2023 Annual Conference & Exposition
Baltimore Convention Center, MD | June 25 - 28, 2023
The Harbor of Engineering Education for 130 Years
Paper ID #39140

Work in Progress: Engineering together - Applying remote collaborative technology to an in-person undergraduate engineering course

Mr. John William Lynch, University of Cincinnati

John Lynch is an engineering and computing education PhD student at the University of Cincinnati. His research interests are in improving computer science education for undergraduates by leveraging technology and unique pedagogical interventions. His overarching goal is to increase the retention rate for studying Computer Science at all education levels and make the field accessible for underrepresented populations.

Dr. Jutshi Agarwal, University of Cincinnati

Jutshi Agarwal is a Doctoral candidate in Engineering Education at the University of Cincinnati. She has a Master's degree in Aerospace Engineering from University of Cincinnati and a Bachelor's degree in Aerospace Engineering from SRM University, India. Her research areas of interest are graduate student professional development for a career in academia, preparing future faculty, and using AI tools to solve non-traditional problems in engineering education. She has published in several international conferences.

Dr. P.K. Imbrie, University of Cincinnati

P.K. Imbrie is the Head and Professor of the Department of Engineering & Computing Education and a Professor in the Department of Aerospace Engineering and Engineering Mechanics at the University of Cincinnati. He received his B.S., M.S., and Ph.D. degrees in Aerospace Engineering from Texas A&M University and is an ASEE Fellow. He is an advocate for research-based approaches to engineering education, curricular reform, and student retention. Imbrie has been involved in both traditional, as well as educational research in experimental mechanics, piezospectroscopic techniques, epistemologies, assessment, and modeling of student learning, student success, student team effectiveness, and global competencies. He helped establish the scholarly foundation for engineering education as an academic discipline through lead authorship of the landmark 2006 JEE special reports "The National Engineering Education Research Colloquies" and "The Research Agenda for the New Discipline of Engineering Education." He has a passion for designing state-of-the-art learning spaces. While at Purdue University, Imbrie co-led the creation of the First-Year Engineering Program's Ideas to Innovation (i2i) Learning Laboratory, a design-oriented facility that engages students in team-based, socially relevant projects. While at Texas A&M University Imbrie co-led the design of a 525,000 square foot state-of-the-art engineering education focused facility; the largest educational building in the state.

Professor Imbrie's expertise in educational pedagogy, student learning, and teaching has impacted thousands of students at the universities for which he has been associated. He is internationally recognized for his work in active/collaborative learning pedagogies and is a co-author of a text on teaming called Teamwork and Project Management. His engineering education leadership has produced fundamental changes in the way students are educated around the world. His current research interests include: epistemologies, assessment, and modeling of student learning, student success, and student team effectiveness.

WIP: Engineering together - Applying remote collaborative technology to an in-person undergraduate engineering course.

Abstract

This evidence-based Work in Progress research paper will explore how collaborative technology impacts student engagement with teams and programming activities in an introductory first-year engineering course. Introduction to engineering has been a historically difficult course for undergraduates as they are introduced to algorithmic thinking, design processes, and problem-solving methodologies. To assist students, a variety of approaches can be employed in the classroom; team-based capstone projects with end of course demonstrations, synchronous collaborative technology that supports teamwork and communication in and out of class, pair-programming, and visual-based programming languages. Each of these provides benefits to the students individually, but with COVID-19 forcing a shift to remote learning, collaborative technologies experienced an unprecedented development of innovations and tools. A return to in-person classes may incline educators to drop collaborative technologies for teaching, but remote control, screen share, and collaborative tools are still beneficial if using teamwork in the class. This paper investigates the following research question: to what degree is student engagement impacted by the usage of synchronous collaborative tools in a team-based, in-person undergraduate introductory engineering course? An experimental setup was implemented in three different sections of an introductory engineering course at a large, midwestern, R1 institution. All three sections had different instructors and a class size of around 50 students. One of the three sections implemented a technology that allowed students to connect to a teammate's computer and share control of keyboard/mouse, enabling real-time collaborative programming in a normally individual programming environment. The other two sections were control sections with no specific implementation differences. A survey instrument grounded in Burch et al.'s conceptual framework was developed and distributed at strategic times to measure students' engagement with their team and in-class programming tasks. Results presented include a Confirmatory Factor Analysis (CFA) that supports the factor structure of the student engagement survey and an analysis of variance (ANOVA) procedure to compare the three sections and investigate significant differences between them through student grades. The results of this research have potential to provide direction for usage of remote collaborative technology for in-person, academic settings. Future implications of research include investigating the impact of similar technologies on student engagement and learning outcomes; contributing a validated instrument to measure students' engagement with their programming tasks and teams; and provide educators with potential methodologies to improve student engagement in team-based coursework.

Introduction

Engineering has historically suffered high student attrition rates [1], [2], [3], with a significant portion of engineering students deciding to leave their major within the first or second year [3],[4]. To help improve engineering undergraduate retention, engineering courses began adoption of project-based learning. These are often team-based courses where students solve an engineering project modeled off real-world case studies and experience the development, testing, and implementation of a solution utilizing design processes and computational skills [5], [6], [7], [8]. Project based learning has reported increased learning outcomes for students [9], [10]. This pedagogy uniquely engages the students with teams and facilitates development of functional teaming and engineering problem solving competencies, characteristics of future engineers required by ABET [11]. Increased student engagement has also been a reliable predictor of higher retention rates. An approach to improving engagement of students in the context of project-based learning is to leverage technologies that support teamwork and communications in and out of class, referred to as Computer Supported Collaborative Learning (CSCL) technologies. CSCL is grounded in the educational theories of social-constructivism that emphasize active, team-based classrooms; the importance of students' communication; students' experience of the environment; and the advancement of collective understanding [12],[13],[14],[15]. In his seminal work, Stahl [16] introduced the concept of computers acting as "mediators" for different forms of collaboration between students. Jeong & Hmelo-Silver [17] created four themes associated with a computer's support of collaboration – collaborative knowledge building; personal perspectives intertwined with the team; technology mediating the team; and analysis of team interactions via technology. They further elaborated these theories by producing some affordances technologies should provide for learners. The CSCL technology should allow learners to have engagement in joint-task, communication between teams, sharing of resources, engagement in collaborative learning processes, and co-construction [17]. The broad definition and multiple affordances allow for CSCL technologies to be successfully implemented at different levels in engineering contexts. Examples of some of these implementations are discussed in the following sub-sections.

CSCL in undergraduate engineering courses, first-year engineering courses, and programming

Prior research applied CSCL technologies in undergraduate engineering education settings. For example, senior students were able to utilize 3D robotic simulation software which facilitated synchronous collaborative movement of the robot via a desktop application, which simulates an environment useful for limited knowledge users [18]. An industrial design course modified its curriculum to use Instagram to archive student presentations and facilitate discussions which coincide with the CSCL affordances of sharing of resources and communication between teams [19]. Student responses indicated improved comprehension of concepts and improved overall knowledge of the subject matter, with in-class discussions followed by virtual discussion as the most useful aspect [19]. Augmented reality has been utilized in first-year engineering courses, where students who collaborated in augmented reality performed better on learning outcome tasks than those who collaborated normally [20]. In a different study, student teams in design ideation groups for their first-year engineering project were split into two modalities; virtual teams that used online note-sharing application and in-person teams that would meet normally [21]. No significant differences were found between the group's design ideation development, suggesting CSCL technologies are as effective as in-person modality in first-year engineering contexts for design ideation [21]. Another study examined how students used Google Docs to manage their team workflow and knowledge generation. They reported role rotation and shared responsibilities of team members, usage of collaborative tool during face-to-face meetings, with 95% of students indicating that the tool was beneficial for the workflow of the team [22]. An intentional application of CSCL to improve student engagement and learning outcomes is pair programming. Pair programming is a software development technique where two individuals work on the same code simultaneously; one as a driver who types code and the other a navigator who reviews, discusses, or dictates code [23],[24],[25]. Implementation in programming and computer science courses showed improved learning outcomes via improved assignments and final exam grades [26], [27], coding quality and achievement [28], and increased student attendance [29]. Recent development of team /mob-programming expands the interaction to three to four students with single driver and multiple navigator roles [30], [31]. To simplify definitions, pair, mob, and team programming are blanketed under team-programming. Bowman et al. [32] concluded that other disciplines should begin to consider application of this pedagogy owing to the increased opportunities for meaningful, academic, and collaborative work.

Current Study and Limitations

The COVID-19 pandemic shifted education modality to a purely remote format. As a result, collaborative technologies experienced an unprecedented development of innovations and tools with both education and industry borrowing techniques from each other. CSCL technologies based on team-programming pedagogies proved indispensable in this modality, with features such as screen share and remote control. The current study aims to gather evidence that advocates for the application of CSCL technologies in an in-person active-collaborative classroom. A CSCL technology commonly used in software industry was piloted in a first-year engineering course to answer the following question: to what degree is student engagement and student learning/performance impacted by the usage of synchronous collaborative tools in a team-based, in-person undergraduate introductory engineering course? To answer the research question a conceptual framework grounded in student engagement literature was used to create an instrument to measure student engagement. To measure learning, specific programming content of the course were used as metrics for learning/performance outcomes for students. The following sections discuss the conceptual framework and development of the student engagement measurement instrument. Development of the student engagement instrument went through an iterative process. Data collected from the first distribution was used to modify the instrument to a 6-point Likert scale and require inverse-coded questions but was not used to analyze programming scores or student engagement. Future iterations of this study will need more datapoints collected, with potential to examine differences throughout the semester. The format of the class and different instructor implementations introduce multiple variables which can confound if engagement and programming score differences can be attributed to CSCL technology. Ways to reduce these confounding variables in control and experimental sections would need to be examined.

Conceptual Framework and Instrument Development

A shared goal of the CSCL methodologies is the improvement of student engagement. In previous studies, measurement of engagement included grades [26], [27], [28], student distributed surveys [19], [22], or number of ideas generated [21]. To expand valid measures, exploration of student engagement literature outside of engineering contexts resulted in more specific ways to measure student engagement. Kuh's [33] seminal work defined student engagement as devotion of energy to coursework. Astin [34] defined it as what the students do during college, in terms of engagement with institution organizations outside class alongside devotion of time and energy to coursework. In Burch's [35] framework, student engagement is comprised of four constructs of emotional engagement, physical engagement, cognitive engagement in class, cognitive engagement out of class. Burch [35] utilized employee engagement literature and framed a students' job as a learner, which developed the four constructs. In the context of the impacts of collaborative technologies in this study, cognitive engagement out of class was not of interest, only three constructs inside the course were considered. Definitions of the constructs are as follows: emotional engagement is defined as student's personal feelings about their experiences with in-class tasks and their interactions with teammates; physical engagement is defined as student's active participation with their teammates on in-class tasks; cognitive engagement is student's individual performance and focus on in-class tasks and their perceptions of teammates performance and focus on in-class tasks. Items in Burch's instrument were reviewed and a pool of new items relevant to the educational context were generated by the first author. The other authors, both educational researchers, reviewed these items and selected a 5-point Likert scale for pilot administration. With pilot data, it was recognized that a process of data validation was not in place. Three reverse coded items for each construct were added to the original instrument to compare responses in other items and eliminate erroneous data. The modified instrument also used a 6-point Likert scale in contrast to the original 5-point Likert scale. The modified scale is presented in Table 1.

Table 1. Modified student engagement survey. Responses on 6-point Likert scale (1 = Strongly Disagree, 6 = Strongly Agree) | (E= Emotional, P= Physical, C= Cognitive)

Question	Statement	Factor
Q1	The resources provided to me made me feel positive about the tasks I completed with my team	E
Q2	The resources provided to me made working with my team enjoyable during in-class programming activities	E
Q3	The resources provided to me allowed me to feel open to frequently discuss errors in my programming code with my teammates during in-class programming activities	E
Q4	The resources provided to me made me feel comfortable asking questions to my teammates about the in-class programming activity	E
Q5	The resources provided to me made it easier to positively contribute to my team's tasks during the in-class programming activities	E
Q6	The resources provided to me made it easy to view my teammates programming work during the in-class programming activities	P
Q7	The resources provided to me allowed me to help teammates that were struggling with completing the in-class programming activities	P
Q8	The resources provided to me allowed me to see my teammate's programming styles and compare, integrate, or contrast them with my own programming style	P
Q9	The resources provided to me allowed me to easily show my programming errors to teammates during the in-class programming activities	P
Q10	The resources provided to me encouraged me to ask questions to teammates during the in-class programming activities	P
Q11	The resources provided to me made it easier to perform programming tasks at a consistent level during the in-class programming activities	C
Q12	The resources provided to me allowed me to regularly pay attention to my teammate's comments about their programming code during in-class programming activities	C
Q13	The resources provided to me allowed me to maintain a similar level of engagement with programming activities in comparison with the rest of my teammates	C
Q14	The resources provided to me allowed my teammates and I to take time to reflect on the solutions we created to the in-class programming activities	C
Q15	The resources provided to me made it easy to concentrate on the programming task during in-class programming activities	C

Methods

Using the modified instrument, data was collected from three sections of a course. The following sub-sections describe the following: educational setting of the study, characteristics of course sections and participants involved with the study, the CSCL intervention implemented, data collection, and data analysis methods implemented in the study.

Educational Setting, Sections and Participants

The study is based in the first semester of a two-semester first-year engineering course sequence in the College of Engineering and Applied Sciences (CEAS) at a large, public, midwestern R1 institution. It introduces concepts and tools for engineering design process including fundamental engineering content, project management, teamwork, and engineering ethics. Algorithmic thinking using multiple computational tools like LabVIEW and Python are also a significant part of course content. Every fall semester, about 1300-1500 students enroll in the course distributed into 24-28 sections, with an average class size of 40-72 students. At the beginning of the semester students are assigned into teams of size 3-4 based on several factors like prior experiences, knowledge, and demographics. Using a flipped-classroom setup, the instructors administer the same in-class activities, quizzes, homeworks, and exams across all sections at the same time throughout the semester. At the end of each class, students are administered a brief quiz that can be in team or individual format as decided by the individual instructor to check for comprehension and attendance. The physical structure of the team seating arrangements in-class divides teams of four into pairs seated across and facing each other. Three sections of the first-year engineering course were selected based on convenience sampling for the researchers. Section A had 40 students, Section B and C had about 50 students. Instructor's enforcement of teaming activities varied by section. Section A's instructor consisted of more laissez-faire attitudes to enforcement of teaming activities which allowed for reduced interactions between team members where preferred, creating a *low interaction* setup between team members. Section B's instructor ensured teams were consistent with work on tasks and not off topic and was a *medium interaction* section. Section C ensured teams stayed on topic and utilized the CSCL technology and was considered a *high interaction* environment between team members.

CSCL intervention for high interaction section

The CSCL intervention in Section C was implemented by means of ParSec. ParSec, originally designed for gaming, is an application that captures the desktop screen of a computer through low latency video streaming and streams it through the ParSec application. Each user must download the application. A host can use the application to stream their desktop screen, and other users can join using the application and see the desktop screen. The host can allow users access to their computers mouse, keyboard, and specific applications that permit movement of mouse or keyboard strokes. ParSec allows hosts to disable access to folder systems and the ability to click taskbars to close out of applications. The potential use case of this technology is the ability for synchronous collaborative editing in applications that normally cannot support that format, and for teams to share a single screen for in-class programming tasks. In this format, students would be team programming [30] [31]. By introducing and enforcing the use of ParSec during the in-class programming activities of Section C, it is believed to have implemented *high interaction* within team's set-up where all members of the team were actively working on the same task during class time.

Data collection and analysis

The student engagement survey was distributed to all three sections when students finished their final in-class Python programming exercises. A total of n = 133 student responses were collected. After final exams, student grades of in-class programming quizzes and exam programming questions were collected and summed for each student, resulting in a sum score of in-class programming quizzes and a sum score for exam programming questions. Analysis was conducted in

RStudio using Build 23.5 and the lavaan package. Results that indicated the same response for all questions or the same response for both the item and its inverse related inverse item were removed to account for erroneous data. With this sample, reliability and validity analyses were conducted on the student engagement survey. Overall sum engagement score and scores of the individual factors of emotional, physical, and cognitive engagement were compared between Sections A, B, and C. To analyze learning/performance impacts scores of in-class programming quizzes and programming questions on three exams were analyzed. Descriptive statistics of the student scores are given below. To check for differences between sections, tests for normality were performed on student engagement responses and the student grades. Data failed assumptions of normality based on Shapiro-Wilk tests and examined Q-Q plots, so non-parametric tests for significant differences between the sections were performed using Kruskal-Wallis to analyze for significant differences between the medians of each section. Subsequently, a post-hoc Dunn's test with Bonferroni corrections was conducted to analyze which sections were significantly different. The sample size for each of the sections were as follows: Section A had a sample size of 33 (n = 33), Section B had a sample size of 38 (n = 38), and Section C had a sample size of 50 (n=50).

Table 2. Descriptive statistics of student engagement scores (n = 121)

Measurement	Total	Emotional	Physical	Cognitive
Mean	67.61	23.13	22.50	21.98
Median	69.00	24.00	23.00	22.00
Range	55.00	22.00	20.00	20.00
SD	9.47	3.30	3.72	3.68
SE	0.89	0.30	0.34	0.33

Table 3. Descriptive statistics of student programming scores (n = 121)

Measurement	In-class programming quizzes	Exam programming questions
Mean	53.12	57.67
Median	54.50	64.14
Range	36.00	68.30
SD	6.71	16.27
SE	0.61	1.48

Results and Discussion

Evidence of reliability and validity of student engagement measurement

The developed student engagement measurement scale was tested on a three-factor model. The instrument contained 15 items with 5 items loaded onto each factor, and loadings of each item are presented in Table 6. Standards for the indices of fit for the CFA were based on recommendations from [36] and [41]. RMSEA < .06 indicates good fit, < .08 indicates reasonable fit. CFI and TLI > .95 indicates adequate fit, SRMR <.10 indicate acceptable model fit. Because of the ordinal nature of the Likert scale response items, Weighted Least Square (WLSMV) was used in Confirmatory Factor Analysis. Sample size (n = 121) was slightly under than recommended but in acceptable range [42]. Chi-square analysis indicated $\chi 2(87) = 131.547$, p-value = .001. This model reports unacceptable fit by chi-square (p= < .001), which is to be expected in large sample sizes. However, indices of fit for CFI = 0.994, TLI = 0.993 are within an adequate fit. RMSEA = 0.065 indicates reasonable fit, and SRMR = 0.069 indicate acceptable model fit. Reliability was adequate for emotional engagement (α = .81), physical engagement (α = .82), and cognitive engagement (α = .83), and strong for total student engagement (α = .91). Fit indices indicate that the theoretical model adequately fits the data collected.

Table 4. Three factor model for student engagement measurement (n = 121)

Question	Standardized Factor Loadings:				
	Emotional Engagement	Physical Engagement	Cognitive Engagement	Cronbach's Alpha	
Q1.	.80	0	0	0.81	0.91
Q2.	.77	0	0		
Q3.	.70	0	0		
Q4.	.61	0	0		
Q5.	.80	0	0		
Q6.	0	.73	0	0.82	
Q7.	0	.80	0		
Q8.	0	.73	0		
Q9.	0	.78	0		
Q10.	0	.71	0		
Q11.	0	0	.76	0.83	
Q12.	0	0	.76		
Q13.	0	0	.70		
Q14.	0	0	.80		
Q15.	0	0	.83		

Survey was distributed after completion of in-class programming tasks. Student's responses to 15 questions were translated to a numeric scale (Strongly Disagree = 1, Strongly Agree = 6).

Table 5. Variance of student engagement by section using Kruskal-Wallis

Engagement	$\chi 2$	**df**	**p**
Total	4.830	2	.089
Emotional	4.505	2	.150
Physical	4.830	2	.089
Cognitive	5.122	2	.077

After translation, results were analyzed and indicated no statistically significant differences in total engagement ($p = .089$), emotional engagement ($p = .150$), physical engagement ($p = .089$), or cognitive engagement ($p = .077$) between three sections. Programming scores were then analyzed.

Table 7. Variance of student programming scores by section using Kruskal-Wallis

Programming Scores	$\chi 2$	**df**	**p**
In-class programming quizzes	20.843	2	2.979e-05
Exam programming	11.23	2	0.004

Results indicate statistically significant differences between the three sections for in-class programming quiz grades ($p= 2.979e\text{-}05$) and exam programming grades ($p=.004$). A post hoc Dunn's test with Bonferroni corrections was conducted to explore which specific sections had significant differences. Results indicated significant differences for in-class programming quiz scores between Section A and Section B ($p = 1.70e\text{-}5$), Section A and Section C ($p = 0.029$), and

Section B and Section C (p = .009). Results suggested significant differences in programming exam scores between Section A and Section B ($p = .003$), and Section A and Section C (p = *.017*). No significant differences were found between Section B and Section C ($p = .336$) for exam programming grades.

The goal of this evidence-based work-in-progress paper was to investigate the following question: to what degree is student engagement impacted by the usage of synchronous collaborative tools in a team-based, in-person undergraduate introductory engineering course? Non-parametric analysis of student responses to overall engagement and levels of emotional, physical, and cognitive engagement revealed no significant differences between the sections. While contrary to our hypothesis, there may be confounding factors that could explain the non-significant results. One of the primary reasons for non-significant results could be the fidelity of the instrument itself. The student engagement instrument may be subject to critiques found in student engagement literature. There is reported lack of explicit understanding on relationships between the types of engagement and how they relate to learning [37]. Cognitive engagement outside of the classroom [35] could be an important predictor not considered for this scale. Oversimplification of student engagement has been a consistent issue [38], [39], so re-analysis of aspects the scale might not be measuring would be necessary. No measures of the interaction levels of each section were accounted for by the student engagement instrument. For example, differences or similarities in student characteristics between the sections, differences in instructor teaching styles, and the engaging nature of a flipped-classroom set-up itself could be potential contributors to the similarity between experimental and control sections. Observation protocols during in-class activities and other instruments measuring engagement might show different results than measuring the student perception of their engagement through self-reported surveys. While student engagement levels between experiment and control sections were not significantly different, significant differences were found in student learning by analysis of quiz and exam grades. This indicates that the use of CSCL technologies could positively impact student outcomes and points to a need for re-examination of the student engagement instrument. Post-hoc analysis shows that exam scores in experimental Section C and B were not significantly different. The higher interaction rate in these sections allowed for more team programming, improving exam grades. Instructors could alter quiz formats to be done individually or in teams, which would alter overall scores. Exam grades are stricter and can better indicate programming competencies.

This work-in-progress paper provides evidence that remote CSCL techniques in in-person classes can impact student outcomes, specifically in programming. Next steps for this research would be to repeat the experiments and distribute the survey to more sections of the course, allowing for a larger sample size. A re-examination of the student engagement instrument, including reliability and validity analyses can be done. Exploration of student engagement literature can provide insights into how to improve the instrument and discuss if outside class student engagement is an important predictor of student outcomes inside the classroom. Additional observation protocols will be appended onto the student survey to ensure the section instructor styles are more effectively documented and potentially more aligned. Information regarding how programming quizzes were distributed, individual or in teams, will be documented to provide further insights and provide deeper conclusions for future iterations of this work.

References

1. E. Seymour and N. M. Hewitt, *Talking about leaving: why undergraduates leave the sciences*. Boulder, Colo: Westview Press, 1997.
2. "STEM Education Data and Trends." https://www.nsf.gov/nsb/sei/edTool/data/college-10.html (accessed Feb. 22, 2023).
3. L. J. Shuman, C. Delaney, H. Wolfe, and A. Scalise, "Engineering Attrition: Student Characteristics And Educational Initiatives," in *1999 ASEE Annual Conference and Exposition Proceedings,* Charlotte, North Carolina, Jun. 1999, p. 4.229.1-4.229.12
4. S. Krause, J. Middleton, E. Judson, J. Ernzen, K. Beeley, and Y.-C. Chen, "Factors Impacting Retention and Success of Undergraduate Engineering Students," in *2015 ASEE Annual Conference and Exposition Proceedings*, Seattle, Washington, Jun. 2015, p. 26.758.1-26.758.19. doi: 10.18260/p.24095.
5. J. J. Kellar et al., "A problem based learning approach for freshman engineering," 30th Annual Frontiers in Education Conference. Building on A Century of Progress in Engineering Education. Conference Proceedings (IEEE Cat. No.00CH37135), Kansas City, MO, USA, 2000, pp. F2G/7-F2G10 vol.2, doi: 10.1109/FIE.2000.896561.
6. C. E. Hmelo-Silver, "Problem-based learning: What and how do students learn?," *Educational psychology review*, vol. 16, pp. 235–266, 2004.
7. H. Matusovich, M. Paretti, B. Jones, and P. Brown, "How Problem-based Learning and Traditional Engineering Design Pedagogies Influence the Motivation of First-year Engineering Students," in *2012 ASEE Annual Conference & Exposition Proceedings*, San Antonio, Texas, Jun. 2012, p. 25.702.1-25.702.17. doi: 10.18260/1-2--21459.
8. K. Gary, "Project-Based Learning," in Computer, vol. 48, no. 9, pp. 98-100, Sept. 2015, doi: 10.1109/MC.2015.268.
9. M. J. Prince and R. M. Felder, "Inductive Teaching and Learning Methods: Definitions, Comparisons, and Research Bases," Journal of Engineering Education, vol. 95, no. 2, pp. 123–138, Apr. 2006, doi: 10.1002/j.2168-9830.2006.tb00884.x.
10. S. Palmer and W. Hall, "An evaluation of a project-based learning initiative in engineering education," European Journal of Engineering Education, vol. 36, no. 4, pp. 357–365, Aug. 2011, doi: 10.1080/03043797.2011.593095.
11. "Criteria for Accrediting Engineering Programs, 2021 – 2022 | ABET." https://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-engineering-programs-2021-2022/ (accessed Feb. 23, 2023).
12. J. Dewey, Experience and education, First free press edition 2015. New York London Toronto Sydney New Delhi: Free Press, 2015.
13. L. S. Vygotskij and M. Cole, Mind in society: the development of higher psychological processes, Nachdr. Cambridge, Mass.: Harvard Univ. Press, 1981.
14. B. Kim, "Social constructivism," Emerging perspectives on learning, teaching, and technology, vol. 1, no. 1, p. 16, 2001.
15. R. J. Amineh and H. D. Asl, "Review of constructivism and social constructivism," Journal of Social Sciences, Literature and Languages, vol. 1, no. 1, pp. 9–16, 2015.
16. G. Stahl, "Contributions to a theoretical framework for CSCL," presented at the Computer Support for Collaborative Learning, 2023, pp. 62–71.
17. H. Jeong and C. E. Hmelo-Silver, "Seven affordances of computer-supported collaborative learning: How to support collaborative learning? How can technologies help?," *Educational Psychologist*, vol. 51, no. 2, pp. 247–265, 2016.

18. C. M. Seow and C. A. Nelson, "Online Robot Simulation for Collaborative Engineering and Education," in Volume 2: 34th Annual Mechanisms and Robotics Conference, Parts A and B, Montreal, Quebec, Canada, Jan. 2010, pp. 961–968. doi: https://doi.org/10.1115/DETC2010-28372
19. H. Q. Cardall and B. F. Howell, "Using Instagram to Increase Student Engagement with Design History," in DS 93: Proceedings of the 20th International Conference on Engineering and Product Design Education (E&PDE 2018), Dyson School of Engineering, Imperial College, London. 6th-7th September 2018, 2018, pp. 726–731.
20. A. Kumar, A. Mantri, G. Singh, and D. P. Kaur, "Impact of AR-based collaborative learning approach on knowledge gain of engineering students in embedded system course," *Educ Inf Technol*, vol. 27, no. 5, pp. 6015–6036, Jun. 2022, doi: 10.1007/s10639-021-10858-9.
21. D. V. Kaweesa, C. McComb, J. Menold, S. Ritter, and N. A. Meisel, "Evaluating Idea Quantity and Variety When Using Digital Collaboration Tools to Support Brainstorming in Non-Collocated Teams," in *Volume 3: 21st International Conference on Advanced Vehicle Technologies; 16th International Conference on Design Education*, Anaheim, California, USA, Aug. 2019, p. V003T04A015. doi: 10.1115/DETC2019-98203.
22. N. Perova-Mello and S. Brophy, "First-Year Engineering Student Perspectives Of Google Docs For Online Collaboration," in *2017 ASEE Annual Conference & Exposition Proceedings*, Columbus, Ohio, Jun. 2017, p. 28364. doi: 10.18260/1-2--28364.
23. K. S. Choi, F. P. Deek, and I. Im, "Exploring the underlying aspects of pair programming: The impact of personality," *Information and Software Technology*, vol. 50, no. 11, pp. 1114–1126, Oct. 2008, doi: 10.1016/j.infsof.2007.11.002.
24. Old Dominion University *et al.*, "Facilitating Online Learning via Zoom Breakout Room Technology : A Case of Pair Programming Involving Students with Learning Disabilities," *CAIS*, vol. 48, pp. 88–92, 2021, doi: 10.17705/1CAIS.04812.
25. J. T. Nosek, "The case for collaborative programming," *Communications of the ACM*, vol. 41, no. 3, pp. 105–108, 1998. https://doi.org/10.1145/272287.272333
26. Z. J. Beasley and A. R. Johnson, "The Impact of Remote Pair Programming in an Upper-Level CS Course," in *Proceedings of the 27th ACM Conference on Innovation and Technology in Computer Science Education Vol. 1*, Dublin Ireland, Jul. 2022, pp. 235–240. doi: 10.1145/3502718.3524772.
27. N. Nagappan *et al.*, "Improving the CS1 experience with pair programming," *SIGCSE Bull.*, vol. 35, no. 1, pp. 359–362, Jan. 2003, doi: 10.1145/792548.612006.
28. Ö. Demir and S. S. Seferoglu, "A Comparison of Solo and Pair Programming in Terms of Flow Experience, Coding Quality, and Coding Achievement," *Journal of Educational Computing Research*, vol. 58, no. 8, pp. 1448–1466, Jan. 2021, doi: 10.1177/0735633120949788.
29. C. O'Donnell, J. Buckley, A. Mahdi, J. Nelson, and M. English, "Evaluating Pair-Programming for Non-Computer Science Major Students," in *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, Kansas City Missouri USA, Feb. 2015, pp. 569–574. doi: 10.1145/2676723.2677289.
30. H. M. Kattan, F. Soares, A. Goldman, E. Deboni, and E. Guerra, "Swarm or pair?: strengths and weaknesses of pair programming and mob programming," in *Proceedings of the 19th International Conference on Agile Software Development: Companion*, Porto Portugal, May 2018, pp. 1–4. doi: 10.1145/3234152.3234169.
31. M. Shiraishi, H. Washizaki, Y. Fukazawa, and J. Yoder, "Mob Programming: A Systematic Literature Review," in *2019 IEEE 43rd Annual Computer Software and Applications*

Conference (COMPSAC), Milwaukee, WI, USA, Jul. 2019, pp. 616–621. doi: 10.1109/COMPSAC.2019.10276.

32. N. A. Bowman, L. Jarratt, K. Culver, and A. M. Segre, “The Impact of Pair Programming on College Students’ Interest, Perceptions, and Achievement in Computer Science,” *ACM Trans. Comput. Educ.*, vol. 21, no. 3, pp. 1–19, Sep. 2021, doi: 10.1145/3440759.
33. G. D. Kuh, “The National Survey of Student Engagement: Conceptual framework and overview of psychometric properties,” 2001.
34. A. W. Astin, “Student involvement: A developmental theory for higher education,” *Journal of college student personnel*, vol. 25, no. 4, pp. 297–308, 1984.
35. G. F. Burch, N. A. Heller, J. J. Burch, R. Freed, and S. A. Steed, “Student Engagement: Developing a Conceptual Framework and Survey Instrument,” *Journal of Education for Business*, vol. 90, no. 4, pp. 224–229, May 2015, doi: 10.1080/08832323.2015.1019821.
36. J. C. Immekus and P. K. Imbrie, “A Test and Cross-Validation of the Revised Two-Factor Study Process Questionnaire Factor Structure Among Western University Students,” Educational and Psychological Measurement, vol. 70, no. 3, pp. 495–510, Jun. 2010, doi: https://doi.org/10.1177/0013164409355685
37. R. D. Axelson and A. Flick, “Defining student engagement,” Change: The magazine of higher learning, vol. 43, no. 1, pp. 38–43, 2010.
38. B. Macfarlane and M. Tomlinson, “Critiques of student engagement,” *Higher Education Policy*, vol. 30, pp. 5–21, 2017.
39. V. Trowler, “Student engagement literature review,” *The higher education academy*, vol. 11, no. 1, pp. 1–15, 2010.
40. N. D. Myers, S. Ahn, and Y. Jin, “Sample Size and Power Estimates for a Confirmatory Factor Analytic Model in Exercise and Sport: A Monte Carlo Approach,” *Research Quarterly for Exercise and Sport*, vol. 82, no. 3, pp. 412–423, Sep. 2011, doi: 10.1080/02701367.2011.10599773.
41. T. A. Brown, *Confirmatory factor analysis for applied research*, Second edition. New York ; London: The Guilford Press, 2015.
42. M. Alavi, D. C. Visentin, D. K. Thapa, G. E. Hunt, R. Watson, and M. Cleary, “Chi-square for model fit in confirmatory factor analysis,” *J Adv Nurs*, vol. 76, no. 9, pp. 2209–2211, Sep. 2020, doi: 10.1111/jan.14399.