  {"challenge_id": " Vdf05KXSrujPySAv7",
   "owner_id": "y5GQMZ8FTN9Dxc38P"}
)
```

Sample of query retrieving documents

MongoDB is categorized as a NoSQL database, as it has a completely different architecture from conventional relational database which stores data as tables. The greatest difference is that it does not need database schemas, which define what fields are

in tables and the data format of the fields, like SQL databases. This has a great advantage in agile development, as we can add fields or modify the data format easily.

5.3 CoffeeScript

CoffeeScript is a language that compiles into JavaScript. It makes JavaScript code shorter and more readable. In this project, all the code is written in CoffeeScript. Below is an example of JavaScript code and the same code written in CoffeeScript.

```
var current_submission = function() {  
  var challenge = current_challenge();  
  
  if (!challenge) {  
    return null;  
  }  
  
  var filter = { challenge_id: challenge._id };  
  
  var submission = findOne(filter);  
  return submission;  
}
```

Sample code written in JavaScript

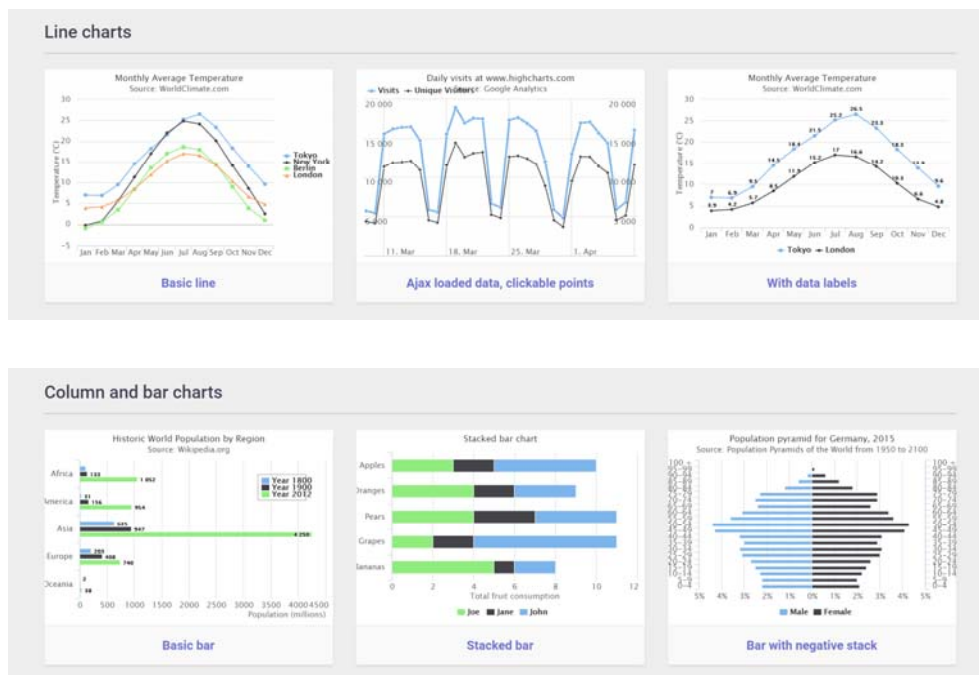
```
current_submission = () ->  
  challenge = current_challenge()  
  
  if !challenge  
    return null  
  
  filter =  
    challenge_id: challenge._id  
  
  submission = Submissions.findOne(filter)  
  return submission
```

Sample code written in CoffeeScript

5.4 Highcharts

Highcharts is a JavaScript library that renders interactive charts. It supports many types of graphs, such as bar graph, line graph, stacked bar graph, pie chart and scatter plot. I implemented the visualizations for grade distributions and how each student is grading others using Highcharts.

D3.js was another available option for data visualization library. Compared with Highcharts, graphs in D3.js are highly customizable and it is possible to create more complex visualizations. However, since I only needed basic visualizations and developing visualizations in D3.js can be time-consuming, I decided to use Highcharts for this project.

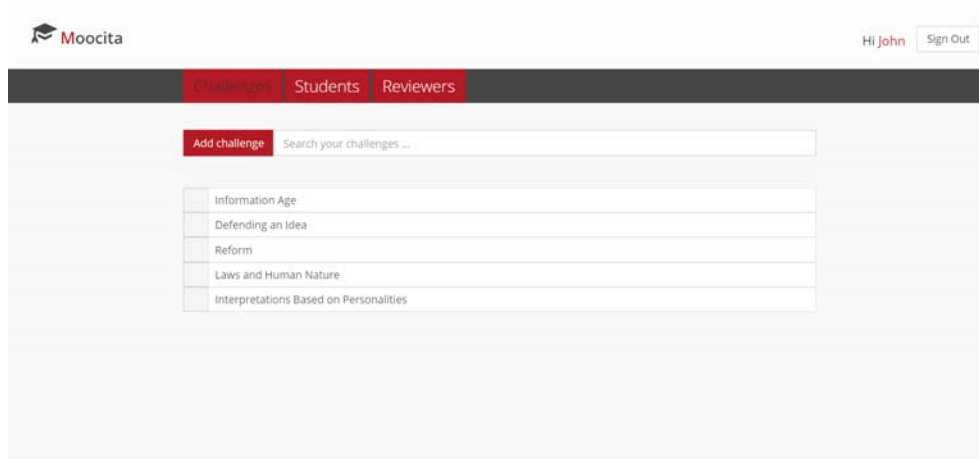


Examples of visualizations by Highcharts
(Figures from <http://www.highcharts.com/demo>)

6. Visualizations Developed

6.1 Visualization for Each Challenge

From the list of challenges page, the instructor can go to the pages showing the data about each challenge.



List of challenges

The screenshot shows the detail page for the 'Information Age' challenge. The challenge description is: "Contemporary technology makes available many small pieces of factual information. As a result, people have become so preoccupied with bits of fragmented information that they pay too little attention to the larger issues and overall perspectives." The challenge settings are: Public: Yes, Minimum Submissions: 10, Required Reviews: 2, Required Grade: 4. A grade distribution bar chart shows the percentage of submissions for each grade (1-5) for both the instructor and students. The grade table shows the grades assigned by the instructor and three peers.

Grade Distribution

Grade	Instructor	Students
1	10%	0%
2	10%	0%
3	10%	20%
4	50%	40%
5	20%	40%

Grade Table

Name	Instructor	Peer 1	Peer 2	Peer 3
Edgar Resendez	4	4	4	4
Arturo Miller	4	4	4	5
Gregory McGee	4	5	3	5
Yong Lu	1	5	4	3
Michael Gardner	4	5	4	3

Detail page for each challenge

The question and the settings of the challenge are shown in the upper part. The instructor can modify the challenge by clicking the “Edit Challenge” button.

In the middle of the page, the visualization of grade distribution is shown. The distribution by the instructor and the students are shown in different colors. As students would usually review more than one peer, I normalized the number by calculating the percentage of the students getting each grade.

Below the visualization, a list of students and the grades they received are shown as a table. The bar graph visualizes the numbers in this table.

The instructor can see the detail of each student's submission and the reviews he received by clicking the students' names. Although the reviewers are anonymous to the students, the instructor can see who reviewed whom.

Moocita

Hi John Sign Out

Challenges Students Reviewers

Submission Detail

[Back to "Information Age"](#)

Challenge Content

"Contemporary technology makes available many small pieces of factual information. As a result, people have become so preoccupied with bits of fragmented information that they pay too little attention to the larger issues and overall perspectives." Write a response in which you discuss the extent to which you agree or disagree with the statement and explain your reasoning for the position you take.

Submission by Gregory McGee

An duo rebum atqui legere, impedit denique tincidunt duo an. Eu hinc inimicus sed. Ius ad erat persequeris, eum habeo solum accusata ut. Ex est vide nullam delectus, vix esse ridens nostrud id. Pri eripuit veritus et, eum ipsum ancillae concludaturque te. Qui fugit harum vituperatoribus ad, quas nulla viderer ut sea. Ignota disputationi sea ei, an per audiam signiferumque. Feugiat habemus facilisis nec no. Nec ex decore contentiones definitionem. Pri eu altera quidam graecis. Regione denique no eos, ignota labores epicuri ne quo. Vim ea putent suscipiantur. An quidam impetus consulatu duo. An numquam delicatissimi cum, vis nisi maluisse ut. Dicit maiorum verterem nam te.

Review by the Instructor (Grade: 4)

Ignota accusant oportere ne pro, ea mea suas meliore. Vix sonet vidisse vocibus ut, ea debet postulant sea, mei ex vocibus dissentias scribentur.

Review by April Westmoreland (Grade: 5)

Ius ad erat persequeris, eum habeo solum accusata ut. Ex est vide nullam delectus, vix esse ridens nostrud id.

Review by Edgar Resendez (Grade: 3)

Laudem possim te nec, ex vel perfecto erroribus. An duo rebum atqui legere, impedit denique tincidunt duo an. Eu hinc inimicus sed. Ius ad erat persequeris, eum habeo solum accusata ut.

Review by Zhi He (Grade: 5)

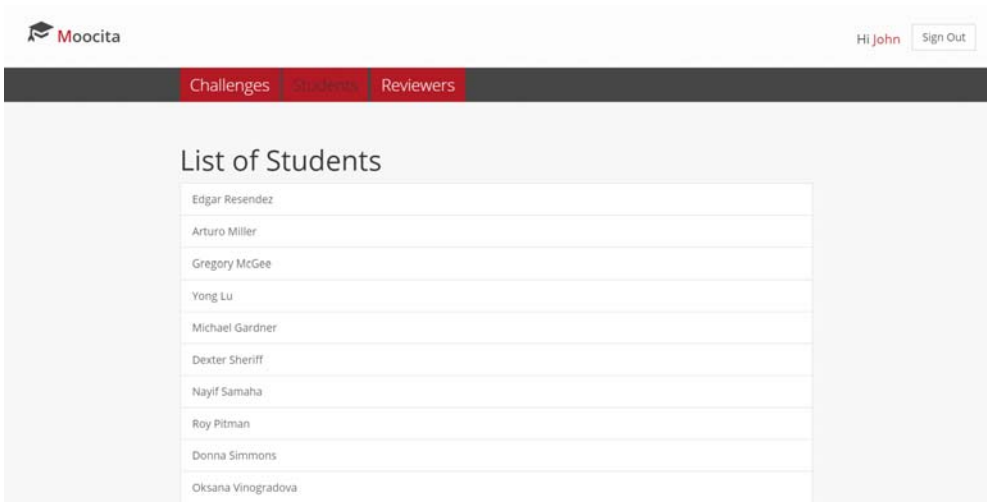
Pri eripuit veritus et, eum ipsum ancillae concludaturque te. Qui fugit harum vituperatoribus ad, quas nulla viderer ut sea. Ignota disputationi sea ei, an per audiam signiferumque. Feugiat habemus facilisis nec no. Nec ex decore contentiones definitionem. Pri eu altera quidam graecis. Regione denique no eos, ignota labores epicuri ne quo. Vim ea putent suscipiantur. An quidam impetus consulatu duo. An numquam delicatissimi cum, vis nisi maluisse ut.

Detail page for each submission
(Dummy text is shown as placeholders)

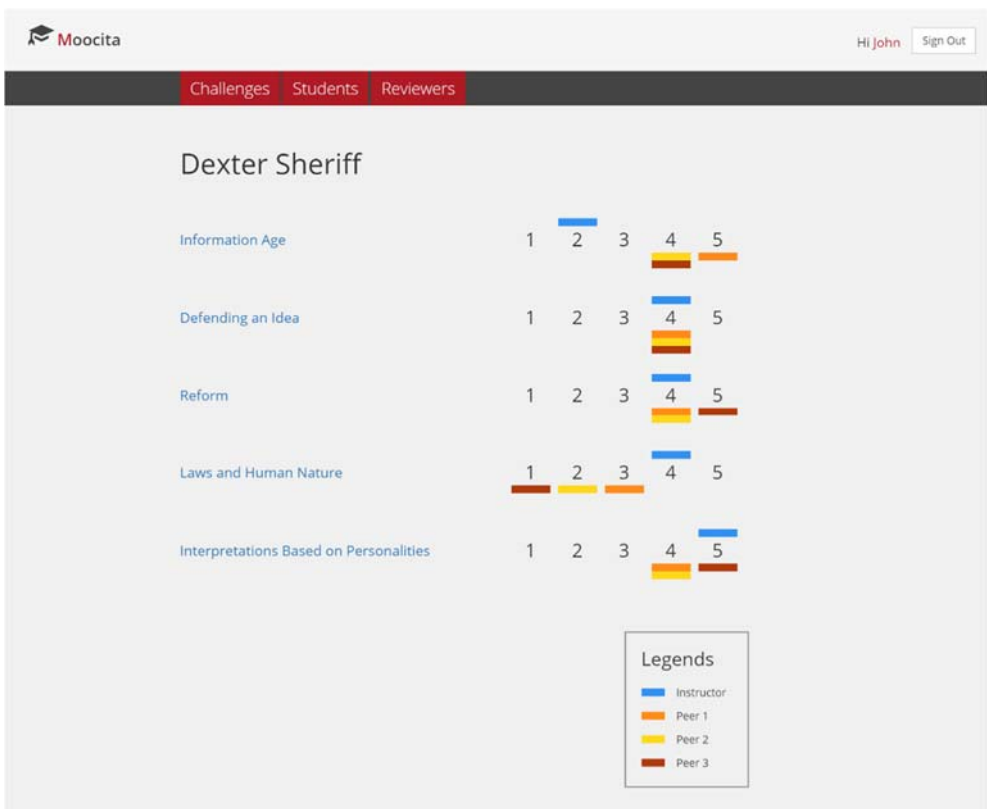
6.2 Visualization for Each Student

From the list of students page, the instructor can go to the pages showing the grades each student has received for each challenge. As it could not be implemented using the

existing libraries, the developed visualization looks significantly different from the prototype due to technical difficulties.



List of students



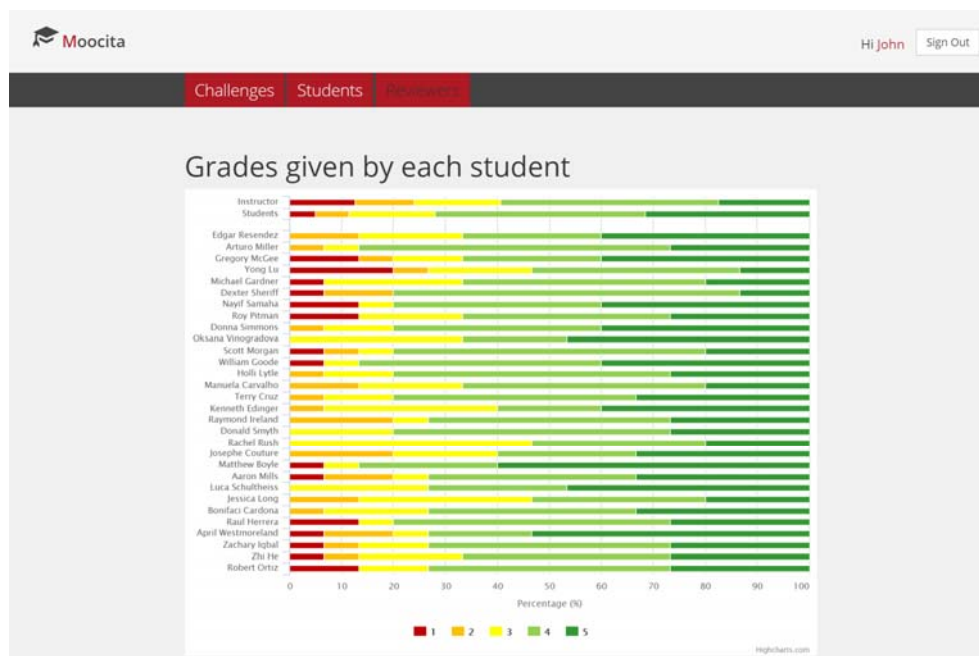
Detail page for each student

The grades given by the instructor are shown as blue markers above the grade numbers. The grades given by the peers are shown as markers in different colors below the numbers.

Similar to the detail page for each challenge, the instructor can see the detail of the submissions by clicking the challenge titles.

6.3 Grades Given by Each Student

In this page, grades given by each student are shown as a stacked bar chart. The aggregate proportion of grades given by all the students is shown in the top, along with the proportion by the instructor. In this way, the instructor can see the overall tendency of the students, as well as the tendency of each student. For example, we can see that students are giving high grades more frequently than the instructor in the sample visualization.



Page showing the grades given by each student

7. User Testing

To validate the visualizations I developed, I conducted user testing with *Professor A* and two graduate students. *Professor A* is the same person as the user interview.

7.1 Methods

Graduate Student X is teaching an online class as a GSI in two sections with approximately 36 students altogether. He created a challenge on Moocita as one of the coursework assignments and asked the students to review their peers. Later, I interviewed

him about how the students interacted with the system and how he thinks about the visualizations for instructors.

Graduate Student Y is also teaching a class as a GSI. Instead of using Moocita in real class, I used a mock dataset for the testing with him. I asked him to freely explore the visualizations and provide feedback. I did similar testing with *Professor A* too.

7.2 Visualized Data

In *Graduate Student X*'s class, students formed teams of around three members for their final projects. For this user testing, each team submitted its intermediately work and then members of the teams reviewed other teams' work. 13 submissions and 30 reviews were posted for this challenge.

Some visualizations were not fully functioning due to the lack of data. For example, Moocita does not have an interface for the instructor to grade the students' submissions at this moment, so the grades given by the instructor were absent. Also, since there was only one challenge in his testing, the visualization for each student's performance over the course was not meaningful.

For the testing with *Graduate Student Y* and *Professor A*, I used a mock dataset with 5 challenges, 30 students and 3 peer reviews per submission. This adds up to 150 submissions and 450 peer reviews. The mock dataset also includes grades given by the instructor. Grades were randomly generated based on distributions separately defined for the instructor and the students, and for each challenge.

7.3 Visualization Feedback

Although only some of the visualizations were functioning due to the lack of data, *Graduate Student X* commented that the meanings of the visualizations were clear for him.

Graduate Student Y and *Professor A* could also interpret the meanings of the visualizations, but they pointed out that some of the attributes were not clear to them.

For the visualization of the grade distribution in each challenge, *Graduate Student Y* could successfully interpret the difference of the grading tendencies between the instructor and the students. However, he pointed out that what the axes mean is not clear.

The y-axis is labeled “Percentage (%)” which stands for the percentage of the students. He was initially confusing this with the grade percentage.

Graduate Student Y commented that it was not the most intuitive diagram. He said that the legends would be absolutely necessary to interpret the visualization. *Professor A* was confused in the beginning whether the grades shown are given to or given from the student. He suggested to make it clear that the visualization is showing the grade a student has received in the page title, and change the title of the legends from “Legends” to “Evaluators.” He also mentioned that he wanted to see the average of the grades given by the peers for each challenge.

For the visualization of grades given by each student, *Graduate Student Y* could see that students tend to give higher grades than the instructor. He could also see who was giving too many high grades or low grades. However, he thinks the percentage shown in the graph was not clear, as he did not know what the percentage is out of. When I explained that the percentage is out of all the reviews an individual student has submitted, he pointed out that it might be a problematic way to normalize the numbers as some students may submit fewer reviews than others. He said that it is not appropriate to compare two students with the same scale when one has submitted 15 reviews and the other has submitted only 3. He also pointed out that it is strange to show the graph reaching 100% while the students are still reviewing other students’ work during the reviewing period.

Professor A was confused in a different way initially. Again, it was not clear for him whether the stacked bar graph is showing the grades given by or given to each student. Also, he confused the grade numbers one to five with challenge numbers, as there were coincidentally five challenges in the mock dataset. Nevertheless, once he understood the meaning, he commented that this visualization is useful.

7.4 Platform Feedback

When *Graduate Student X* rolled out Moocita in his class, several problems were reported to me. Some of them are related to the bugs in the system. For example, submissions for different challenges appeared for reviews. I also found some missing values in the database, which implies that some of the submissions were not stored properly.

However, many of the problems reported by the students seem to come from their confusion with the system. For example, some students were confused with the “save” or “publish” button which appears only when edits are made to the textbox. In addition, *Graduate Student X* pointed out that students can get lost in the pages as they cannot use the browser’s “back” or “forward” button. This is related to Moocita being developed as a single page application. He also suggested that students cannot see where they are in the entire process of peer reviewing.

He also mentioned that it is not clear when students can start reviewing others. By design, students cannot start peer reviewing when there are not enough submissions yet. Multiple students reported that they could not review other students’ submissions, but I am not sure whether this is due to technical problems or the number of total submissions.

7.5 Summary

The participants could interpret the visualizations, but I realized that there are some unclear points. Both Professor A and Graduate Student Y commented that I have to make the legends and the axes of the graphs clearer. Although I assumed that the meanings of the visualizations were obvious, I found that I need more explanations.

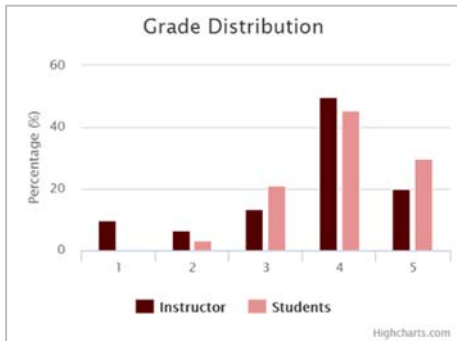
In addition, I found some issues related to the entire Moocita platform. While some of them are technical, many of them are related with the user interface. *Graduate Student X* pointed out that students can get lost in the process.

8. Modifications Based on the User Testing

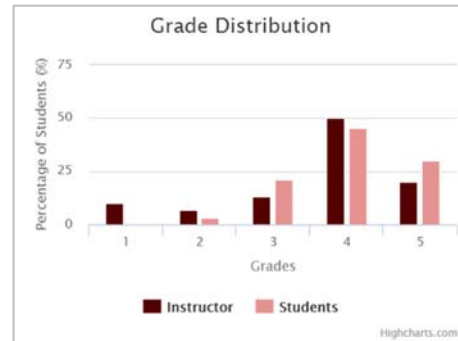
From the user testing, I found that there are some unclear points about the visualizations. Some of the issues could be solved easily by making small modifications. I will explain the minor modifications I made based on the feedback from the user testing.

8.1 Visualization for Each Challenge

Graduate Student Y confused the percentage in the y-axis with grades. To address this issue, I added a label “Grades” to the x-axis. Also, I changed the label of the y-axis from “Percentage (%)” to “Percentage of Students” so that the meaning of the numbers is clear.



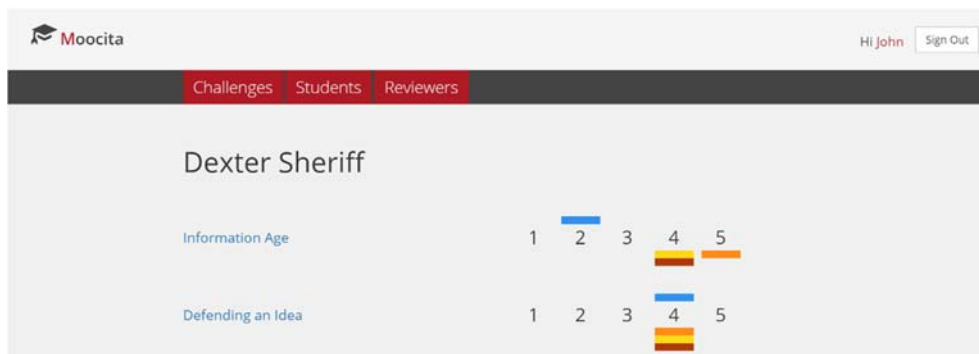
Bar graph (before)



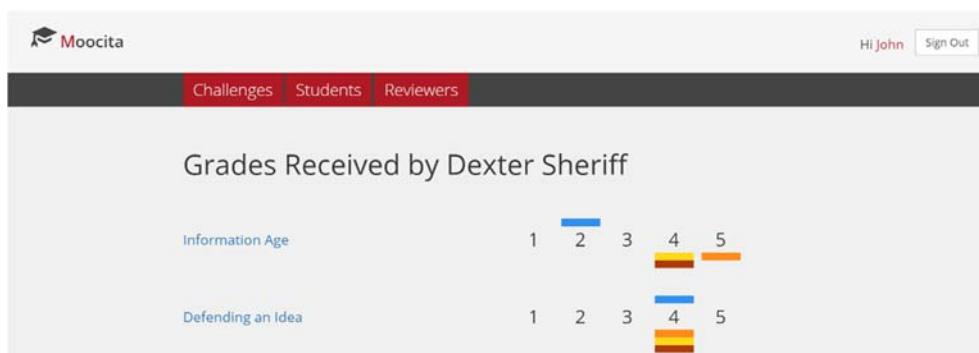
Bar graph (after)

8.2 Visualization for Each Student

In the beginning, *Professor A* was not sure whether the grades shown were the grades given to or given from the student. To make this clear, I changed the page title and the legends title.



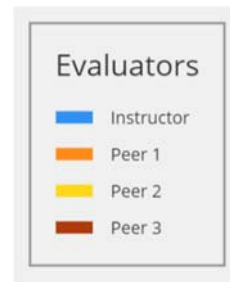
Page title (before)



Page title (after)



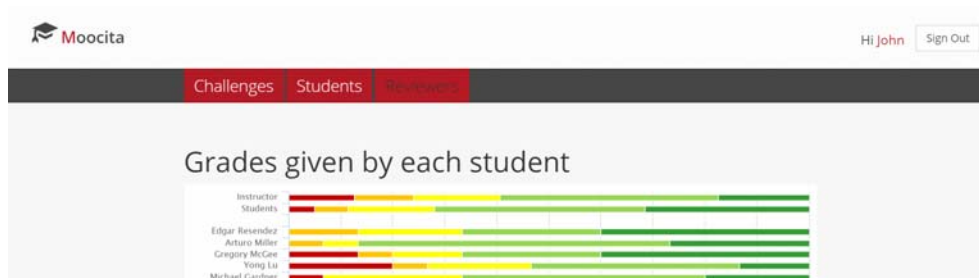
Legends (before)



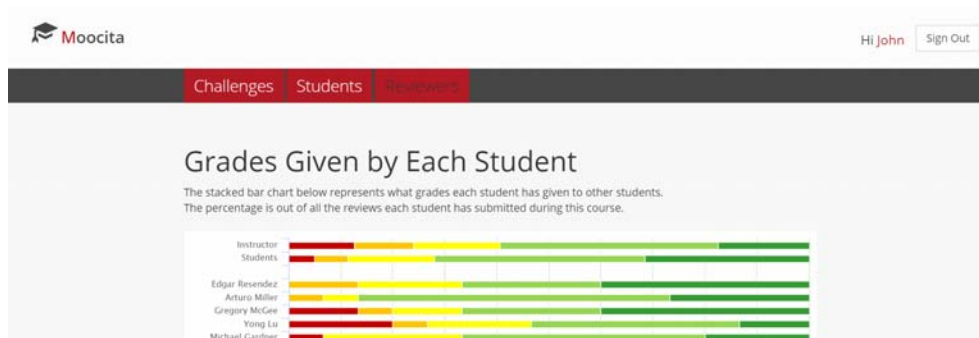
Legends (after)

8.3 Grades Given by Each Student

Both *Professor A* and *Graduate Student Y* were confused about what the percentage meant. I added some explanations above the visualization.

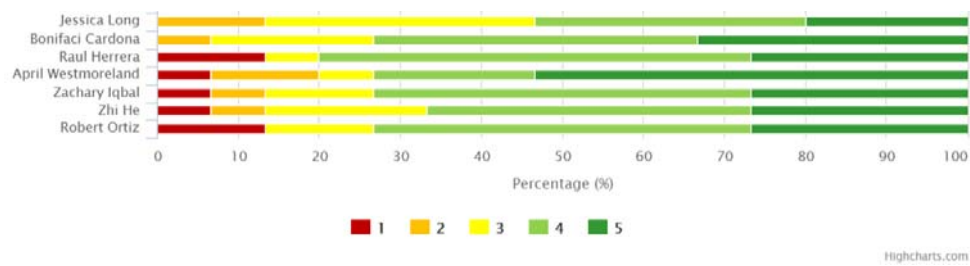


Visualization of grades given by each student (before)

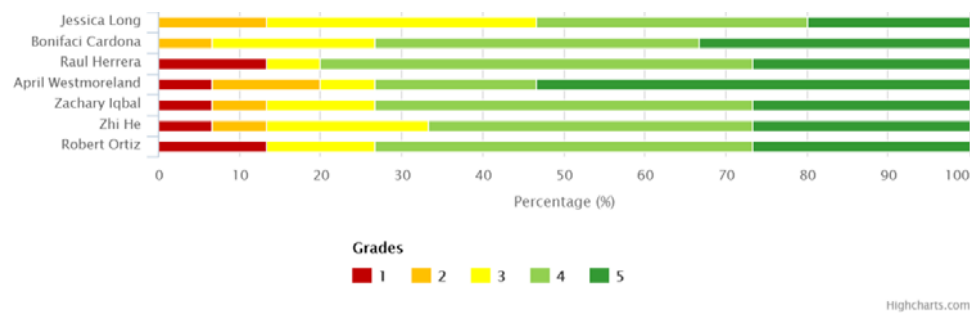


Visualization of grades given by each student (after)

Professor A initially thought the numbers one to five correspond to the challenges. To avoid such confusion, I added a label “Grades” to the legend.



Legends (before)



Legends (after)

9. Summary and Future Work

Although peer assessment in coursework assignments has multiple benefits, it is not always easy for instructors to monitor how students are reviewing each other's work. This project aims to address this problem by developing visualization tools for instructors.

To understand the needs of instructors, I first conducted user interviews. I found that instructors are concerned with the grading bias of students. Interviewees also suggested that visualization tools should be simple and actionable. Based on these findings, I decided to develop visualizations to show the grading bias. I developed three prototypes focusing on each challenge, each student and grades given by each student. I then implemented the visualizations using various JavaScript frameworks and libraries. The visualizations are built as a part of Moocita, which is an online platform for peer assessment.

To validate the visualization tools I developed, I conducted user testing with a professor and graduate students. I found that although instructors could interpret the meanings of the visualizations, there were some unclear points. Some of the issues could be easily

dealt with by making small adjustments, so I made minor modifications to the visualizations by adding explanations.

Other problems need more time to solve. For example, there are some bugs in the peer reviewing feature of Moocita. Moreover, some aspects of the user interface caused confusion by the students. It is our future work to improve the usability of Moocita. We would also need to conduct user tests with a larger scale in real classes to further validate our product.

References

- de Alfaro, L., & Shavlovsky, M. (2016). Dynamics of Peer Grading: An Empirical Study. *Technical Report UCSC-SOE-16-04*.
- Kulkarni, C., Wei, K. P., Le, H., Chia, D., Papadopoulos, K., Cheng, J., Koller, D., Klemmer, S. R. (2014). Peer and Self Assessment in Massive Online Classes. *Understanding Innovation Design Thinking Research*, 131-168.