

Understanding the Effects of Integrating Music Programming and Web Development in a Summer Camp for High School Students

Daniel Manesh
danielmanesh@vt.edu
Virginia Tech
Blacksburg, VA, USA

Andrew Jelson
jelson9854@vt.edu
Virginia Tech
Blacksburg, VA, USA

Emily Altland
emilya99@vt.edu
Virginia Tech
Blacksburg, VA, USA

Jason Freeman
jason.freeman@gatech.edu
Georgia Institute of Technology
Atlanta, GA, USA

Sang Won Lee
sangwonlee@vt.edu
Virginia Tech
Blacksburg, VA, USA

Abstract

This poster presents the development and implementation of a 10-day remix-based summer camp curriculum designed to introduce high school students, particularly a multinational cohort of young women, to programming through creative coding. The curriculum integrates music composition using EarSketch and web development with HTML and CSS. The camp aims to inspire participants to gain self-efficacy in programming and motivate them to explore STEM/computing careers. Preliminary results from surveys and interviews indicate increased confidence in programming skills. This ongoing research explores the impact of remixing as a gateway for transitioning into more general-purpose computing domains such as web development.

ACM Reference Format:

Daniel Manesh, Andrew Jelson, Emily Altland, Jason Freeman, and Sang Won Lee. 2025. Understanding the Effects of Integrating Music Programming and Web Development in a Summer Camp for High School Students. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 2 (SIGCSE TS 2025)*, February 26-March 1, 2025, Pittsburgh, PA, USA. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3641555.3705275>

1 Introduction and Background

Using culturally relevant, non-traditional topics to engage learners has become a popular approach in programming education [6]. One example is EarSketch, a platform that teaches programming and computational thinking skills through music creation, where students write code to create music using pre-existing audio samples [3]. A key finding from studies on EarSketch is its effectiveness as a gateway for introducing beginners to programming [5]. However, an important question remains: does EarSketch effectively motivate learners to transition to more general-purpose programming domains with practical applications, such as web development or data science? Previous research on programming environments designed to engage children and adolescents, such as Scratch, has

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SIGCSE TS 2025, February 26-March 1, 2025, Pittsburgh, PA, USA

© 2025 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-0532-8/25/02

<https://doi.org/10.1145/3641555.3705275>

shown mixed results regarding their ability to facilitate the transition to general-purpose programming [1, 7]. Similarly, while music-based programming can successfully engage students, there is limited understanding of the extent to which such experiences foster a broader interest in other programming and computing areas.

To address this question, we study the authors' recent involvement in a 10-day summer camp curriculum integrating EarSketch programming and web development (HTML/CSS). The camp, TechGirls¹, is designed to empower and inspire young women worldwide to pursue careers in STEM fields. Participants were female high school students ages 15-17, primarily from developing countries². Over the past two years, the authors have conducted the camp each summer, each time with a cohort of 32 students.

We present our ongoing work, which aims to understand how students' experiences learning programming with EarSketch and web programming together influence their self-efficacy in programming and interest in computing-related careers. Our approach includes observations, surveys, and student interviews. We detail the curriculum design and describe the types of data collected. Additionally, we share preliminary findings based on pre-camp and post-camp survey responses and share our plan to further analyze the summer camp results.

2 TechGirls Curriculum

The camp structure is designed to provide extensive hands-on experiences within a full-day schedule. Each day consists of four hour-long sessions that alternate between lectures and hands-on programming tasks. We employed the metaphor of remixing in music, which involves arranging existing sound samples using EarSketch functions to place the samples programmatically [4]. This remixing approach can be particularly beneficial for beginners, as it allows them to create engaging music without needing a background in music theory. Moreover, the concept of remixing aligns well with programming, where repurposing and reusing existing code is considered a key aspect of computational thinking [2].

For the EarSketch portion of the camp, participants engaged in three types of activities: (1) pair programming, (2) individual projects where they created original music using JavaScript, and (3)

¹TechGirls is an initiative of the US Department of State's Bureau of Educational and Cultural Affairs, implemented by Legacy International in partnership with Virginia Tech.

²For a list of countries eligible for the program, see: <https://techgirlsglobal.org/>

a remixing activity in which they modified and enhanced another student’s code. We emphasized that the remix in (3) could take the form of a ‘musical remix’ involving the rearrangement of existing sounds or the addition of new ones, or a ‘code remix’ which involved commenting on or rewriting the code to improve its clarity or functionality (e.g., using functions, if-statements, or for-loops).

In the second half of the camp, we introduced the students to basic web programming concepts, such as HTML and CSS. For their final project, students were asked to create a website, either individually or in teams, using one of several provided website templates as a starting point. Throughout the camp, we emphasized the connection between the two activities: just as they reused audio samples to create new music, they could adopt the same remixing approach in programming by reusing functions and HTML templates. This approach encouraged them to explore ideas in their final projects, empowering them to exercise agency in creating unique outcomes.

3 Data Collection

In the second year of the camp, we collected various qualitative and quantitative data. A total of 32 female students in the age range of 15-17 participated in the camp. We asked participants to complete a survey before and after the camp to gather self-reported measures across four dimensions, each consisting of three 5-point scale questions: perceived significance of programming, self-efficacy, career interests in computing, and the relevance of programming to their careers, adapting existing questionnaires [5]. We also conducted 10-15 minute interviews with each participant to understand the positive experiences and challenges they encountered during the camp. We also gathered all the code and individual project outcomes from both the EarSketch and web programming portions. The entire data collection protocol was reviewed and approved by Virginia Tech’s internal review board. Furthermore, we obtained parental consent and participant assent for all 32 students involved.

4 Preliminary Results

The notion of remixing inherent in EarSketch proved particularly helpful for enabling beginners to create more advanced projects than they might have otherwise achieved. For their EarSketch pieces, students without prior musical experience could leverage existing audio samples to compose songs that they were eager and often excited to present to their classmates. One challenge we encountered was the varying levels of programming experience among students prior to the camp. Nevertheless, EarSketch was new to all participants and successfully introduced a unique creative challenge through music composition, even for those with extensive programming experience.

An additional benefit of using EarSketch was its flexibility and the absence of rigid evaluation criteria. Unlike conventional programming tasks with fixed goals (e.g., passing test cases), the outcome (a piece of music) could not be deemed right or wrong. Instead, students composed music based on their preferences and cultural influences, fostering an environment where individual differences were respected and served as sources of inspiration. Further analysis of the students’ code is needed to assess how effectively they

applied programming concepts taught during the camp, such as abstraction, if statements, for-loops, and code reuse.

In the website-building activity, students with little prior web development experience were able to learn what we taught. According to our observation, their experience writing code in EarSketch seemed to familiarize them with the syntax of programming languages, using programming editors, debugging when program outcome did not match their expectations, and helping them understand the importance of following rules.

We compared the pre-post survey responses and found a significant difference in self-efficacy, showing an increase of self-efficacy in their programming based on the t-test results (Pre-camp: $\mu = 2.46$, $\sigma = 0.93$; Post-camp: $\mu = 2.91$, $\sigma = 1.00$, $p = 0.005$). We could not find any statistical significance in other dimensions.

5 Ongoing Efforts

Our preliminary results and observations suggest that the summer camp curriculum supported students in creating projects in both domains—music and web development—and helped them gain more confidence in programming. We plan to validate our findings by conducting a more in-depth analysis of the data we collected to answer our research questions. Our first step will be to transcribe and analyze the interviews conducted with each camper from the most recent cohort. Additionally, we will examine the code they submitted for the two projects to assess how effectively they applied the computational concepts they learned and how the code quality related to their prior programming backgrounds.

References

- [1] Michal Armoni, Orni Meerbaum-Salant, and Mordechai Ben-Ari. 2015. From Scratch to “Real” Programming. *ACM Trans. Comput. Educ.* 14, 4, Article 25 (Feb. 2015), 15 pages. <https://doi.org/10.1145/2677087>
- [2] Sayamindu Dasgupta, William Hale, Andrés Monroy-Hernández, and Benjamin Mako Hill. 2016. Remixing as a Pathway to Computational Thinking. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing* (San Francisco, California, USA) (CSCW ’16). Association for Computing Machinery, New York, NY, USA, 1438–1449. <https://doi.org/10.1145/2818048.2819984>
- [3] Jason Freeman, Brian Magerko, Tom McKlin, Mike Reilly, Justin Permar, Cameron Summers, and Eric Fruchter. 2014. 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*. 85–90.
- [4] Tom McKlin, Lauren McCall, Taneisha Lee, Brian Magerko, Michael Horn, and Jason Freeman. 2021. Leveraging Prior Computing and Music Experience for Situational Interest Formation. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (SIGCSE ’21)*. Association for Computing Machinery, New York, NY, USA, 928–933. <https://doi.org/10.1145/3408877.3432431>
- [5] 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 the 50th ACM Technical Symposium on Computer Science Education (SIGCSE ’19)*. Association for Computing Machinery, New York, NY, USA, 634–640. <https://doi.org/10.1145/3287324.3287379>
- [6] Jessica Morales-Chicas, Mauricio Castillo, Ireri Bernal, Paloma Ramos, and Bianca L. Guzman. 2019. Computing with relevance and purpose: A review of culturally relevant education in computing. *International Journal of Multicultural Education* 21, 1 (2019), 125–155.
- [7] Dale Parsons and Patricia Haden. 2007. Programming osmosis: Knowledge transfer from imperative to visual programming environments. In *Proceedings of The Twentieth Annual NACCC Conference*. Citeseer, 209–215.