Implementing EarSketch: Connecting Classroom Implementation to Student Outcomes

Tom McKlin
The Findings Group
125 E. Trinity Place, Ste 249
Decatur, Georgia USA
tom@thefindingsgroup.org

Dana Wanzer
The Findings Group
125 E. Trinity Place, Ste 249
Decatur, Georgia USA
dana@thefindingsgroup.org

Taneisha Lee
The Findings Group
125 E. Trinity Place, Ste 249
Decatur, Georgia USA
taneisha@thefindingsgroup.org

Brian Magerko
Digital Media Program
Georgia Institute of Technology
85 Fifth Street NW
Atlanta, Georgia USA
magerko@gatech.edu

Doug Edwards
Center for Education Integrating Science Mathematics and Computing
Georgia Institute of Technology
Atlanta, Georgia USA
doug.edwards@ceismc.gatech.edu

Sabrina Grossman
Center for Education Integrating Science Mathematics and Computing
Georgia Institute of Technology
Atlanta, Georgia USA
sabrina.grossman@ceismc.gatech.edu

Jason Freeman
Center for Music Technology
Georgia Institute of Technology
840 McMillan Street
Atlanta, Georgia USA
jason.freeman@gatech.edu

ABSTRACT

The expansion of computer science into more classrooms invites researchers and evaluators to shift their focus from predominantly measuring student-level factors to measuring both student- and classroom-level variables. Research presented in this article uses multi-level modeling to study student-level factors within the larger context of classroom-level factors. Specifically, we analyze EarSketch, a collaborative and authentic learning tool, that introduces students to programming through music remixing, has previously been shown to increase student engagement, and increases learners' intentions to persist in computing.

This article presents classroom implementation frameworks commonly used in math and science education but rarely, if ever, applied to computer science. The results from a multi-level modeling analysis show that classroom implementation correlates with students' intentions to persist in computing but may not be related to student attitudes toward computing or content knowledge acquisition. Further analysis reveals that one of the five classroom implementation factors, elaboration, emerges as the most salient. This article triangulates these results with qualitative findings from school administrators and teachers, and the article concludes by theorizing how classroom implementation frameworks may be adapted to meet the unique needs of computer science teachers, learners, researchers, evaluators, and curriculum developers.

SIGCSE '19, February 27-March 2, 2019, Minneapolis, MN, USA

ACM ISBN 978-1-4503-5890-3/19/02...$15.00
https://doi.org/10.1145/3287324.3287379

KEYWORDS: Computer science education, creativity, STEAM, music, broadening participation

ACM Reference format:

Tom McKlin, Dana Wanzer, Taneisha Lee, Brian Magerko, Doug Edwards, Sabrina Grossman, and Jason Freeman. 2019. Implementing EarSketch: Connecting Classroom Implementation to Student Outcomes. In *Proceedings of 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19), February 27-Mar. 2, 2019, Minneapolis, MN, USA.* ACM, NY, NY, USA, 7 pages. https://doi.org/*10.1145/3287324.3287379*

1 INTRODUCTION

The expansion of computer science curricula into more classrooms requires that evaluators and researchers study how teachers implement curricula. This paper proposes that we incorporate classroom implementation, the study of enacted curriculum, into research and evaluation designs as outcome variables for professional learning and predictors of student success. This paper describes EarSketch, the EarSketch curriculum, how it was enacted by 18 computer science teachers, the variation across teachers, and how that variation may affect student success.

1.1 EarSketch

EarSketch seeks to engage diverse student populations in introductory computer science by emphasizing the personally expressive role of computing in the domain of music. EarSketch students learn elements of computing and sample-based music composition (i.e. composition using musical beats, samples, and effects). They write Python or JavaScript code to algorithmically create music in popular genres and use fundamental computing concepts such as loops, lists, and user-defined functions to manipulate musical samples, beats, and effects.

EarSketch is a web-based learning platform that includes a Python and JavaScript code editor with both text and blocks-based modes;

a multi-track digital audio workstation view that shows the musical results of code execution; an audio loop library with 4000+ musical clips in a variety of popular music genres that serve as building blocks for new musical compositions; and an inline curriculum, with student and teacher-facing components, that is closely aligned with Computer Science Principles [8].

The design of EarSketch leverages music's potential as a hook to engage students in computing, as well as the natural conceptual parallels between the two disciplines. It affords an "immediate opportunity to act" [5] in that students with no prior musical or computational background can quickly become musically expressive, writing a few lines of code to assemble audio clips from the loop library. EarSketch is also perceived to be authentic [19][25] by students; it utilizes industry-relevant programming languages, borrows heavily from music production paradigms in its user interface and API, and incorporates popular musical styles and content created by well-known engineers and producers.

1.2 EarSketch Curriculum

The EarSketch curriculum is aligned with the programming standards of the College Board's Advanced Placement (AP) Computer Science Principles (CSP) course, as well as a related (non-AP) Computer Science Principles course that is a standard high-school course in the state of Georgia. The course introduces students to the creative aspects of programming, abstractions, algorithms, large data sets, the Internet, cybersecurity, and the impacts of computing across multiple domains [2]. AP CSP does not mandate a specific programming language or problem domain: students submit performance tasks created with a programming language or environment of their choice, and they take a language-agnostic end-of-course exam.

The EarSketch curriculum for CSP consists of a 10-to-12-week module that covers the CSP learning objectives for programming and many of the objectives for creativity, abstraction, and algorithms. We have also provided scaffolding and support for teachers in three areas: a) teaching materials that include day-by-day lesson plans, slides, worksheets, mini-tasks, videos, project descriptions and rubrics, assessments, and integration guides; b) face-to-face and online professional development that introduces teachers to EarSketch, the curriculum, and new pedagogical practices such as studio-based learning; and c) a community where teachers can ask questions, share materials, and review additional training resources in both an online website and a series of in-person events.

The EarSketch CSP module is organized into three units that have teacher lesson plans based on the 5E instructional model. Bybee et al. [4] describe the 5E model as a series of instructional phases designed to help learners better understand scientific and technical knowledge, attitude, and skills. The phases of the model are:

1. Engagement: Accessing prior knowledge and helping students become curious about a new topic.
2. Exploration: Experiencing a common set of activities to facilitate the growth of concepts, processes, and skills. This may include lab activities that use prior knowledge to generate new ideas, explore questions, and conduct an investigation.
3. Explanation: Opportunities for students to demonstrate conceptual understanding, process skills, or behaviors. Learners explain their understanding of the concept which guides students toward deeper understanding.
4. Elaboration: Additional activities and new experiences that challenge and extend students' conceptual understanding.
5. Evaluation: Students assess their understanding and abilities and demonstrate progress toward achieving educational objectives.

Further, each unit has an authentic challenge that requires the student to code musical concepts to satisfy the musical and technical criteria of the challenge. For example, the first challenge requires the student to select a client that could be a business in their community or a school organization. The student must develop a short musical introduction for a client advertisement that applies research on how tempo and pitch affect mood. Students share their music and code with their classmates, teacher, and client to see if the intended mood is elicited and also discuss their code. Based on the feedback, they then iterate on their creation to reach a final product. In open-ended projects such as the above challenge, there is no single, correct solution. Students must collaborate and communicate with their classmates, their teacher, and external partners to iteratively refine the project goals, assess work in progress, and devise new musical and computational strategies to address feedback. The EarSketch CSP module follows this studio-based learning (SBL) approach across all three units: a) designing an artifact; b) presenting work to peers and teachers, along with a detailed justification of the decisions made; c) discussing the work of peers and offering feedback; and d) revising work based on feedback [14].

1.3 Theory of Change

The EarSketch team modified its original theory of change [10][19], which only included student-level variables, to account for classroom-level variables (*self-efficacy*, *classroom implementation*, and *teacher CS content knowledge* prior to entering the program). A theory of change is a tool that illustrates how and why a desired change occurs [1]. The updated EarSketch theory of change (see Figure 1) posits that classroom variables combined with a learning environment conducive to music and computing contribute to changes in student attitudes, which then lead to changes in two important student outcomes: increased computer science content knowledge and an intention to persist in future computing education and computing-centric careers.

2 CLASSROOM IMPLEMENTATION

Observations and interviews with EarSketch teachers reveal that they may approach implementation in different ways. Indeed, Kilpatrick [17] cautions that two classrooms using the same curriculum may look very different. The learning activities may vary widely, may afford different learning opportunities for students, and ultimately may produce different learning outcomes. Variation in classrooms using the same curriculum reveals the importance of measuring classroom implementation. Chval et al. [7] recommend a systematic examination of curriculum enactment to determine whether, how, and under what conditions differences in implementation affect classroom activities and student success.

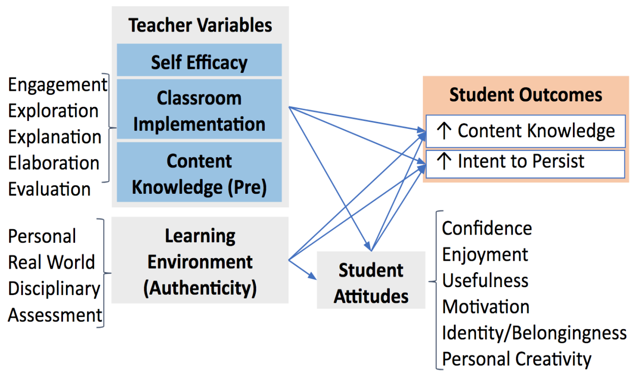

Figure 1. EarSketch Theory of Change

2.1 Consider the Teacher

We must consider the teacher's role in relation to the curriculum and our own perspectives on teaching in order to adequately measure classroom implementation. Implementing a curriculum may have many meanings ranging from the teacher as simply the curriculum deliverer to the teacher as the curriculum co-creator. Snyder et al. [26] offer three conceptualizations of teaching: 'teacher as consumer', 'teacher as shaper', and 'teacher as creator.' Under the **'teacher as consumer'** model, one measures "fidelity," the degree to which the curriculum is delivered as intended to discern factors that facilitate or inhibit intended implementation. The intent of the curriculum designers is central, and some argue that the first step is to interview curriculum authors to understand curriculum design while others discern authors' intent through the curriculum and supporting materials. Mathematics researchers have used two broad approaches to measuring fidelity: some measure fidelity as a binary operation in which the curriculum either is or is not implemented with fidelity while others measure the degree to which the curriculum is implemented as intended. The **'teacher as shaper'** model measures mutual adaptations which are modifications that teachers make to enact the curriculum to meet student, classroom, and school contextual needs. Teachers are 'active implementers' who modify the curriculum to meet their students' needs. An important consideration within the 'mutual adaptation' camp is to discern which adaptations acceptably comply with the authors' original intent and which are unacceptable deviations. In the **'teacher as creator'** model, one measures enactment which is the jointly-created experience of the curriculum among teacher and students. Chval et al. [7] cite Ben-Peretz [3] who describes curriculum development as a two-stage process: the first stage is the authors' conceptualization of the curriculum and support materials; the second stage is the teachers' adaptations, alterations, and translations of these materials to specific students in specific classrooms.

2.2 The Need for Measuring Classroom Implementation in CS Education

The expansion of K-12 Computer Science education in recent years means that researchers may treat the classroom as a unit of analysis. More classrooms [9], the Computer Science Principles framework [2], and numerous curricula make it imperative that researchers consider classroom implementation as a critical variable in program theories of change.

Further, a 2017 empowerment evaluation [12] conducted among CS education evaluators prior to and during the NSF CISE/EHR 2017 meeting revealed six pressing evaluation needs. Two needs are important to this work: CS education evaluators expressed a need to better assess the quality of instructional strategies and to better assess the quality of curriculum materials. Additionally, the NSF-funded Evaluation Working Group collected data across all CS10K projects and found that only 23% report classroom implementation measures [11].

This study presents one initial approach to measuring classroom implementation, and we offer it as an early example upon which to improve the capacity of researchers and evaluators at using classroom implementation as a set of variables to both help explain student success and describe the effects of professional learning and curriculum support materials.

2.3 Approaches to Measuring Implementation

Space constraints prohibit a thorough review of all approaches to measuring implementation; however, the following two approaches offer unique and applicable perspectives. First, Century's [6] framework consists of four broad areas and proposes multiple data sources (observations, focus groups/interviews, teacher and student surveys, enactment checklists, and data from the school's student information system). The first component, *what the curriculum guides teachers to do*, addresses time spent on instruction, the order of lessons, materials used in each lesson, lesson preparation, readings, assessments, and instructional delivery formats required by the intervention. The second, *what teachers know*, includes teacher background knowledge and specific knowledge needed to enact the intervention. The third component, *how teachers interact*, includes pedagogical actions such as facilitating student engagement with the content, with each other, and with the teacher; and facilitating student autonomy, risk taking, and interest. This component focuses on the roles teachers play to facilitate student engagement while the fourth component focuses directly on student engagement. The final component, *how and in what ways students engage*, measures students engaging with each other, with the content, and with instructional materials and activities.

Second, Huntley [16] describes classroom observation as the gold standard and uses the concerns-based adoption model to develop innovation configuration maps. This model [3] posits that instructional behaviors either align with the intent of the curriculum developers or fall outside of the curriculum developer's intent. Innovation configuration (IC) maps are rubrics that describe the degree to which classroom practice is faithful to the curriculum designer's intent. Maps are developed by interviewing curriculum designers to understand what classroom practices are consistent with their intent and which ones are not, which practices are acceptable and which are not. The concerns-based adoption model assumes that an innovation (such as curriculum materials) may take many different forms when implemented, and innovation configuration maps are diagnostic tools that carefully describe the ideal way of implementing an innovation and variations of the ideal. An IC map is a grid in which the rows represent major components of the innovation and the columns represent variations of implementation. The first column represents the ideal implementation; the second column is an acceptable variation of an ideal implementation that still falls within the curriculum envelope. The third and fourth columns describe implementations that fall outside of the acceptable curriculum envelope. For computer science curricula, we might develop an innovation configuration map for major components like Introducing a Concept, Facilitating Pair Programming, or Modeling Programming.

3 METHODS

3.1 Procedures

Participating teachers attended a summer professional learning workshop and received ongoing support during the academic year. The workshop consisted of three day-long, face-to-face sessions combined with a month-long asynchronous online session. These two learning events are designed to prepare teachers to teach the Computer Science Principles programming unit (Big Idea 5). The 18 teachers taught 882 students during the 2017-18 school year. Of those, 473 (53.6%) students consented to participate in the study.

3.2 Assessment Instruments

This study examines the theory of change to discern whether the overall theory holds and to determine how classroom-level variables relate to student attitudes and student success. While the theory of change suggests causality, the statistical approaches below show correlation and cannot prove causality.

3.2.1 Self Efficacy

We administered a 16-item self-efficacy survey to teachers [27] that measures five subconstructs: *Computing Pedagogical Knowledge Self Efficacy*, *Instructional Self Efficacy*, *Engagement Self Efficacy*, *Disciplinary Self Efficacy*, and *Outcome Expectancy*. The instrument was administered upon entry (pre) into the program and then after teachers finished teaching all EarSketch units (post). For this analysis, we used responses from the post administration.

3.2.2 Classroom Implementation

We administered an enactment checklist to teachers for three lessons over the course of the EarSketch module. The self-report enactment checklist was adapted from two prior projects [13][21] that focused on curriculum development and implementation. The enactment surveys were constructed as an efficient method to measure curriculum implementation with multiple teachers in diverse geographic settings. The self-reported enactment checklist was used during curriculum implementation and supported by teacher interviews and observations. Data from the checklist aligned with teacher responses from interviews, encouraging continued use of this tool when evaluating enactment on a large scale curricular intervention.

The first step in development of the checklists was to identify the critical components, or essential program elements of the curriculum [20][6]. In EarSketch, these critical components were defined by the learning goals of each lesson. These learning goals were organized on the enactment checklist using the 5E instructional model to match the lesson plan format provided to the teachers. The enactment checklist asks teachers whether and to what degree they used curriculum resources and the percentage of students engaged in each stage of the lesson.

3.2.3 Content Knowledge

We measured computing content knowledge for both students and teachers using a 20-item, language agnostic instrument developed by the EarSketch team with support from an Advanced Placement Computer Science teacher. The assessment is aligned with the College Board's AP CSP Framework [8] and measures objectives 5.1.1-5.4.1 (Big Idea 5: Programming). We administered the assessment as a traditional pre-post test to both students and teachers. Students took the test before and after the EarSketch units; teachers took the test during professional learning and again after teaching the EarSketch units. The assessment addresses the essential knowledge areas appropriate for multiple-choice format. The team also conducted think-aloud interviews with high school computer science students to verify that students accurately interpret the items. Findings were also triangulated through six focus groups with a representative sample of students.

3.2.4 Learning Environment and Student Attitudes

The EarSketch student survey measures students' perceptions of the learning environment (whether they perceive it to be authentic to making music and to computing) along with attitudes (*confidence*, *enjoyment*, *importance*, *motivation*, *identity*, *intent to persist*, and *personal creativity*). See Engelman et al. [10] and McKlin et al. [19] for more on the student survey, particularly *authenticity* and *personal creativity*. The student survey contains subscales from the Computer Science Attitude Survey (CSAS) [28] and the Computer Attitude Questionnaire (CAQ) [18]. The student survey was administered at the end of the unit as a retrospective pre-post, except for the authenticity scale which was administered as a post-only scale. (Students cannot rate the *authenticity* of a learning environment prior to experiencing the learning environment.)

3.2.5 Qualitative Analyses

Teachers and administrators participated in 30 – 60 minute semi-structured interviews after teaching the EarSketch module. Interviews were audio recorded and transcribed verbatim. The first cycle of coding involved creating attribute (school district, number of years teaching CSP, etc.) and structural codes [24]. Structural codes identify content-based phrases related to implementation, decision making in selecting CS curriculum and courses, experiences with EarSketch, differing interest and engagement by subgroups, factors that enhance teaching, and types of support needed to teach CSP with EarSketch. During the second cycle of coding, pattern coding was used to develop major themes.

3.3 Multi-Level Modeling

We used multilevel modeling (MLM) to answer the question: How do classroom-level variables (*teaching efficacy, classroom implementation, and content knowledge*) and student perceptions of the learning environment relate to student attitudes and student outcomes? Multilevel modeling (MLM) is a statistical approach that accounts for grouped data [15]. This is important to use because individuals in a group tend to be more similar than individuals across groups. This study has two levels of data: students (level 1 data) are grouped within classrooms (level 2 data). *Teaching efficacy, classroom implementation,* and *teacher content knowledge (pre)* are classroom-level data (level 2 data) while student ratings of the *learning environment (authenticity)*, the six student attitudinal constructs, and student outcomes (*content knowledge* and *intent to persist*) are student-level data (level 1 data). We performed three MLMs: one for each of the three outcomes (*student attitudes*, *student content knowledge*, and *intent to persist*). The first step is to determine what percentage of the outcome at the student-level is predicted by classroom-level variables. The second step includes all student-level predictors. The third step adds the classroom-level predictors to see how well they predict the outcome above and beyond the student-level predictors.

4 RESULTS

4.1 Percentage of Each Outcome Predicted by Classroom Variables

The first step is determining what percentage of the outcome at the student-level is explained by the grouping structure of the data by calculating the Intraclass Correlation Coefficient (ICC). These analyses essentially run a simple regression with only the grouping variable as a predictor of each of the three outcomes. The ICC values indicate 21% of the variation in student attitudes, 32% of the variation in student content knowledge, and 13% of the variation in students' intention to persist is explained by classroom membership. These findings indicate that a significant portion of the variation in student outcomes is explained by the students' classroom. The following analyses examine which student- and classroom-level variables contribute to the variation in student outcomes.

4.2 Student- and Classroom-Level Predictors of Project Outcomes

Table 1 shows the results of three MLM analyses. Each analysis uses a different outcome variable (*student attitudes, student content knowledge, and students' intention to persist*). Each analysis contains two models: model 1 uses student-level variables to predict the outcome variable, and model 2 adds classroom-level variables to the student-level variables. For example, the first MLMs in Table 1 use student attitudes as the outcome variable: Model 1 shows that *authenticity* (student perceptions that the learning environment is authentic) and attitudes toward computing upon entering the program both significantly predict post student attitudes. Model 2 adds classroom-level variables and shows that they do not significantly predict student attitudes over and above the student-level variables. The same is true for student content knowledge; adding classroom-level variables does not significantly improve the model's ability to explain content knowledge growth. However, two classroom-level variables (teacher scores on the content knowledge assessment at pre and the composite classroom implementation score) significantly predict students' intention to persist in computing.

Table 1: MLM Results Across Student Outcomes

Student Attitudes (post)	**Model 1**	**Model 2**
Student-level variables		
Authenticity	**.275 (.04)***	**.220 (.04)***
Attitudes (Pre)	**.591 (.04)***	**.635 (.04)***
Classroom-level variables		
Self-efficacy		-.047 (.06)
CKA (Pre)		.109 (.06)+
Implementation		.089 (.06)+
Content Knowledge (Post)	**Model 1**	**Model 2**
Student-level variables		
Authenticity	.065 (.05)+	.031 (.05)
Attitudes (Pre)	.048 (.05)+	.014 (.05)
CKA (Pre)	**.585 (.05)***	**.594 (.07)***
Classroom-level variables		
Self-efficacy		.034 (.11)
CKA (Pre)		-.004 (.13)
Implementation		-.028 (.11)
Intent to Persist (Post)	**Model 1**	**Model 2**
Student-level variables		
Authenticity	**.156 (.04)***	**.119 (.04)***
Attitudes (Pre)	-.028 (.07)	.013 (.07)
Intent to Persist (Pre)	**.717 (.07)***	**.738 (.07)***
Classroom-level variables		
Self-efficacy		-.049 (.04)
CKA (Pre)		**.094 (.04)***
Implementation		**.100 (.04)***

Note: Values shown are the Standardized Beta (standardized error) coefficients. * indicates p < .05; + indicates p < .10

4.3 Classroom Implementation Factors

Given the significant relationship between classroom implementation and students' intention to persist in computing, we further asked which aspects of classroom implementation (i.e., engagement, exploration, explanation, elaboration, evaluation), if any, correlate with students' intentions to persist. Table 2 shows the results of an MLM analysis extending beyond Model 2 of Table 1 for Intent to Persist. It included all student-level and classroom-level variables, but instead of including Implementation as a single variable, it includes the 5Es of implementation individually. Results indicate that elaboration (i.e., additional activities and new experiences that challenge and extend students' conceptual understanding and skills) is the most salient factor related to students' intent to persist.

Table 2: Elaboration is the Most Salient Classroom Implementation Factor

Classroom Implementation Factor	Intent to Persist Model 3
Engagement	-.013 (.04)
Exploration	-.043 (.06)
Explanation	.028 (.07)
Elaboration	**.061 (.03)***
Evaluation	-.067 (.05)+

Note: This model also includes student-level variables and the other classroom-level variables (teacher self-efficacy, teacher CKA at pre) but they are not shown here as values are similar to those from Table 1 Model 2 for Intent to Persist. * indicates $p < .05$ + indicates $p < .10$

4.4 Results from teacher and administrator interviews

Having found a significant relationship between classroom implementation and students' intentions to persist and further finding that elaboration is a critical component of classroom implementation, we sought to understand this relationship through interviews with school administrators and teachers. The following administrator observed a class as students completed their mini-tasks, EarSketch elaboration projects that extend students' conceptual understanding in CS and music.

[A]s a district administrator, how we kind of view the effectiveness of a curriculum is if you can walk in and ask the students 'What are you learning? Why are you learning it? How does it apply to this class?' During the EarSketch modules, they've been very good at explaining what they're doing, and why they're doing it, and how it applies to computer science.

A first year EarSketch and experienced CSP teacher shared an experience with students creating and sharing their mini-tasks.

"*they're able to create something. It's theirs. They can show it to their friends. A lot of them like that aspect of it, that they're able to show them...They will say, "Hey, this is what I made in EarSketch. This is similar to what you just made in GarageBand." They can sit there and actually have a conversation about that, and that's not something they were not doing before [EarSketch]. So, to have them really engage in that way...They can see the benefit of what they're learning, that real-world connection right away. I think that's what's so beneficial with using EarSketch, because they can see it.*

Qualitative findings also support the relationship between implementation and students' intent to persist in future computing education. The following first year EarSketch and experienced CSP teacher shared how using EarSketch resulted in a larger number of students enrolling in the next-level computing course, AP Computer Science.

As a result of us using EarSketch, they're a lot more confident, and many of them have signed up for AP Computer Science when they would not have before...Because now they feel like, Yeah, I can do this. I'm not afraid of programming. I'm not afraid of doing an actual language. So, I actually have, besides the seniors which I do have which is only like 4, the rest of the class is underclassmen, and the majority of them have signed up for AP Computer Science.

When asked about student interest in computer science careers, teachers shared that the EarSketch course may motivate students to gather information about related post-secondary education and careers in computer science.

I have some students who, because we did EarSketch, they've researched Georgia Tech and their Music Technology Department. So, I did have some kids do research there. I've had some kids who say, "Hey, [teacher's name], how can I get other skills? How can I do other things outside of EarSketch, but still learn programming?" So, I've had some kids who have looked online and done some research on how to take additional computer science classes and coding classes.

I have a lot of kids in my CSP courses. I even have some in my intro course in Web Design who say, "Hey, this is what I want to do. What schools should I be looking at? What jobs are available if I want to go this route?" Most of them want to know how much money they can make.

5 DISCUSSION

This paper presents an initial approach for measuring classroom implementation. This work indicates that classroom level variables explain a large enough portion of the variation in student outcomes to necessitate further study, finds that classroom level variables affect students' intention to persist in computing, and shows that one element of classroom implementation stands out: elaboration (additional activities and new experiences that challenge and extend students' conceptual skills). Qualitative findings from administrators and teachers describe students who appear more capable of associating their work with real-world disciplinary endeavors in computing and music and appear motivated to pursue further high-school and post-secondary education in computing and music technology.

We also base our classroom implementation on the 5E model of instruction. While initially reasonable, we perceive that a better approach would be to incorporate a framework that is more sensitive to classroom-based phenomena. We perceive that Century's four-stage model or Huntley's CBAM approach might be more sensitive to the shifts in classroom implementation because these models are tied to the observable enactment of the curriculum rather than lesson plans. Two new instruments [22] [23] have emerged from Century's work that offer promise for the measurement of classroom implementation in future studies.

6 ACKNOWLEDGEMENTS

The EarSketch project receives funding from the National Science Foundation (1417835 and 1649671). EarSketch is available online at earsketch.gatech.edu.

7 REFERENCES

[1] Annie E. Casey Foundation and Organizational Research Services. 2004. Theory of Change: A practical tool for action, results and learning. Retrieved March 3, 2016 from: http://www.aecf.org/m/resourcedoc/aecf- theoryofchange-2004.pdf

[2] Astrachan, O. et al. (2011). CS principles: piloting a new course at national scale. Proceedings of the 42nd ACM technical symposium on Computer science education (New York, NY, USA, 2011), 397–398.

[3] Ben-Peretz, M. (1990). The Teacher-Curriculum Encounter: Freeing Teachers from the Tyranny of Texts. Albany, NY: SUNY Press.

[4] Bybee, R., Taylor, J., Gardner, A., Van Scotter, P., Powell, J., Westbrook, A., & Landes, N. (2006). *The BSCS 5E Instructional Model: Origins and Effectiveness.* Colorado Springs, CO: BSCS. Retrieved from https://bscs.org/sites/default/files/_legacy/BSCS_5E_Instructional_Model-Full_Report.pdf

[5] Carroll, J. 1998. Minimalism Beyond the Nurnberg Funnel. MIT Press.

[6] Century, J., Rudnick, M., & Freeman, C. (2010). A Framework for Measuring Fidelity of Implementation: A Foundation for Shared Language and Accumulation of Knowledge. American Journal of Evaluation, 31(2), 199–218. https://doi.org/10.1177/1098214010366173

[7] Chval, K., Wilson, L. D., Ziebarth, S., Heck, D. J., & Weiss, I. R. (2012). Introduction. In Approaches to Studying the Enacted Mathematics Curriculum (pp. 1–18). Charlotte, N.C: Information Age Publishing.

[8] College Board. 2016. AP Computer Science Principles Curriculum Framework Accessed March 3, 2016 from https://secure-media.collegeboard.org/digitalServices/pdf/ap/ap-computer-science- principles-course-and-exam-description.pdf

[9] College Board Communications. (2018, June 12). AP Computer Science Principles participation grows by over 50% in Year Two. Retrieved from https://www.collegeboard.org/membership/all-access/counseling-admissions-academic/ap-computer-science-principles-participation

[10] Engelman, S., Magerko, B., McKlin, T., Miller, M., Edwards, D., & Freeman, J. (2017). Creativity in Authentic STEAM Education with EarSketch (pp. 183–188). ACM Press. https://doi.org/10.1145/3017680.3017763

[11] Evaluation Working Group. (2017). CS10K Common Data Elements: Year 3 Common Data Collection Report (2016-2017). Amherst, MA: SageFox Consulting Group. Retrieved from http://sagefoxgroup.com/dev/wp-content/uploads/2018/02/Combined-CS10K-Year-3-Report-2018.pdf

[12] Fetterman, D., Haynie, K., McKlin, T., Ravitz, J., Bates, T., DuBow, W., … Tiffany-Morales, J. (2017, November). *Building a Computer Science and STEM Evaluation Learning Community: Using Empowerment Evaluation and Technology Tools.* Presented at the American Evaluation Association, Washington, D.C.

[13] Gale, J., Grossman, S., Ryan, M., Newton, S., Usselman, M. (2018). Exploring curriculum implementation using self-report enactment checklists. Paper presented at National Association of Research in Science Teaching conference. (Atlanta, GA).

[14] Hendrix, D. et al. 2010. Implementing studio-based learning in CS2. Proceedings of the 41st ACM technical symposium on Computer science education (New York, NY, USA, 2010), 505–509.

[15] Hox, J. J. (2002). Multilevel analysis: Techniques and applications. Mahwah, NJ: Erlbaum.

[16] Huntley, M. A. (2012). Using concerns-based adoption model theory to develop tools to examine variations in mathematics textbook implementation. In D. J. Heck, K. B. Chval, I. R. Weiss, & S. W. Ziebarth (Eds.), Approaches to studying the enacted mathematics curriculum. Charlotte, N.C: Information Age Publishing.

[17] Kilpatrick, J. (2003). What works? In S. L. Senk & D. R. Thompson (Eds.), Standards-based school mathematics curricula: What are they? What do students learn? (pp. 471–488). Mahwah: Erlbaum.

[18] Knezek, G. and Christensen, R. 1996. Validating the Computer Attitude Questionnaire (CAQ). Presented at Annual Meeting of the Southwest Education Research Association (New Orleans, Louisiana, January 26, 1996). AMSERA'96.

[19] McKlin, T., Magerko, B., Lee, T., Wanzer, D., Edwards, D., & Freeman, J. (2018). Authenticity and Personal Creativity: How EarSketch Affects Student Persistence (pp. 987–992). ACM Press. https://doi.org/10.1145/3159450.3159523

[20] Mowbray, C. T., Holter, M. C., Teague, G. B., & Bybee, D. (2003). Fidelity Criteria: Development, Measurement, and Validation. American Journal of Evaluation, 24(3), 315–340. https://doi.org/10.1177/109821400302400303

[21] Newton, Sunni H.; Gale, Jessica; Alemdar, Meltem; and Wind, Stefanie. (2018) "Assessment of Practice-Focused Middle School Science Modules". Interdisciplinary STEM Teaching & Learning Conference. (Savannah, GA) https://digitalcommons.georgiasouthern.edu/stem/2018/2018/37

[22] Outlier Research & Evaluation (September, 2017). BASICS Study ECS Student Implementation and Contextual Factor Questionnaire Measures [Measurement scales]. Chicago, IL; Outlier Research & Evaluation at UChicago STEM Education | University of Chicago. Retrieved from http://outlier.uchicago.edu/basics/resources/Measures- StudentImplementation/

[23] Outlier Research & Evaluation (September, 2017). BASICS Study ECS Teacher Implementation and Contextual Factor Questionnaire Measures [Measurement scales]. Chicago, IL; Outlier Research & Evaluation at UChicago STEM Education | University of Chicago. Retrieved from http://outlier.uchicago.edu/basics/resources/Measures- TeacherImplementation/

[24] Saldaña, J. (2013). The coding manual for qualitative researchers. Thousand Oaks, CA: Sage

[25] Shaffer, D.W., and Resnick, M. 1999. Thick Authenticity: New media and authentic learning. Journal of Interact. Learn. Res. 10, 2, 195-215.

[26] Snyder, J., Bolin, F., & Zumwalt, K. (1992). Curriculum implementation. In *Handbook of research on curriculum: A project of the American Educational Research Association* (pp. 402–435). New York, NY: Macmillan.

[27] Yoon Yoon, S., Evans, M. G., & Strobel, J. (2014). Validation of the Teaching Engineering Self-Efficacy Scale for K-12 Teachers: A Structural Equation Modeling Approach: Validation of the Teaching Engineering Self-Efficacy Scale. *Journal of Engineering Education, 103*(3), 463–485. https://doi.org/10.1002/jee.20049

[28] Wiebe, E., Williams, L., Yang, K., and Miller, C. 2003. Computer science attitude survey. Comput. Scie. 14.

[29] American Psychological Association. (2010). *Publication manual of the American Psychological Association* (6th ed). Washington, DC: American Psychological Association.