Developing a system to increase motivation and engagement in student code peer review

Mireilla Bikanga Ada
School of Computing Science
The University of Glasgow
Glasgow, United Kingdom
https://orcid.org/0000-0001-5406-6935

Mohammad U. Majid
School of Computing Science
The University of Glasgow
Glasgow, United Kingdom
mohammadmajid581@gmail.com

***Abstract*—This paper describes a system that aims to provide course instructors with the capability to create and conduct formative and summative peer code review exercises for their students. It also aims to improve student engagement with the review process and motivation to achieve better learning. It enables students to rate the reviews they receive, thus providing feedback to the reviewer. The system awards badges to students for completing reviews and high-quality reviews. The preliminary user evaluation of the prototype was very positive.**

Keywords—peer review, programming, coding, badges, higher education, summative assessment, formative assessment

I. Introduction

Peer review is “a context-dependent, relational concept” which involves evaluative activities and is increasingly being used in Academia [1] and has seen increased calls for further research [2]. Peer code review (PCR), also known as code review, is an established practice in software development and is used to share knowledge, ensure high-quality code, and reduce code defects. This subsequently leads to more readable and understandable code [3]. However, PCR is not a standard practice in higher education but is receiving considerable attention as there are many learning benefits from student participation in the process [4, 5]. Despite these benefits, there are two key issues with the PCR process: low code review quality and lack of student motivation. Student engagement with reviews and student motivation plays an important role in determining how successful the PCR process is.

Reference [6] conducted a peer review study with 278 higher education students to examine student perceptions before and after experiencing peer review. The study showed that some students were concerned about the variation in review quality, especially when they put in great effort and received little in return. As a result, it revealed a degree of dissatisfaction amongst students and decreased learning in the peer-review process. Reference [7] implemented a peer code review process with 9 level two students over a period of two academic years. They analysed student interviews to find that students rushed through code reviews and did not put in sufficient effort to review their peer’s code. Reference [8] conducted a peer code review assessment with 87 level one students. They observed that students were submitting their reviews at the last minute with superficial or meaningless comments. Low code review quality can result in dissatisfaction with the peer review process, as it impacts the student’s learning experience for both the reviewer and code author. There is, therefore, a need to improve student engagement with the review process to ensure higher quality reviews.

On the other hand, [9] carried out a peer code review study with 134 students from two universities. Over one semester, students completed an attitudinal survey before and after four assessed PCR assignments. They noted that students showed a lack of motivation, which resulted in them missing out on a valuable learning opportunity. In addition, [10] identified a lack of motivation as a barrier to successful PCR in education. Low motivation can lead to students avoiding or delaying their PCR tasks. It can also affect student engagement with the review process. So, there is a need to improve student motivation to ensure students learn as much as possible from the PCR process. This paper presents a Peer Code Review system that could improve student engagement with the review process. The rest of the paper is divided as follows: Section II provides the background, including a brief overview of similar applications; Section III is concerned with the framework used to develop the system; Section IV presents the user interface; Section V presents a preliminary user evaluation, and the last section provides a brief conclusion and future work.

II. Background

A. Importance of multiple student reviews per assignment

Peer feedback provides benefits to both the reviewer and the author. The quality of feedback a student receives can impact their learning experience. Reference [11] and [12] show that peers can provide feedback of equal value to the feedback given by course instructors. Reference [13] found that aggregating multiple student reviews can improve overall review accuracy. Receiving more reviews is more likely to ensure students get the feedback they need. Reference [14] suggest having at least 2-3 reviewers per assignment. Meanwhile, [15] found that obtaining five reviews for each student submission increased review accuracy and created grades similar to that of teacher assistant reviews.

B. *Anonymity in Peer Review*

The peer-review process can reveal the identity of the code's author, the reviewer, or both. Hiding the reviewer's identity allows peers to deliver more critical feedback [16]. Reference [17] found that anonymity improved participation and reduced biases in student review tasks. These studies demonstrate that anonymity improved the peer-review process.

C. *Rubrics for fairness and accuracy*

During the peer review process, an evaluation rubric is used by students to review their peer's work. The course instructor defines the evaluation rubric for an assignment, consisting of a series of questions that provide structure to peer reviews and represent the assignment's assessment criteria. Peer review guided by a rubric has more potential for learning as rubrics make assessment expectations and criteria explicit [15]. Students can also use a rubric to mark their peer's work. Reference [18] found that peer assessment with a rubric was more valid and reliable. A good peer code review must be fair and accurate in terms of review quality. Reference [19] conducted four code review studies with level one university students. They used their findings to establish a set of ground rules for PCR. One rule states that reviews should identify strengths to provide the student with positive reinforcement. Another rule mentions that reviews should focus on improvements, ideally phrased in positive, action-oriented terms to be well-received. A good quality code review in higher education must identify strengths and weaknesses, provide actionable improvements, and be fair and accurate.

D. *Student motivation*

Game-based learning can enhance learning and motivation through the introduction of game elements [20]. Gamification has been used to improve student peer code review [5]. Reward-based gamification can be particularly effective in teaching skills with real-world value [21]. Once students understand the value, rewards are no longer needed. However, rewards must be small since extrinsic rewards can undermine intrinsic motivation. In higher education, the introduction of a points or badge-based incentive to peer code review has the potential to improve and maintain motivation.

E. *Existing Applications*

Peergrade [22] is an all-in-one web-based peer review platform used by over 8,000 institutions with over 100,000 monthly active users. It supports peer review of all types of work. It enables instructors to use peer review with students using an instructor-defined rubric. Firstly, instructors create an assignment and an evaluation rubric. Students upload their work to Peergrade, and after the hand-in deadline, work is distributed amongst peers for review. Students complete peer reviews, and after the deadline, feedback is returned to students. Students can 'flag feedback' by reporting to the teacher what they do not like about it, and can rate the feedback they receive. Peergrade provides a vast array of features for students and instructors. However, it does not provide support for revised final submissions and summative assignments. It is also costly for institutions. Peergrade plans charge per student per year. Lastly, it is limiting for institutions as they cannot customise or modify the system.

Aropa [23] is a web-based peer review platform developed by Hamer and Purchase. The peer-review process is similar to Peergrade: Peers submit their work and complete reviews. The platform supports instructor-defined rubrics, anonymity, and summative assignments. Aropa supports fifteen different assignment configuration options. Reference [24] analysed the different assignment configurations in over one thousand Aropa peer-review assignments and found that peer review assignments are set up in very different ways. Students can provide qualitative and quantitative feedback to their peers. Reviewer allocations can be made manually or automatically with options for adjustments. There are options to utilise tutors in parts of the review process and award marks for submission quality, where quality is determined through peer assessment via the rubric. There is also an option to award marks for completing reviews and review quality, where quality is marked manually via tutors or instructors. Aropa has been free for educational institutions since 2009. The list of features above highlights a subset of Aropa's features only. One of the main limitations is that Aropa is built for general peer review. Therefore, to utilise Aropa for peer code review, students must place their code into a file and upload the file to Aropa. In addition, Aropa does not attempt to improve student engagement or motivation, which are crucial to the students' learning experience.

III. Cross-Platform Framework and System Architecture

The PCR system architecture (Fig. 1.), which allows cross-platform compatibility, comprises a client-side and a server-side. The client-side consists of the user's web browser and a Frontend User Interface. The frontend communicates with the Backend Server via the REST API. For frontend user interface (UI) development, React JS was chosen for its flexibility, efficiency and speed. At the same time, Django was selected for the backend because of its extensive built-in support for the data model. The server-side comprises a Backend Server with three parts: an API exposing REST endpoints, the business logic of the system, and a data model for interacting with the database. The architecture diagram below also shows four additional components connected to the Backend Server. These components comprise an email server, a message broker and worker nodes, and a relational database.

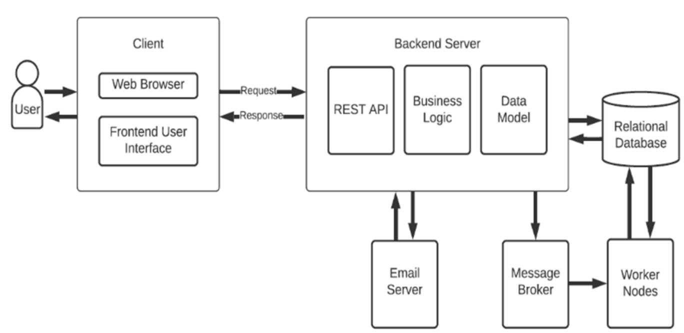

Fig. 1. System Architecture

IV. User Interface

The instructor can create courses, assignments, and rubrics. The instructor will enter the title, weight, task description and select the rubric the new assignment will use (see Fig. 2). The system utilises double-blind peer review to provide students with anonymity in all stages of the process. An assignment consists of several stages in a fixed order. Each stage has a deadline to ensure all students receive feedback simultaneously and have equal time to complete their work. Students first submit their initial code (Fig. 4) and then review three of their peers' code (Fig. 3). Once the review deadline has passed, feedback is released to students. Three reviews ensure students get the feedback they need, and the overhead for students is not a burden. Students would then rate the reviews they received (Fig. 5). Having students rate the reviews they receive aims at fostering student engagement with the review process. Reviews are rated based on four categories: accuracy, fairness, identification of strengths, and actionable improvements. Review ratings are released to reviewers once the rate review deadline has passed. Assignments can be formative or summative. Summative assignments involve a further stage wherein students submit their revised code after peer-review.

Assignment weights can be configured to specify how the student's work is graded. Students can receive marks for submitting code and reviewing their peer's code. Rubric questions can contain marks that allow for code quality to be numerically computed. Similarly, rating reviews allows for review quality to be numerically computed. Students can receive marks for rating their peer's reviews as an incentive. In the case of summative assignments, a student can receive marks for submitting a revised code. The weight of the final submission on the assignment grade and the number of marks it is marked out of can be configurable. Instructors can mark and provide feedback on final student submissions. Students are also awarded badges (Fig. 6) for completing reviews and high-quality reviews. The purpose of badges is to improve student motivation. Students are awarded bronze, silver, and gold badges for high-quality reviews.

V. User Evaluation of System, Engagement, and Motivation

A. Methodology

Participants ($n = 22$) - 17 Year 3 & 4 university students and 5 instructors - completed a task-centric activity. Instructors were tasked to create a new course assignment, add students to the new course, mark final submissions, download the results, and publish the assignment results. On the other hand, students had to submit their initial code, complete up to 3 peer reviews, rate any review they received and explore badges they were awarded. The instrument comprised the System Usability Scale (SUS) developed by Brooke [25]. It consists of ten statements (see Table I). Participants rated the extent to which they agreed with each statement (1 strongly disagree to 7 strongly agree). The student version of the SUS questionnaire did not include the first statement, "I think that I would like to use this system frequently" because students only use the system to complete assignments set by instructors. The student questionnaire included additional questions related to student engagement and motivation (Table II).

TABLE I. System Usability Scale questions[25]

	SUS Questions
Q1	I think that I would like to use this system frequently
Q2	I found the system unnecessarily complex
Q3	I thought the system was easy to use
Q4	I think that I would need the support of a technical person to be able to use this system
Q5	I found the various functions in this system were well integrated.
Q6	I thought there was too much inconsistency in this system.
Q7	I would imagine that most people would learn to use this system very quickly.
Q8	I found the system very cumbersome to use.
Q9	I felt very confident using the system
Q10	I needed to learn a lot of things before I could get going with this system.

TABLE II. Student engagement and motivation questions

	Questions on engagement and motivation
1	I feel more engaged in the review process because I know my reviews will be rated.
2	I am more likely to learn what makes a good review because I know my reviews will be rated.
3	It does not make a positive difference to my motivation when I see I will be awarded badges for completing reviews.
4	I feel more motivated when I see I will be awarded bronze, silver or gold badges for high-quality reviews.
5	I would prefer not to see badges in the system.
6	I enjoyed reviewing other students' codes.
7	I feel I have learned from the review I received.

Questions one and two aimed to evaluate the effect of rating reviews on student engagement with the review process. Questions three through five aimed to evaluate the effect of being awarded badges on student motivation. Questions six and seven aimed to understand the student's experience quantitatively. Furthermore, both students and instructors were asked two open-ended questions about what they liked and disliked.

Analysis of the system usability score consists of four steps [26]. Step one is to convert participant ratings into points. For questions that generate a positive response, *points = rating - 1*. For questions that generate a negative response, *points = 7 - rating*. Points range from 0 to six, with six being the most positive response. Step two is to sum the points to get the total points for each participant. Step three is to calculate the SUS score of each participant by scaling the total points out of 100. In the final step, the SUS score of the system is calculated as the average SUS score over all participants.

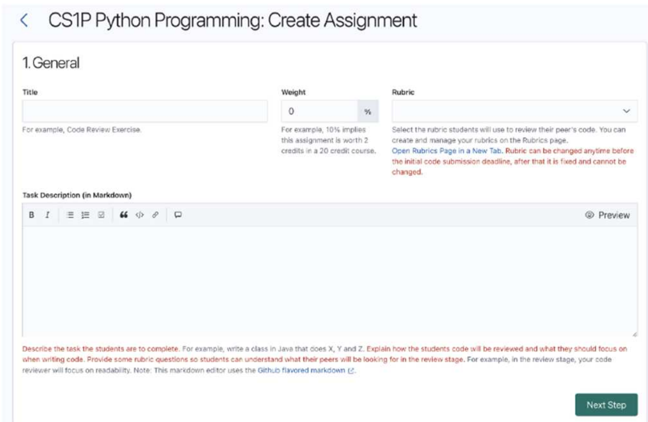

Fig. 2. Instructor "Create Assignment" page

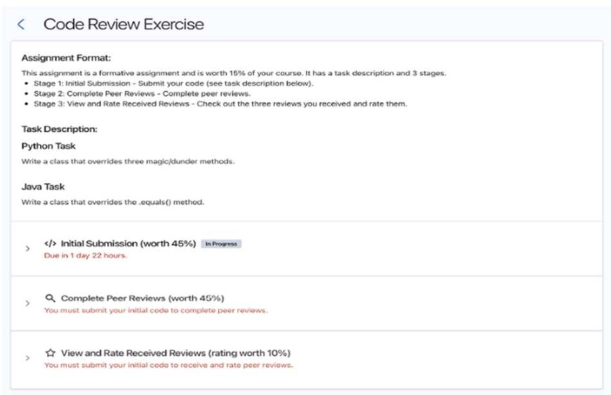

Fig. 3. Student peer code review page

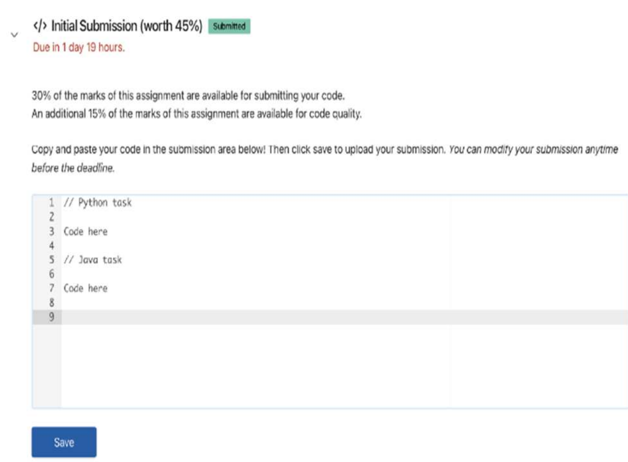

Fig. 4. Student submission page

B. System Usability Results

The average SUS score for instructors and students is 82 (Table III) and 92 (Table IV), respectively. The average SUS score is 68, and a score above 80 is considered to be in the top 10% of scores [27]; therefore, the PCR ranks in the top 10% on the System Usability Scale. Overall, both instructors and students reported that the Peer Code Review system was easy to use as it is a "very intuitive system", "pleasing minimalist design", and "dark mode is also a good addition, easy on the eyes". However, the students disliked the programming exercise itself, saying "the exercises were quite trivial".

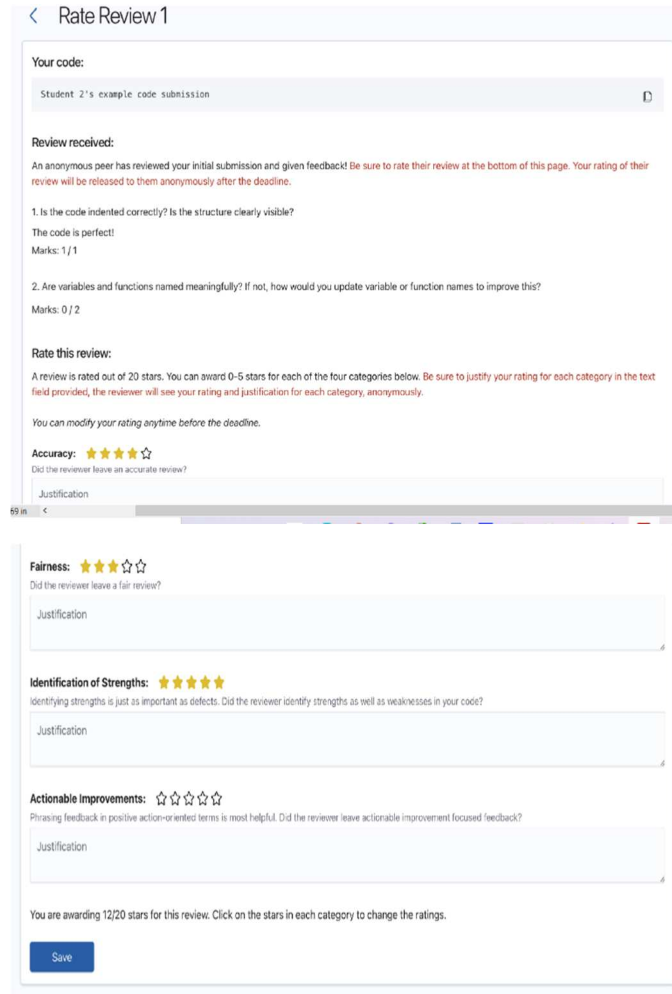

Fig. 5. Rate review Page

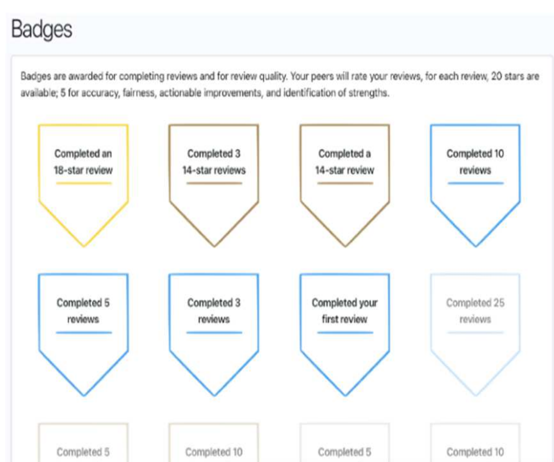

Fig. 6. Badges

TABLE III. INSTRUCTORS' SUS RESULTS

SUS Question Participant	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total Points	SUS Score
Instructor 1	4	4	5	5	3	4	4	4	4	4	41	68.33
Instructor 2	5	5	5	6	4	5	4	6	5	6	51	85
Instructor 3	4	5	5	6	6	6	6	6	6	6	56	93.33
Instructor 4	5	6	6	6	5	5	5	6	5	6	55	91.67
Instructor 5	4	4	4	5	4	4	4	5	4	5	43	71.67

TABLE IV. STUDENTS' SUS RESULTS

SUS Question Participant	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total Points	SUS Score
Student 1	5	5	6	6	5	4	5	6	3	45	83.33
Student 2	5	6	6	5	6	6	6	6	5	51	94.44
Student 3	5	5	6	6	6	6	4	5	4	47	87.04
Student 4	5	6	5	2	5	5	6	6	6	46	85.19
Student 5	6	6	6	6	6	6	6	6	6	54	100
Student 6	6	6	6	6	6	6	6	6	6	54	100
Student 7	6	6	6	6	6	6	6	6	6	54	100
Student 8	5	5	6	6	5	6	5	6	5	49	90.74
Student 9	5	6	5	5	6	6	6	4	5	48	88.89
Student 10	6	6	6	6	6	6	6	6	6	54	100
Student 11	5	6	6	6	6	6	6	6	5	52	96.30
Student 12	6	6	6	5	5	6	5	5	6	50	92.60
Student 13	5	5	6	6	6	6	6	5	6	51	94.45
Student 14	4	4	6	5	6	5	5	4	5	44	81.48
Student 15	6	6	6	5	6	6	6	5	5	51	94.45
Student 16	5	5	4	5	6	5	5	4	4	43	79.63
Student 17	6	6	5	6	6	6	6	6	6	53	98.15

C. *Results on student engagement and motivation*

The student questionnaire included additional questions related to student engagement and motivation (see Table II). Students were asked if they felt more engaged in the review process because they knew their reviews would be rated. Most of them (88%) agreed or strongly agreed with this statement, with 65% of those students strongly agreeing. Students were also asked if they were more likely to learn what makes a good review because they knew their reviews would be rated. Similar results were found. Again, 88% of students agreed or strongly agreed. Six per cent of students disagreed in both cases, while 6% neither agreed nor disagreed. That sense of engagement and motivation was also felt in their comments from open-ended questions, as many students positively mentioned rating reviews. For example, one student stated, "*Having our reviews rated was a good addition. Means you are more likely to spend more time writing the review and thinking critically about how to write a good review*". Another student said, "*great to know how well you performed in rating and how well is the other person satisfied*".

Students were asked if knowing that they would be awarded badges for completing reviews made a positive difference to their motivation. Most students (70%) agreed and strongly agreed with this statement, and 18% disagreed. An additional 12% of students neither agreed nor disagreed with the statement. When asked if they would prefer to see badges in the system, 88% of students agreed and strongly agreed with this statement, and 6% disagreed. In comparison, 6% of student participants neither agreed nor disagreed with the statement. Students were also asked if they felt more motivated knowing that they would be awarded badges for completing high-quality reviews. Most of them (70%) agreed or strongly agreed with this statement. However, 24% of students disagreed with the statement, while a small subset (6%) neither agreed nor disagreed. Students' comments on badges were also positive; for example, one student commented, "*very good system, easy to use and encourages fair and correct code reviews through the use of badges and code review reviews*".

VI. CONCLUSION

This paper presents a peer code review system (PCR). The system enables the instructor to create formative or summative code review assignments. The students can submit their code, review their peers' code, rate the reviews they received on their assignment, and earn badges as part of task completion. If the code review is a summative assignment, the instructors will mark the students' final code submission. The evaluation of the system yielded very positive results not just in terms of the system evaluation itself but its potential to help solve some of the issues identified in peer review activities, namely the lack of student engagement and lack of student motivation [10][6][9] [7] [8]. It indicates that a system that fosters the peer review process through rating reviews and badges can be engaging and motivating. These are a combination of components existing applications discussed in this paper lacked.

However, the main limitation is the size of the sample, which was small. A bigger sample should be considered. Furthermore, as a student commented, the exercises were trivial; therefore, the instructors should create peer-review assignments that are significantly linked to their curriculum. However, students taking part in this evaluation already had some programming experience which may have instigated these comments. A sample of student participants with different programming experience levels will be considered in the future.

References

[1] E. Forsberg, L. Geschwind, S. Levander, and W. Wermke. (2022) "Peer Review in Academia," in Peer review in an Era of Evaluation, E. Forsberg, L. Geschwind, S. Levander and W. Wermke, Eds. Palgrave Macmillan, 2022, pp. 3-36.

[2] J. P. Tennant, and T. Ross-Hellauer. "The limitations to our understanding of peer review," *Research Integrity and Peer Review*, vol. 5, no. 6, pp. 1-14. 2020, doi: 10.1186/s41073-020-00092-1.

[3] N. U. Eisty, and J. C. Carver. "Developers perception of peer code review in research software development," *Empirical Software Engineering,* vol. 27, no. 13. 2022, doi: 10.1007/s10664-021-10053-x.

[4] K. Lundstrom, and W. Baker. "To give is better than to receive: The benefits of peer review to the reviewer's own writing," *Journal of Second Language Writing*, vol. 18, no. 1, pp. 30–43, 2009, doi: 10.1016/j.jslw.2008.06.002.

[5] T. D. Indriasari, A. Luxton-Reilly, and P. Denny. "Improving Student Peer Code Review Using Gamification," in *Australasian Computing Education Conference* (ACE '21). Association for Computing Machinery, New York, NY, USA, pp. 80–87, Feb 2021, doi: 10.1145/3441636.3442308.

[6] R. A. Mulder, J. Pearce, and C. Baik. "Peer review in higher education: Student perceptions before and after participation," *Active Learning in Higher Education*, vol. 15, no. 2, pp. 157–171, 2014.

[7] Y. Wang, L. Yijun, M. Collins, and P. LIU. "Process improvement of peer code review and behavior analysis of its participants," *in Association for Computing Machinery SIGCSE Bulletin*, vol. 40, no. 1, pp. 107–111, March 2008, doi:10.1145/1352322.1352171.

[8] Y.Wang, H. Li, Y. Feng, Y. Jiang, and Y. Liu. "Assessment of programming language learning based on peer code review model: Implementation and experience report," *in Computers & Education*, vol. 59, no. 2, pp. 412–422, 2012, doi:10.1016/j.compedu.2012.01.007.

[9] S.Turner, M. A. Pérez-Quiñones, S. Edwards, and J. Chase. "Student attitudes and motivation for peer review in CS2," *in Proceedings of the 42nd ACM technical symposium on Computer science education (SIGCSE '11).* Association for Computing Machinery, New York, NY, USA, pp. 347–352, 2011, doi:10.1145/1953163.1953268.

[10] T. D. Indriasari, A. Luxton-Reilly, and P. Denny. "A review of peer code review in higher education," *ACM Transactions on Computing Education*, vol. 20, no. 3, pp. 1-25, Sept. 2020. doi: 10.1145/3403935.

[11] K. Cho, C. Schunn, and R. Wilson. "Validity and reliability of scaffolded peer assessment of writing from instructor and student perspectives," *Journal of Educational Psychology*, vol. 98, no. 4, pp. 891–901, 2006, doi: 10.1037/0022-0663.98.4.891.

[12] S. Gielen, L. Tops, F. Dochy, P. Onghena, and S. Smeets. "A comparative study of peer and teacher feedback and of various peer feedback forms in a secondary school writing curriculum," *British Educational Research Journal*, vol. 36, n. 1, pp. 143–162, 2010, doi: 10.1080/01411920902894070.

[13] K. Reily, P. Finnerty, and L. Terveen. "Two peers are better than one: Aggregating peer reviews for computing assignments is surprisingly accurate," *in Proceedings of the ACM 2009 international conference on Supporting group work*. Association for Computing Machinery, New York, USA, pp. 115–124, May 2009.

[14] J. Pearce, R. Mulder, and C. Baik. "Involving students in peer review: case studies and practical strategies for university teaching,". Centre for the Study of Higher Education, University of Melbourne, 2009, https://apo.org.au/node/20259.

[15] X. Song, S. C. Goldstein, and M. Sakr. "Using peer code review as an educational tool," *In Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education, ITiCSE '20*. Association for Computing Machinery, New York, NY, USA, pp.173–179, Jun 2020.

[16] R. Lu, and L. Bol. "A comparison of anonymous versus identifiable e-peer review on college student writing performance and the extent of critical feedback," *Journal of Interactive Online Learning*, vol. 6, n.2, pp. 100–115, 2007, https://digitalcommons.odu.edu/efl_fac_pubs/5/.

[17] G. Morales-Martinez, P. Latreille, and P. Denny. "Nationality and Gender Biases in Multicultural Online Learning Environments: The Effects of Anonymity," *in Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, pp. 1–14, Apr 2020.

[18] E. Panadero, M. Romero, and J.-W. Strijbos. "The impact of a rubric and friendship on peer assessment: Effects on construct validity, performance, and perceptions of fairness and comfort," *Studies in Educational Evaluation*, vol. 39, no. 4, pp. 195–203, 2013, doi. 10.1016/j.stueduc.2013.10.005.

[19] C. D. Hundhausen, A. Agrawal, and P. Agarwal. "Talking about code: Integrating pedagogical code reviews into early computing courses," *ACM Transactions on Computing Education*, vol. 13, no. 3, pp. 1-28, Aug. 2013, doi: 10.1145/2499947.2499951.

[20] E. Z. F. Liu, P-K. Chen. "The Effect of Game-Based Learning on Students' Learning Performance in Science Learning – A Case of 'Conveyance Go'," *in Procedia - Social and Behavioral Sciences*, vol. 103, pp. 10-44 – 1051, Nov 2013.

[21] S. Nicholson. A RECIPE for Meaningful Gamification. In: Reiners T., Wood L. (eds) Gamification in Education and Business. Springer, Cham. https://doi.org/10.1007/978-3-319-10208-5_1

[22] Peergrade. Peergrade - engaging student peer review, 2021. URL https://www.peergrade.

[23] Aropa, 2021. URL http://www.dcs.gla.ac.uk/~hcp/aropa/.

[24] H. Purchase, and J. Hamer. "Peer-review in practice: eight years of aropä," *Assessment & Evaluation in Higher Education*, vol. 43, no. 7, pp. 1146–1165, 2018.

[25] J. Brooke. "Sus: A quick and dirty usability scale," in Usability Evaluation In Industry, vol. 189, 1995.

[26] K. Betteridge. "What every client should know about sus scores", 2021. https://www.bentley.edu/centers/user-experience-center/-every-client-should-know-about-sus-scores. Accessed 18 June 2021

[27] J. Sauro. "Measuring usability with the system usability scale (sus)," 2011. https://measuringu.com/sus/. Accessed 18 June 2021.