## Title

WIP: Empowering Teachers and Engaging Students Through Flow-Based Music Programming in K-12 CS Education



**Journal**

2025 IEEE Frontiers in Education Conference (FIE), 00

**Authors**

Liu, Zifeng
Zhang, Fan
Barron, Alec
et al.

**Publication Date**

2025-11-05

**DOI**

10.1109/fie63693.2025.11328473

# WIP: Empowering Teachers and Engaging Students Through Flow-Based Music Programming in K-12 CS Education


Zifeng Liu
*University of Florida*
Gainesville, USA
liuzifeng@ufl.edu

Fan Zhang
*University of Florida*
Gainesville, USA
f.zhang1@ufl.edu

Alec Barron
*University of California, San Diego*
San Diego, USA
awbarron@ucsd.edu

Wanli Xing
*University of Florida*
Gainesville, USA
wanli.xing@coe.ufl.edu

Maya Israel
*University of Florida*
Gainesville, USA
misrael@coe.ufl.edu

Victor Minces
*University of California, San Diego*
San Diego, USA
vminces@ucsd.edu



***Abstract*—This research WIP paper explores integrating flow-based music programming (FBP) into K–12 computer science (CS) education, focusing on its impact on teachers' confidence and attitudes, and students' experience and engagement. Despite increasing research on music and CS education, limited empirical studies examine how FBP impacts novice CS teachers and student programming learning. This study investigates FBP's role in supporting teachers' confidence and pedagogical practices while fostering student experience and engagement. It employs a two-phase design: (1) Teacher workshop: A six-hour professional development workshop with ten elementary teachers used a platform for composing music through programming. It included hands-on activities and pedagogical strategies. Pre- and post-surveys measured changes in teachers' confidence, interest, and attitudes toward CS. (2) Classroom implementation: One teacher implemented the curriculum with 28 fourth-grade students. Pre- and post-surveys evaluated changes in students' experience, self-efficacy, interest, and perceived value toward programming. Findings indicate a significant increase in teachers' self-efficacy, interest, and attitudes. Before the workshop, none had CS teaching experience, and six reported low confidence teaching programming. Afterward, all teachers expressed improved confidence teaching programming, troubleshooting issues, and learning new concepts. Their programming interest and perceptions of its professional value also increased. For students, pre- and post-surveys showed statistically significant improvements in experience and attitudes, demonstrating FBP's potential to enhance engagement and learning.**



***Index Terms*—flow-based programming, confidence, professional development, music, computer science.**


## I. Introduction

In recent years, computer science (CS) education has become a critical component of K–12 curricula, equipping students with essential 21st-century skills such as problem-solving, creativity, and critical thinking [1]–[3]. To support broader participation, educators are increasingly turning to interdisciplinary and arts-integrated approaches that can make CS more accessible and engaging [4]. Among these, music-integrated programming is gaining traction for its ability to foster creativity and deepen student engagement. Programming environments that incorporate music not only offer a novel way for students to express themselves but also reveal structural parallels between music and computing concepts [5], [6]. For example, sequences in music mirror computational sequences, loops correspond to musical repetitions, and conditionals can be used to initiate or modify musical elements [7].

Flow-based programming (FBP) offers a compelling entry point for K–12 learners. Its visual interface, for example, where code is represented through interconnected blocks, lowers the barrier to entry for students and teachers with little or no programming experience [8], [9]. FBP's intuitive structure also aligns well with musical composition, it allows users to build interactive soundscapes through creative exploration. This dual focus on logic and expression makes it a promising tool for introducing programming in ways that are both pedagogically effective and emotionally resonant.

Despite growing enthusiasm for integrating music and CS, empirical studies examining how such approaches influence both teachers and students in K–12 settings remain limited. Teachers play a central role in bringing CS into classrooms, yet many elementary school teachers report lacking the confidence, content knowledge, or pedagogical strategies to teach programming effectively [10], [11]. Professional development is therefore essential to help teachers gain the skills and self-efficacy needed to adopt and implement innovative tools like FBP [12]. At the same time, understanding how these tools affect students' attitudes and self-perception in programming is critical to evaluating their broader educational value.

This study investigates how a flow-based music programming environment can support both teachers and students in the context of K–12 CS education. Specifically, it examines how a flow-based platform for composing music through code influences elementary teachers' confidence, attitudes, and interest in CS, as well as students' engagement and self-efficacy in learning programming. A two-phase research design was employed. In Phase 1, a six-hour professional development workshop was conducted with ten elementary school teachers, incorporating hands-on programming with M-flow and strategies for classroom integration. Pre- and post-surveys were used to capture changes in teachers' confidence, interest, and attitudes. In Phase 2, one teacher implemented the curriculum in their fourth-grade classroom with 50 students. Pre- and post-assessments were administered to evaluate shifts in students' interest, attitudes, and self-efficacy in programming.

To guide this study, we posed the following research questions:

1) RQ1: How does participating in a flow-based music programming workshop impact elementary school teachers' confidence, attitudes, and interest in CS education?
2) RQ2: How does a music-based flow programming curriculum impact students' experience, self-efficacy, interest and engagement in learning programming?

## II. Methods

### A. M-flow and aligned curriculum

M-flow is a public flow-based music programming environment that allows users to create sounds by dragging blocks onto a canvas. For example, users can record sounds, trigger drum blocks, create loops, use keyboard strokes, introduce randomness, set alarms, and share their creations. A fourth-grade curriculum aligned with science standards was also developed, consisting of ten lessons. For a more detailed description of M-flow and its curriculum, see *Anonymized*.

### B. Workshop design and implementation

The design of the workshop draws on both Social Cultural Learning Theory [13] and Social Constructivist Learning Theory [14]. These theories provide a robust framework for understanding the learning processes involved in the professional development of teachers. Sociocultural Learning Theory emphasizes the role of social interactions and cultural tools in cognitive development. Key concepts applied in this study include the Zone of Proximal Development (ZPD) and scaffolding, which are reflected in the workshop design that provides structured, scaffolded learning experiences. The M-flow platform serves as a cultural tool that teachers learn to integrate into their teaching practices. Socioconstructivist Learning Theory focuses on collaborative learning and knowledge construction through social interactions [14]. The workshop promotes collaborative learning through group activities and discussions, it allows teachers to share ideas and build collective knowledge.

The workshop was six hours long, and was guided by an expert from [*Anonymized Institution*]. There were 10 teacher participants, with an average class size of 25 students, plus the district's science coordinator. Generally, the teachers had little or no experience teaching programming. The curriculum consists of 10 lessons, in which students learn Mflow in consecutive rounds of free exploration, guided exploration, and discussion. In the workshop, teachers experienced the curriculum as students, after which they discussed their thoughts and how they would implement it in their classrooms.

### C. Classroom implementation

Following the teacher workshop, ten participating teachers began implementing Mflow in their classrooms to engage students in computing and music-integrated learning. This work-in-progress study reports on the classroom implementation of one teacher. The class included 28 fourth-grade students from an elementary school located in the southwestern United States. All students completed a pre-survey, and 25 of them completed the post-survey after participating in the ten-week M-flow curriculum. The implementation took place during regularly scheduled science classes. Over the course of ten lessons, students engaged with a curriculum designed to introduce core computing concepts through music-making activities. The lessons encouraged students to reflect on the process of sound creation and the ways in which sound can convey emotions.

In-class activities were organized around group-based, hands-on exploration. Students collaboratively engaged with the M-flow platform to investigate its features and functionalities. As part of the curriculum, students worked in teams to create user manuals based on their experiences. Each exploration session was followed by a whole-class discussion, during which students shared observations and explained different components of the M-flow system. This explore-discuss cycle promoted student agency and supported a constructivist approach to learning.

### D. Measurements and data collection

Pre- and post-workshop surveys were conducted to investigate the impact of flow-based music programming environments on teachers' self-efficacy, interest, and attitudes toward teaching programming. Table I outlines the specific survey questions, all rated on a five-point Likert scale. For example, the post-survey #1 has "Much more confident/more Confident/Equally confident/Less confident/Much less confident". The pre-workshop survey established a baseline of participants' confidence, interest in programming, and perspectives on teaching programming, while the post-workshop survey assessed changes in these areas, focusing on shifts in confidence, interest, and perceptions following the workshop.

For students, Table II presents the pre- and post-survey questions addressing their experience, self-efficacy, interest, and perceived value. The first two items were measured using a three-point scale, all other items utilized a five-point Likert scale. While many of the survey constructs and items were adapted from prior research (e.g., Kukul et al. [15]; Mason and Rich [16]; Outlier Research & Evaluation [17]), several constructs were revised or newly developed to better align with the goals of this study. These adaptations involved simplifying the language and contextualizing the items based on students' experiences. In addition, four post-survey-only questions were included following the classroom implementation (see Table V), focusing on students' engagement and their experiences using M-flow, e.g., "I want to use M-flow at home."

TABLE I
TEACHER SURVEY QUESTIONS

| Phase | No. | Category | Question Item (Scale: 1-5) |
|---|---|---|---|
| Pre | 1 | Self-efficacy | How confident do you feel about teaching programming? |
| | 2 | | How confident do you feel about addressing the issues you might encounter while learning programming? |
| | 3 | | How effective do you think your teaching of programming has been? |
| | 4 | | How confident do you feel about solving the problems students encounter while learning programming? |
| | 5 | Interest | Are you interested in learning and understanding programming? |
| | 6 | | Are you interested in improving and innovating programming teaching methods? |
| | 7 | Attitude | How important do you think computer science or programming education is for yourself as a teacher? |
| | 8 | | How much do you agree with the following statement: "Learning computer science or programming can benefit my professional development?" |
| Post | 1 | Self-efficacy | After this workshop, do you feel more confident in teaching programming courses? |
| | 2 | | After this workshop, do you feel more confident in addressing the issues students might encounter while learning programming? |
| | 3 | | After this workshop, do you feel more confident in learning new programming concepts on your own? |
| | 4 | Interest | After this workshop, has your interest in learning more about programming increased? |
| | 5 | | After this workshop, are you more interested in integrating programming into your teaching practices? |
| | 6 | | After this workshop, are you more willing to participate in further training or workshops on computer science or programming? |
| | 7 | Attitude | After this workshop, I view programming education as more important for my role as a teacher. |
| | 8 | | After this workshop, I believe more strongly that learning programming can benefit my professional development. |
| | 9 | | After this workshop, I am more convinced of the value of incorporating programming into my curriculum. |
| | 10 | | I recommend this workshop and M-flow curriculum to other teachers. |

TABLE II
STUDENT SURVEY QUESTIONS

| Construct | No. | Pre-survey & Post Survey Question |
|---|---|---|
| Experience | Q1 | Do you know what programming a computer means? (1-3) |
| | Q2 | How often did you program a computer in the past? (1-3) |
| | Q3 | I have programmed a computer to make music and sounds. (1-5) |
| | Q4 | I have programmed computers to do things I want. (1-5) |
| Self-efficacy | Q5 | I am able to learn to program computers to make music and sounds. (1-5) |
| | Q6 | I am able to learn to program a computer to do things I want. (1-5) |
| Interest | Q7 | I want to program at home. (1-5) |
| | Q8 | I enjoy programming computers. (1-5) |
| | Q9 | I would like to study more about programming in the future. (1-5) |
| Perceived Value | Q10 | I want programming to be popular at other schools. (1-5) |
| | Q11 | It is important for me to use programming in my future job. (1-5) |
| | Q12 | I think programmers are cool. (1-5) |

TABLE III
TEACHER SURVEY RESULTS.

| | Pre-survey | | | Post-survey | | |
|---|---|---|---|---|---|---|
| Construct | Item | Mean (SD) | T-test results | Item | Mean (SD) | T-test results |
| Self-efficacy | #1 | 2.30 (0.67) | t = 8.82, p <0.01** Cohen's D = -1.04 | #1 | 4.50 (0.53) | t = 9.00, p <0.001*** Cohen's D = 2.85 |
| | #2 | 3.20 (0.79) | t = 0.80, p = 0.44 | #2 | 5.00 (0.00) | t = 8.51, p <0.001*** Cohen's D = 2.69 |
| | #3 | / | / | #3 | 4.40 (0.52) | t = 8.57, p <0.001*** Cohen's D = 2.71 |
| | #4 | 2.80 (0.79) | t = 0.80, p = 0.44 | / | / | / |
| Interest | #5 | 4.40 (0.70) | t = 6.33, p = 0.001*** Cohen's D = 2.00 | #4 | 4.40 (0.52) | t = 8.57, p <0.001*** Cohen's D = 2.71 |
| | #6 | / | / | #5 | 4.60 (0.52) | t = 9.80, p <0.001*** Cohen's D = 3.10 |
| | / | / | / | #6 | 4.50 (0.85) | t = 5.58, p <0.001*** Cohen's D = 1.77 |
| Attitude | #7 | 4.30 (0.67) | t = 6.09, p <0.001*** Cohen's D = 1.93 | #7 | 4.60 (0.52) | t = 9.80, p <0.001*** Cohen's D = 3.10 |
| | #8 | 3.30 (1.49) | t = 0.63, p = 0.54 | #8 | 4.80 (0.42) | t = 13.50, p <0.001*** Cohen's D = 4.27 |
| | / | / | / | #9 | 4.80 (0.42) | t = 13.50, p <0.001*** Cohen's D = 4.27 |
| | / | / | / | #10 | 4.90 (0.32) | t = 19.00, p <0.001*** Cohen's D = 6.01 |

## III. RESULTS

### A. Teachers' Data Results

As shown in Table III[1], the pre-survey results indicate that for question #1, the mean score is significantly lower than the neutral midpoint value of 3 on a five-point Likert scale ($p < 0.01$). In contrast, for question #2, the mean score showed no significant difference compared to the neutral value. Post-workshop, the results for the corresponding post-survey questions revealed a notable increase in confidence. For post-survey questions #1, #2, and #3, the mean scores were all above 4.4, significantly exceeding the neutral midpoint ($p < 0.001$). This suggests that after the workshop, teachers demonstrated markedly higher confidence in teaching programming, addressing challenges their students might face while learning programming, and acquiring new programming concepts.

For teachers' interest, the pre-survey results for question #5 showed a significant difference compared to the neutral midpoint value of 3.0 ($p = 0.001$), indicating that teachers were already interested in programming prior to the workshop. For pre-survey question #6, no responses were recorded due to participants' lack of prior experience in teaching programming. Post-workshop, question #4 continued to demonstrate a significant difference from the neutral value ($p = 0.001$), reflecting sustained interest. For the post-only questions (#5 and #6), the mean scores were both significantly higher than the neutral midpoint of 3.0 ($p < 0.001$ for both). These results suggest that while these teachers have no prior knowledge about CS teaching, the workshop effectively enhanced teachers' interest and confidence in learning and teaching programming concepts.

For teachers' attitudes, the pre-survey results indicate a significant difference from a neutral attitude for question #7 ($p < 0.05$). However, for question #8, there was no significant difference, suggesting teachers did not initially hold a more positive attitude toward this belief ($p = 0.54$). Post-workshop, teachers showed significantly improved attitudes, emphasizing the importance of programming for their role, professional growth, curriculum integration, and recommending the workshop and M-flow curriculum to others. All these post-survey responses showed significant differences compared to a neutral attitude ($p < 0.001$).

[1]*Note: For all question items, the t-test compares the scores with a neutral valence (e.g., 3.0 for a five-point scale).*

TABLE IV
STUDENT SURVEY RESULTS

| Construct | item | pre (n = 28) | post (n = 25) | T-statistics | Effect Size |
|---|---|---|---|---|---|
| Experience | #1 | 2.35 (0.49) | 2.72 (0.46) | 3.464*** | 0.693 |
| | #2 | 1.71 (0.66) | 2.12 (0.67) | 2.864*** | 0.573 |
| | #3 | 3.29 (1.21) | 4.43 (0.84) | 3.864*** | 0.644 |
| | #4 | 3.64 (1.19) | 3.80 (1.15) | 0.137 | 0.027 |
| Self-efficacy | #5 | 4.43 (0.84) | 4.64 (0.57) | 1.100 | 0.220 |
| | #6 | 4.36 (0.95) | 4.36 (0.81) | 0.171 | 0.034 |
| Interest | #7 | 4.07 (1.09) | 3.52 (1.36) | -1.637 | -0.327 |
| | #8 | 4.04 (0.79) | - | - | - |
| | #9 | 3.48 (1.36) | 4.14 (0.93) | 2.641** | 0.528 |
| Perceived Value | #10 | 3.54 (1.26) | 3.72 (1.17) | 1.000 | 0.200 |
| | #11 | 3.61 (0.92) | - | - | - |
| | #12 | 4.35 (0.78) | 3.84 (1.07) | -1.745 | 0.349 |

TABLE V
STUDENT POST SURVEY ONLY QUESTIONS RESULTS

| Question | Mean (SD) | Agreement | T-statistic | Effect Size |
|---|---|---|---|---|
| I want to use M-flow at home. | 3.72 (1.46) | 56% | 2.469** | 0.494 |
| I have used M-flow at home. | 2.00 (1.22) | 16% | -4.082*** | -0.816 |
| I want M-flow to be popular at other schools. | 3.68 (1.11) | 60% | 3.070*** | 0.614 |
| Using Mflow is programming. | 4.04 (0.79) | 72% | 6.586*** | 1.317 |

### B. Children's Data Results

As shown in Table IV, after the classroom implementation., there are improvements of score in terms of all four items in students' experience construct, three of these improvements were significant compared with the pre survey results ($p < 0.001$). For self-efficacy, both two items (#5 and #6) are improved, although the improvements compared to the pre-survey were not significant. For interest, the #7 shows a little bit decrease, but the #9 showed significantly improved compared to pre-survey ($p < 0.01$). For perceived value, student show increased score about #10, however, no significant improvements were found from compared t-test. For the post-only questions, students' responses were compared against a neutral midpoint value using a one-sample t-test. Three out of the four items showed significantly higher scores than the neutral point ($p < 0.01$), specifically on the items: "I want to use M-flow at home," "I want M-flow to be popular at other schools," and "Using M-flow is programming."

## IV. DISCUSSION

This study underscores the potential of a flow-based music programming environment, in enhancing elementary teachers' confidence, interests, and attitudes toward CS. After a six-hour workshop, teachers reported significant gains in their ability to teach and troubleshoot programming, highlighting the value of integrating hands-on practice with engaging content. By immersing participants in an environment where they could experiment with flow-based programming concepts through creative music tasks, M-flow provided a concrete means of exploring CS fundamentals. This approach aligns with prior research suggesting that accessible, user-friendly programming tools help reduce anxiety and foster a sense of accomplishment [7]. Additionally, employing creative mediums such as music can bolster user's sense of agency, further increasing their motivation to explore computational concepts [9]. Moreover, these findings align with the qualitative reports of teachers' growing appreciation for programming as a powerful medium for creativity, problem-solving, and collaboration. By linking coding with a familiar art form (i.e., music), the workshop grounded an often abstract technical skill in a more tangible and engaging context, thereby supporting teachers' recognition of programming's interdisciplinary utility [9], [18]. Such contextualization has been shown to support shifts from situational interest to more enduring personal interest, as teachers begin to see programming as both meaningful and compatible with their pedagogical practices [19]. Lastly, the workshop reshaped teachers' attitudes about the significance of CS education and its alignment with their professional growth. The majority of participants left with a stronger conviction about the importance of integrating computer science, recognizing it as a critical 21st-century skill set for their students' future [20]. This attitudinal change is vital, as teachers who see clear benefits in adopting new pedagogical methods are more likely to remain resilient in the face of challenges and incorporate technology long-term [21], [22]. By experiencing firsthand how programming and music reinforcement can yield both technical and creative skill development, teachers discovered new ways to enrich their classrooms and elevate student engagement. Together, these positive shifts in self-efficacy, interest, and attitudes show the importance of well-designed, interdisciplinary professional development initiatives like M-flow, setting the stage for broader and more sustained adoption and integration of CS in elementary education.

The results also suggest that M-flow effectively enhanced students' perceptions of engaging in computer programming, particularly in terms of their reported experiences. Significant improvements in three out of four experience-related items indicate that students felt more involved in programming-related tasks after the classroom implementation. Although the increase in self-efficacy scores was not statistically significant, the upward trend is encouraging and may suggest potential for longer-term growth with extended exposure. Despite these promising outcomes, students' intentions to pursue a career in computer programming did not change. This result is not unexpected, given that the curriculum did not explicitly address programming-related career paths—an area where younger learners often have limited awareness. Previous research has noted that fostering career intentions in young students is particularly challenging and typically requires sustained interventions that highlight real-world applications and professional opportunities [7], [11], [23]. Regarding perceived value, students expressed stronger agreement with the idea that programming should be popular at other schools, although this improvement did not reach statistical significance. This may indicate a general appreciation for the tool and its broader applicability, even if not strongly felt at an individual level.

The post-only survey responses further support the positive reception of M-flow, including students' desire to use M-flow at home, their endorsement of its popularity in other schools, and their recognition that "Using M-flow is programming." These findings suggest that students not only enjoyed the experience but also developed a meaningful understanding of the nature of programming through the platform.

Future work will extend the research to include classroom implementation of the M-flow curriculum. With approximately 200–300 students taught by these ten teachers, future studies can evaluate how flow-based music programming influences student engagement, learning outcomes, and attitudes toward programming. By observing real-world applications in diverse classroom settings, the research can provide a deeper understanding of the curriculum's effectiveness and scalability. Additionally, longitudinal studies could explore how sustained teacher training and curriculum use further impact both teacher proficiency and student success in computer science education. Future research will also explore how generative AI tools (e.g., [24]) can enhance this environment and expand its potential applications.

## ACKNOWLEDGMENT

This work was supported by the National Science Foundation's ITEST program (22-585), "Using Flow-Based Music Programming to Engage Children in Computer Science."

## REFERENCES

[1] W. Huang and C.-K. Looi, "A critical review of literature on "unplugged" pedagogies in k-12 computer science and computational thinking education," *Computer Science Education*, vol. 31, no. 1, pp. 83–111, 2021.

[2] Z. Liu, W. Xing, X. Jiao, *et al.*, "What are the differences between student and chatgpt-generated pseudocode? detecting ai-generated pseudocode in high school programming using explainable machine learning," *Education and Information Technologies*, 2025.

[3] Z. Liu, R. Guo, X. Jiao, X. Gao, H. Oh, and W. Xing, "How ai assisted k-12 computer science education: A systematic review," in *Proceedings of the 2024 ASEE Annual Conference & Exposition*, (Portland, Oregon), June 2024.

[4] V. H. Minces and N. Akshay, "Steam for all: a vision for stem and arts integration," in *International Encyclopedia of Education* (R. J. Tierney, F. Rizvi, and K. Erkican, eds.), vol. 11, Elsevier, 2023.

[5] C.-M. Dabu, "Computer science education and interdisciplinarity," in *Science education: Research and new technologies*, pp. 137–151, IntechOpen, 2017.

[6] A. Repenning, J. Zurmühle, A. Lamprou, and D. Hug, "Computational music thinking patterns: Connecting music education with computer science education through the design of interactive notations.," in *CSEDU (1)*, pp. 641–652, 2020.

[7] Y. Song, W. Xing, A. Barron, H. Oh, C. Li, and V. Minces, "M-flow: a flow-based music creation platform improves underrepresented children's attitudes toward computer programming," in *Proceedings of the 22nd Annual ACM Interaction Design and Children Conference*, pp. 233–238, 2023.

[8] J. P. Morrison, *Flow-Based Programming, 2nd Edition: A New Approach to Application Development*. CreateSpace, 2010.

[9] C. Goudouris, A. C. de Abreu Mol, A. P. Legey, P. V. R. de Carvalho, J. L. Freire, B. M. R. Martins, and A. Jatobá, "Applying flow-based principles in teaching computer programming to high school students: A semiotic perspective," *Education and Information Technologies*, vol. 25, no. 6, pp. 5451–5476, 2020.

[10] B. Twarek, "The cs teacher landscape: Results from a nationwide teacher survey." https://www.csteachers.org, 2021. Computer Science Teachers Association.

[11] Z. Liu, S. Zhang, M. Israel, R. Smith, W. Xing, and V. Minces, "Engaging k-12 students with flow-based music programming: An experience report on its impact on teaching and learning," in *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE TS 2025)*, (Pittsburgh, PA, USA), ACM, February 26-March 1 2025.

[12] D. Leyzberg and C. Moretti, "Teaching cs to cs teachers: Addressing the need for advanced content in k-12 professional development," in *Proceedings of the 2017 ACM SIGCSE technical symposium on Computer Science Education*, pp. 369–374, 2017.

[13] L. S. Vygotsky, *Mind in Society: The Development of Higher Psychological Processes*. Cambridge, MA: Harvard University Press, 1978.

[14] J. Roschelle, "Learning by collaborating: Convergent conceptual change," *The Journal of the Learning Sciences*, vol. 2, no. 3, pp. 235–276, 1992.

[15] V. Kukul, Şahin Gökçearslan, and M. S. Günbatar, "Computer programming self-efficacy scale (cpses) for secondary school students: Development, validation and reliability," *Eğitim Teknolojisi Kuram ve Uygulama*, vol. 7, no. 1, pp. 158–179, 2017.

[16] S. L. Mason and P. J. Rich, "Development and analysis of the elementary student coding attitudes survey," *Computers & Education*, vol. 153, p. 103898, 2020.

[17] Outlier Research & Evaluation, "BASICS Study ECS Student Implementation and Contextual Factor Questionnaire Measures [Measurement scales]," tech. rep., Outlier Research & Evaluation at UChicago STEM Education, University of Chicago, Chicago, IL, Sept. 2017.

[18] J. Freeman, B. Magerko, T. McKlin, M. Reilly, J. Permar, C. Summers, and E. Fruchter, "Engaging underrepresented groups in high school introductory computing through computational remixing with earsketch," in *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, pp. 85–90, 2014.

[19] S. Hidi and K. A. Renninger, "The four-phase model of interest development," *Educational Psychologist*, vol. 41, no. 2, pp. 111–127, 2006.

[20] L. Hu, "Programming and 21st century skill development in k-12 schools: A multidimensional meta-analysis," *Journal of Computer Assisted Learning*, vol. 40, no. 2, pp. 610–636, 2024.

[21] M. S. Horn, M. West, and C. Roberts, *Introduction to Digital Music with Python Programming: Learning Music with Code*. Focal Press, 1st ed., 2022.

[22] S. Siva, T. Im, T. McKlin, J. Freeman, and B. Magerko, "Using music to engage students in an introductory undergraduate programming course for non-majors," in *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, pp. 975–980, 2018.

[23] M. Edwin, "Fostering elementary career exploration with an interactive, technology-based career development unit," *Journal of School Counseling*, vol. 16, no. 13, p. 1, 2018.

[24] W. Xing, Y. Song, C. Li, Z. Liu, W. Zhu, and H. Oh, "Development of a generative ai-powered teachable agent for middle school mathematics learning: A design-based research study," *British Journal of Educational Technology*, 2025. Early View, published April 17, 2025.