

Teachers' Experiences Customizing Scratch Instructional Resources with an Interactive AI-Powered Tool

Minh Tran
ngminhtran@uchicago.edu
University of Chicago
Chicago, IL, United States

Nitzan Koren
nkoren@unc.edu
University of North Carolina
Chapel Hill, NC, United States

David Gonzalez-Maldonado
dagm@uchicago.edu
University of Chicago
Chicago, IL, United States

David Weintrop
weintrop@umd.edu
University of Maryland
College Park, MD, United States

Diana Franklin
dmfranklin@uchicago.edu
University of Chicago
Chicago, IL, United States

Abstract

The emergence of generative AI (GenAI) can support computer science (CS) instructors to prepare classroom materials. This study investigates K-12 teachers' experiences using CONJUROR, an interactive tool using text- and image-based GenAI to create customized Scratch materials. We conducted semi-structured and stimulated recall interviews using CONJUROR's automated interaction logs with 25 elementary and middle school teachers (students ages 9-13) after they created and used CONJUROR-customized materials in their classroom. We used thematic analysis to explore (1) how teachers describe their role in the process, (2) how teachers meet instructional design goals, and (3) how the process affects teacher knowledge and teaching opportunities. We found that teachers orchestrated the use of GenAI suggestions to ensure high-quality, locally and culturally relevant materials for their students. By drastically reducing prep time, the tool empowers teachers to integrate interdisciplinary themes, support special needs students, and foster student-centered instruction. Additionally, this human-AI collaboration improved teachers' own CS pedagogical practices and AI literacy, sparking spontaneous classroom discussions on responsible AI use.

CCS Concepts

• **Social and professional topics** → **Computing education**.

Keywords

K-8, generative AI, curriculum customization, teacher tool, Scratch

ACM Reference Format:

Minh Tran, Nitzan Koren, David Gonzalez-Maldonado, David Weintrop, and Diana Franklin. 2027. Teachers' Experiences Customizing Scratch Instructional Resources with an Interactive AI-Powered Tool. In *Proceedings of the 58th ACM Technical Symposium on Computer Science Education V. 1 (to appear)*, February 17-20, 2027, Sacramento, CA, USA. ACM, New York, NY, USA, 7 pages. <https://doi.org/XXXXXXX.XXXXXXX>

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 2027, February 17-20, 2027, Sacramento, CA, USA

© 2027 Copyright held by the owner/author(s).

ACM ISBN 978-1-XXXX-XXXX-X/27/02

<https://doi.org/XXXXXXX.XXXXXXX>

1 Introduction

The expansion in computer science (CS) instruction in primary and secondary schools has led to a growing library of CS curricula for young learners [2, 6, 11, 14, 17]. However, no single curriculum can serve the diversity of students in every classroom. As the experts on their own classroom, teachers are best positioned to customize instructional materials to fit their students' local contexts. However, limited time and formal CS background are major challenges for teachers in doing so effectively [35, 36].

Generative AI (GenAI) has potential to help CS instructors prepare classroom materials [1, 10]. Creating college-level CS instructional resources with GenAI has been well studied [16, 20, 22, 23, 30]. However, the K-12 context requires different attention due to block-based languages and the need for graphics, which are challenging for GenAI to generate [12]. Researchers have evaluated GenAI for helping CS teachers brainstorm ideas for Scratch projects [33]. Yet, to our knowledge, no prior work has investigated K-12 CS teachers' experiences using GenAI to prepare classroom-ready materials.

This paper presents results from a qualitative study involving 25 elementary and middle school teachers who used CONJUROR, an interactive tool using text- and image-based GenAI to customize Scratch instructional resources and teach them in their classrooms. This paper focuses on the experiences of the teachers working with the AI-powered tool, in which answer the following questions:

- **RQ1:** How do teachers describe their roles in the customization process with an interactive AI-powered tool?
- **RQ2:** To what extent do teachers meet their instructional design goals through customizing with an AI-powered tool?
- **RQ3:** How does customizing with an AI-powered tool create or enhance teacher knowledge and teaching opportunities?

2 Background and Prior Work

2.1 Customizing CS Instructional Resources

While there are several fully-featured curricula designed to support the diversity of CS learners in K-12 classrooms [2, 6, 11, 14, 17], there is no one-size-fits-all curriculum. Much work on CS instructional design and curriculum development has relied on expertise outside of the classroom, often led by researchers through research-practitioner partnerships [5, 7, 13, 18, 26]. Teachers, with their localized expertise, are in a unique position to customize curricular resources to resonate with students in their own classroom.

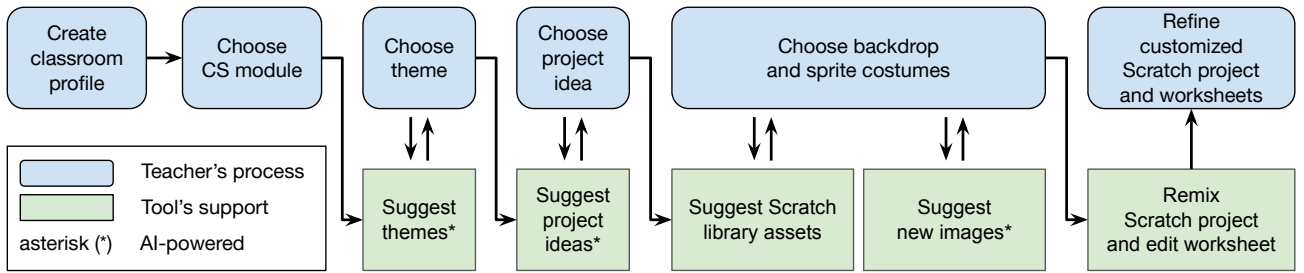


Figure 1: Overview of the process to customize Scratch instructional resources using CONJUROR

Unfortunately, some prior research had shown that customizing curricular resources requires teachers to engage in a time-intensive process with researchers. For example, a study involved K-5 teachers collaborating with researchers in a workshop and follow-up monthly meetings over four months to adapt computing lessons to be more culturally responsive [4]. Another study involved K-8 CS teachers participating in multiple co-design sessions led by researchers over five months to identify challenges and iteratively develop scaffolds for customization [34]. While feasible, this process is challenging to accomplish during the school year, when teachers have many other professional responsibilities and time constraints.

2.2 GenAI for CS Classroom Preparation

With the emergence of GenAI, computing education researchers has explored using GenAI to support *college-level CS faculty* in preparing classroom materials. For example, instructors can use GenAI to create materials for text-based programming languages at scale (e.g., coding assignments [19, 30], code explanations [21, 23], worked examples [20]). Additionally, GenAI can be used to create materials on demand by college students (e.g., personalized Parson puzzles [16] and contextualized programming exercises [8, 22])

In K-12 settings, GenAI has been an intriguing topic [27], yet little research in computing education has explicitly investigated its potential to assist *K-12 CS teachers* in preparing classroom materials. Prior studies found that while GenAI can help teachers brainstorm creative Scratch project ideas, they often suffer from code complexity, block-based programming syntax, and cultural insensitivity issues when used for curriculum design or auto-grading [12, 33].

2.3 Teacher Agency vs. Automation

Using GenAI for lesson planning offers several advantages but also introduces a tension: *the increasing ability of these systems to handle routine instructional design tasks may undermine teacher autonomy* [3, 24]. This dilemma establishes a framework for developing AI-powered educational technology: AI interventions must augment, rather than supplant, the professional judgment of educators [15, 31]. Aligning with this framework, Ravi et al. collaborated with K-12 educators to build GenAI systems and proposed design guidelines [28]. They suggested positioning these systems as brainstorming partners and workflow scaffold while the human teachers remain in control. Teachers must define key pedagogical parameters (e.g., learning goals, curriculum standards) to prompt AI outputs and actively refine the resulting materials instead of adopted blindly.

Objective. To support K-12 CS teachers in customizing Scratch instructional materials for their classroom, we employed a design-based implementation research [9] approach to develop CONJUROR, an interactive tool powered by GenAI (GPT4 and Stable Image Core) [32]. CONJUROR’s architecture is fundamentally rooted in Ravi et al.’s paradigm of human-AI collaboration. This paper presents the results from a follow-up study investigating teachers’ experiences using CONJUROR during the school year to create and teach customized lessons in practice.

3 Methods

This study received institutional IRB approval. We now describe the customization process using CONJUROR, before providing details about our study procedure and data analysis approach.

3.1 Customization Process

CONJUROR customizes instructional materials from Scratch Encore (Encore), an intermediate CS curriculum [11], in several steps (see Figure 1). First, teacher create a classroom profile by answering questions about their students, including students’ demographics and cultural backgrounds, students’ interests and hobbies, significant local and school events, and topics in other school subjects that students are learning. They then choose a CS module (e.g., animation, conditional loops) to customize. Based on the teacher’s classroom profile, CONJUROR suggests ten themes (e.g., Lunar New Year, Monarch butterfly migration, Mardi Gras parade) that teachers could consider incorporating into their customized materials. Next, CONJUROR suggests four AI-generated, specific project ideas that fit with the chosen theme and module’s existing project. Teachers then choose or edit the specific project idea, and CONJUROR then searches through available Scratch library assets and suggests backdrop and sprite costumes relevant to the project idea teachers choose. Teachers can also request new AI-generated images for backdrop and sprite costumes if they would like to. Finally, CONJUROR remixes an Scratch project from Encore and edits the associate student-facing worksheets to match with the new, customized project. Teachers then refine the customized project (e.g., adjust sprite starting position, change values in code blocks) and worksheets (e.g., check for typos, updating wording if needed) before handing them out to students. It is important to note that, in all steps that involve interactivity with AI components, teachers have the options to retain the AI-generated suggestions, edit them, or replace them with their own ideas. More details on CONJUROR can be found in [32].

Table 1: Participant self-reported information and themes of their customized instructional materials

ID	Current role	Years teaching CS	Age	Gender	Racial identity	Customized lesson (CS topic × Theme)
ES1	Technology teacher	4	45-54	Female	White	Events × New York State map*
ES2	Technology teacher	12	55-64	Female	White	Animation × Oktoberfest market
ES3	Technology teacher	20	45-54	Female	Asian	Animation × Newton's 2nd law*
ES4	Interdisciplinary teacher	10	45-54	Female	PNS	Conditional loops × Hamantaschen baker*
ES5	Math/Science teacher	21	45-54	Female	PNS	Decomposition × Southern cooking*
ES6	CS teacher	12	35-44	Female	White	Animation × Spring*
ES7	Media specialist	13	45-54	Female	Black	Conditional loops × Frederick Douglass
ES8	Gen Ed teacher	8	45-54	Female	White	Events × Everglades ecosystem*
ES9	STEAM teacher	15	45-54	Male	White	Animation × Alaska gold rush*
ES10	Technology teacher	11	45-54	Female	White	Events × Historic Pullman*
ES11	Interdisciplinary teacher	3	25-34	Male	Asian	Animation × The Tortoise and the Hare*
ES12	Bilingual CS teacher	7	45-54	Female	Hispanic/Latinx	Conditional loops × Chicago architecture
ES13	STEAM teacher	4	35-44	Female	Black	Decomposition × Mardi Gras parade*
ES14	Math/Science teacher	4	35-44	Female	PNS	Decomposition × Roblox*
ES15	CS teacher	22	45-54	Male	Hispanic/Latinx	Animation × Día del Niño*
ES16	STEAM teacher	10	55-64	Female	White	Conditional loops × Harriet Tubman
MS1	Technology teacher	6	45-54	Female	Asian, NHPI	Events × Kwanzaa*
MS2	Media specialist	2	45-54	Female	White	Conditional loops × Minnesota's fur trade*
MS3	Technology teacher	5	PNS	Female	PNS	Animation × Dim sum restaurant*
MS4	Bilingual librarian	2	45-54	Female	Hispanic/Latinx	Events × Abuela's tamales
MS5	CS teacher	24	45-54	Female	Asian	Decomposition × Christmas*
MS6	CS teacher	8	55-64	Female	White	Conditional loops × Coyote in SF
MS7	Technology teacher	2	35-44	Male	Black	Conditional loops × Space game*
MS8	CS teacher	5	55-64	Male	White	Conditional loops × Summer vacation*
MS9	Math/Science teacher	13	44-54	Female	Black	Animation × East LA drag racing*

ES = elementary school teacher; MS = middle school teacher; PNS = prefer not to state; * = customized during school year

3.2 Study Procedure

To recruit participants, we sent emails to all teachers who had previously downloaded Encore and mailing lists for educators across the US. Teachers who completed an interest survey were invited to a professional development (PD) workshop in Summer 2025 based on grades taught and experience in Scratch and Encore. The PD equipped teachers with foundational knowledge about Encore (3 days) and why and how to use CONJUROR to customize Encore's materials (2 days). A total of 43 teachers completed all PD sessions and signed up to participate in a follow-up classroom implementation.

During the 2025-26 classroom implementation, participants were required to (1) complete a 30-min pre-interview to share their customizing and teaching plan, (2) teach at least three Encore modules of their choice; at least one needed to be customized using CONJUROR, and (3) complete a 60-min post-interview to reflect on their customizing and teaching experiences. Teachers were encouraged, but not required, to customize new modules beyond the ones they already customized during PD. During the school year, the research team provided additional support for teachers on request.

By the end of June 2026, 25 teachers completed the all study's requirements (see Table 1 for teacher information). Some had to drop out for various reasons, such as school districts banned Scratch, replacing colleagues who retired or resigned unexpectedly, or realizing they cannot commit to meeting all study's requirements.

3.3 Data Analysis

We analyzed transcripts of 25 post-implementation Zoom interviews. Each interview began with semi-structured questions asking teachers to reflect on their experiences in each step within CONJUROR (e.g., project brainstorming, choosing sprites) and describe how teaching the customized module(s) went (e.g., how they introduced materials to students, how students responded). We used a stimulated recall approach [25], showing teachers CONJUROR's automated interaction logs and asking them to explain the reasoning behind specific decisions they made working with the tool.

To analyze the data, we followed a three-cycle thematic coding process as described by Saldaña [29].

Cycle 1: Descriptive Coding. Three researchers familiarized themselves with the data by individually reviewing the same three transcripts, jotting down ideas that caught their attention, and meeting to discuss overarching takeaways.

Cycle 2: Axial Coding. The discussion at the conclusion of the first cycle evolved into a process of axial coding, where the team crossed and matched initial descriptive codes to form a preliminary codebook. They then tested this codebook on five additional transcripts and met to resolve discrepancies and refine the codes. The final codebook contained 14 codes grouped into four categories. Two researchers then split and coded the entire dataset (including re-coding the initial eight transcripts) to align the data with the final codebook. The axial code frequencies can be found in Table 2.

Cycle 3: Selective Coding. Finally, after two structural coding cycles, three researchers collaboratively reviewed the final codes, discussed similarities based on conceptual meaning, and synthesized the overarching themes presented in this paper.

Table 2: Axial Code Frequencies

Category	Code	Count	Total
Instructional design	Time	26	81
	Resources	30	
	Pedagogy	25	
Teacher roles	Coming up with theme	17	108
	Localizing theme	13	
	Evaluating AI suggestions	36	
	Prompting AI	21	
	Refine outputs	21	
Teacher learning	Pedagogical knowledge	5	33
	CS content knowledge	11	
	AI literacy knowledge	17	
Teaching opportunities	Teachable moments	9	28
	Supporting student learning	10	
	Co-customizing with students	9	

3.4 Limitations

Several methodological factors could have limited the results presented in this paper. First, we had limited visibility into who could sign to participate, given school privacy policies that restricted some teachers’ ability to use Scratch and meet the study’s requirements. Second, while our qualitative analysis was conducted by researchers from diverse backgrounds, it remains susceptible to researcher bias. To mitigate this, our coding process encompassed multiple coding cycles and the development of a structural codebook to ensure transparency, reliability, and accuracy. Finally, capturing teachers’ experiences relied on self-reported data from post-implementation interviews. We strengthened this by triangulating the self-reports with CONJUROR’s logs using a stimulated recall method [25].

4 Results and Discussion

We structure this section by first presenting how teachers positioned their roles in the process with CONJUROR (RQ1). Then, we describe how teachers were able to meet their instructional design goals (RQ2), followed by emergent knowledge and teaching opportunities teachers gained through the entire customizing and teaching experience (RQ3). Each subsection concludes with a paragraph discussing the key takeaways from our findings.

4.1 Positioning Teacher Roles

Our analysis showed that while customizing with CONJUROR, teachers took on different roles and degrees of involvement in the process. First, **teachers described CONJUROR as a “springboard” that helped them expand their original ideas, and positioned themselves as an “orchestrator” tailoring specific project elements to better represent their students’ contexts.** For example, ES1 wanted to create a customized project about ‘New York State’

(see Figure 2A). She typed in ‘New York State map’ and ‘New York State symbols’ into CONJUROR, explaining,

Fourth grade social studies is all about New York State so I thought it’d be cool to use that as the re-purposing of it... They do a lot of mapping for the state and another aspect of New York State to bring in is our own symbols. I knew I could do that with Scratch.

The tool then helped flesh out her ideas into four specific project descriptions, from which she chose one that included a tour guide, a big apple, the Statue of Liberty, a bear, and a background of the NY state map. Then, she edited the suggestion to make sprites compatible with local contexts. She changed the bear to a beaver, “It’s [the bear] supposed to be our state animal, which is the beaver... I don’t know where the bear came from.” She also replaced the tour guide with a Jell-O character, “The town where we are is home of Jell-O... that’s kind of our claim to fame. So I thought it would be cute to have a Jell-O character be the little narrator guy...”

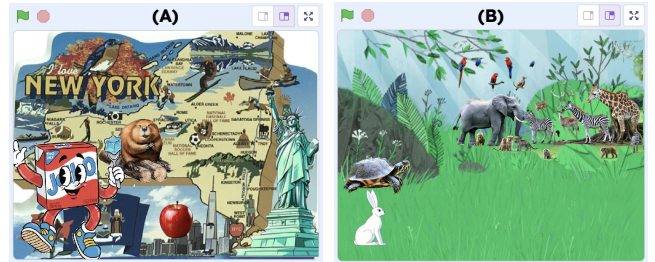


Figure 2: Examples of teachers’ customized Scratch projects

Teachers made critical judgments on when and how to use AI suggestions to ensure the customized materials were appropriate for students. For example, ES11, an interdisciplinary teacher, wanted to incorporate Language Arts topics into his customized Scratch project. He typed a detailed answer for one of the brainstorming questions: ‘Folktales carry cultural wisdom, humor, and values across generations. By bringing them into Scratch Encore, students practice computational storytelling while preserving cultural heritage. This activity allows students to honor their roots or explore new cultures while building creative coding skills’. After reviewing AI-suggested themes, ES11 selected ‘The Epic Journey of The Monkey King’ and continued to reviewing suggested project ideas. At this stage, he filtered the AI suggestions based on thematic content, and ultimately typed in his own project idea mimicking the children’s fable ‘The Tortoise and the Hare’ (Figure 2B). He reflected on his decision, “He [the Monkey King] steals something from someone... and I say, okay, this is stealing, I’ll not teach you [students], I will not choose this.”

Other teachers, like ES4, anticipated thematic complexity that would result in overly complex Scratch projects for students to learn, and decided to edit the AI suggestion,

I thought it would be too hard to make the [Purim costume] parade and the [Hamantaschen] baker together because they’re two different things... Both are part of the same holiday, but I thought it would be easier to focus on one.

Finally, **teachers emphasized their responsibility to assure the quality of customized materials and polish them before distributing to students.** ES15, expressed, *"It is a very powerful tool, but we have to make sure that we go in and change to make sure that everything is in the right place."* Another teacher, ES6, shared how she learned it was her responsibility to review the final resulting materials after discovering she missed errors during class, *"For myself as a learning curve, is to make sure that I'm checking the project and the worksheets fully before giving them out."*

Discussion: Rather than passively accepting AI-generated content, teachers actively negotiated their roles as "orchestrators" throughout the customization process. Aligning with the human-AI complementary design approach [15, 31], CONJUROR served as a "springboard" rather than a replacement for teacher expertise. By critically evaluating, modifying, or discarding AI suggestions, teachers ensured that the resulting customized materials remained authentic to their local contexts and pedagogically appropriate for their classrooms.

4.2 Meeting Instructional Design Goals

Beyond the expected improvements in time management enabled by many other AI tools, CONJUROR supported teachers in meeting unique and unimaginable instructional design goals.

4.2.1 Increasing customization process efficiency. Teachers described that **CONJUROR provided various complementary resources that made Scratch customization more achievable.** Regardless of their domain expertise, they were able to customize projects to meet their goals. For example, ES8, a general education teacher, emphasized how the project remix feature complimented her CS content knowledge, *"My technical knowledge is limited when it comes to the content, but it was nice to have a tool that took care of that for me."* ES15, an experienced CS teacher, highlighted how he enjoyed the AI support for project ideation, which empowered him to customize a project that connected better with students' cultures,

I wanted to teach a lesson during the week of Children's Day in Mexico... When I typed in 'Día del Niño', CONJUROR gave me different [project idea] options like parades... With the resources in Scratch, I couldn't modify something that really connects to the kids... I feel that 'this is the idea that I have behind' is exactly what I received from the tool.

Other teachers were excited about the new possibilities to make Scratch projects more appealing for students. MS7 claimed that the images generated by CONJUROR better aligned with some newer youth culture, *"I liked the sprites that I got. I enjoyed the clarity of the backdrop... Scratch has some bland things at times. I like that these added to the new age."* Teachers also noted that the visual resources beyond available Scratch assets added to student engagement when they taught the customized lesson. ES4 shared,

My students really enjoyed the new sprites and the new background. They've been working with Scratch for a few years, so they're used to all the same sprites and backgrounds. It's always the same stuff... So for me to have a baker, a doughball, a rolling pin and a

kitchen background that look nothing like Scratch, they were like, 'Where did you get this from?'

4.2.2 Pushing the boundaries of instructional goals. Our analysis revealed that **teachers were able to use CONJUROR in their classrooms beyond its intended purpose.** Six teachers used CONJUROR live to co-customize with students. The teachers acted as "drivers" (projecting the tool and showing students how it works), while students acted as "navigators" (making design choices such as editing project ideas and selecting sprite costumes). ES7 reflected on the co-customizing activity she did in class, *"I'll poll the class. Then I can put into the customization what they want, and then we see what it generates...from there, they can modify it... based upon what they wrote."* ES13 also co-customized with her students. She described how she was able to work with students on modifying sprite costumes in real time: *"[I] let them see it for themselves or how the system is working and let them get an idea to where, 'Hey, we wanted to add this. Let's make this a little bit bigger, a little smaller.'"*

Additionally, two teachers used CONJUROR to support students with special needs. MS1 found the brainstorming features helpful for students who usually struggled to come up with ideas for their own open-ended Scratch projects. She shared,

I started using CONJUROR for themes for students' projects. Very helpful for my students with IEPs because even though we go through the Use-Modify and the Create... they want to do something, but they can't say what they need to do. So now I have them... pulling out their cell phone to take a picture of the themes suggested... And they absolutely love it.

ES5 mentioned that her students with special accommodations and special needs were especially excited about what CONJUROR allowed them to do. She shared, *"Some of my 504 and IEP kids took to this easier than the kids that are not. So it kind of gave them the chance to be like, 'Hey, I can teach you a little something.'"* She added, *"It did my heart so good to see them become my little tutors."*

Discussion: Our findings reveal that interactive AI-powered tools designed to complement teachers' work, like CONJUROR, create opportunities for teachers to prioritize a student-centered instruction. Working with CONJUROR not only allowed teachers to meet their original instructional design goals (customizing existing resources to resonate with their students' cultures and experiences or existing curricular priorities in their school), but also allowed them to pivot nimbly to create dynamic classroom experiences (foster student agency through co-customizing and provide dedicated support for students with special needs).

4.3 Teacher Learning With and About AI

Beyond experiencing CONJUROR as a "customization helper", our participants discussed impromptu teacher knowledge they gained through working with it and how they leveraged their own learning experiences to encourage student learning.

4.3.1 Strengthening CS knowledge and teaching practices. Working through the ideation steps with CONJUROR **strengthened teachers' understanding of specific CS concepts they were going to teach.** For example, MS2 shared,

...definitely made me more intentional about conditional loops, and thinking about how to best apply that concept...If I would have just created any old project, I might not have taken the time to think about... how am I going to use this Scratch skill in a way that also goes with the story I'm telling with the project.

The workflow with CONJUROR also entailed greater attention to aspects that teachers might have previously overlooked: *reviewing and debugging code*. Our participants reported **improving their CS teaching practices by explicitly modeling the debugging process and normalizing mistakes for their students**. MS3 recalled *"It helped me... think through all the things that were happening for the students in terms of debugging... what they're going to think about, what they need to..."* While strengthening CS knowledge and teaching practices through working with CONJUROR was not expected, teachers realized that checking and reviewing the resulting materials is essential to maintain their skills and support students working with and learning from their customized materials.

4.3.2 Gaining AI literacy. Through interactively customizing with an AI-powered tool, teachers gained AI literacy. We captured multiple moments of such realization, *"I kept adding different descriptions to try to get closer to what I was looking for, and then I noticed that I could... get different results running the same description"* (MS6). Some teachers further embodied the opportunities to experience how AI works and realize its limitations, *"That's the other part with AI is you have to be as specific as possible. The more specific you are, the better the prompt, the better the product"* (ES1).

Initially, some teachers expected CONJUROR to automatically get all the customization tasks done for them perfectly. However, after using it more and connecting this experience to other AI tools they had used, teachers realized the limitations of AI and the boundaries they should place when customizing with CONJUROR. ES1 reflected,

It's like AI in general. It's not perfect. You always have to go in and tweak it and make it exactly what you want. But I found that a lot with AI that it's never perfect the first time.

For some of the teachers, it was their first time working with AI at all or to customize CS instructional resources for their own classrooms. They discussed the mystery that accompanied such work, learning to accept that the process may look different every time they use it, *"Sometimes I found CONJUROR to be like it has its own thing. Sometimes it gives me, sometimes it makes me wait"* (ES3). Teachers also reflected on their own limitations, sharing how they were unsure how to prompt the tool so that the outputs would meet their expectations, *"I know the Miguel [sprite name] was ended up generating very small, tiny and I don't know how to tell AI to keep aspect ratio, try to fill the space with the character"* (MS3).

4.3.3 Discussing and sharing about AI with students. Finally, while CONJUROR was designed as a teacher-facing tool, **teachers highlighted how working with the tool entailed opportunities for spontaneous, teachable moments in the classroom, when teaching with the resulting customized materials**. For instance, knowing her students often used or would become heavy AI users,

MS1 used her customized project to discuss AI etiquette with students and demonstrate how she was able to use an AI tool in a responsible way and what to pay attention to,

I'm not putting my focus on, 'Are you cheating? Are you doing this? Are you doing that?' but 'Look, I'm the one that's encouraging AI, use AI, figure out how it can make it work, because as we know, AI is not going to give you exactly perfect. So that means you gotta go in and debug.'

Another teacher, ES4, mentioned showing the AI tool to her students, discussing the opportunities it allows and demonstrating to her students that she uses AI, affirming she accepts and even encourages the responsible use of AI in her classroom,

After we did the project, we talked about AI. That's one of the things we cover. So I was like, 'Hey, look, this is an AI thing for Scratch.' And then they're like, 'Make me a character.' So then I'm making characters and downloading them... but they love the ability to have a character or a background generated.

Discussion: Our study shows how customizing instructional resources with interactive AI tools can help teachers strengthen CS knowledge and teaching practices, gain AI literacy, and raise critical ethical questions about AI with students. Teachers prioritized aspects they had previously given less attention to (e.g., debugging), which denotes an important role they play in the process when AI is involved. The hands-on learning from working with CONJUROR liberated teachers from perfection, and opened avenues for them to raise students' awareness about working with AI through their experiences. Importantly, our findings emphasize the integrity of teachers who took it upon themselves to discuss the benefits and pitfalls of AI with their students, showing them how using AI can bring both new challenges and opportunities.

5 Conclusion

This study explored how K-12 teachers experienced customizing Scratch instructional materials using an interactive tool powered by GenAI. Our findings highlight that teachers actively relied on their professional judgment to orchestrate AI-generated elements throughout the customization process, ensuring high-quality and locally relevant materials for their students. This AI-augmented workflow also empowered teachers to achieve unique and unimaginable instructional goals, such as co-customizing instructional materials with students during class time and supporting special needs learners. Further, navigating this human-AI collaboration unexpectedly enhanced teachers' own CS pedagogical knowledge and AI literacy, and sparked spontaneous classroom discussions about responsible AI use. This work contributes to our understanding of how GenAI-powered tools can be used to support K-12 CS teachers in creating customized instructional resources. Future research should explore how to further scaffold teacher agency and scale these customization practices across other computing curricula.

Acknowledgements

We would like to thank all participating teachers. This work is supported by the National Science Foundation Grant No. 2201313.

References

- [1] Brett A Becker, Paul Denny, James Finnie-Ansley, Andrew Luxton-Reilly, James Prather, and Eddie Antonio Santos. 2023. Programming is hard-or at least it used to be: Educational opportunities and challenges of ai code generation. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1*. 500–506.
- [2] Karen Brennan, Christan Balch, and Michelle Chung. 2014. *Creative computing: A design-based introduction to computational thinking*. <https://creativecomputing.gse.harvard.edu/guide/>
- [3] Ismail Celik, Muhterem Dindar, Hanni Muukkonen, and Sanna Järvelä. 2022. The promises and challenges of artificial intelligence for teachers: A systematic review of research. *TechTrends* 66, 4 (2022), 616–630.
- [4] Katharine Childs and Jane Waite. 2024. Funds of identity and culturally responsive computing: K-5 teachers' adaptations to computing resources. In *Proceedings of the 19th WiPSCE Conference on Primary and Secondary Computing Education Research*. 1–10.
- [5] Cynthia E Coburn and William R Penuel. 2016. Research–practice partnerships in education: Outcomes, dynamics, and open questions. *Educational researcher* 45, 1 (2016), 48–54.
- [6] Code.org. Retrieved 2025. *Computer Science Discoveries Curriculum*. <https://code.org/en-US/curriculum/computer-science-discoveries>
- [7] Merijke Coenraad, Jen Palmer, Donna Eater, David Weintrop, and Diana Franklin. 2022. Using participatory design to integrate stakeholder voices in the creation of a culturally relevant computing curriculum. *International Journal of Child-Computer Interaction* 31 (2022), 100353.
- [8] Andre Del Carpio Gutierrez, Paul Denny, and Andrew Luxton-Reilly. 2024. Evaluating Automatically Generated Contextualised Programming Exercises. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*. 289–295.
- [9] Barry J. Fishman, William R. Penuel, Anna-Ruth Allen, Britte Haugan Cheng, and Nora Sabelli. 2013. Design-Based Implementation Research: An Emerging Model for Transforming the Relationship of Research and Practice. *Teachers College Record* 115, 14 (Dec. 2013), 136–156. <https://doi.org/10.1177/016146811311501415> Publisher: SAGE Publications.
- [10] Diana Franklin, Paul Denny, David A Gonzalez-Maldonado, and Minh Tran. 2025. *Generative AI in computer science education: Challenges and opportunities*. Cambridge University Press.
- [11] Diana Franklin, David Weintrop, Jennifer Palmer, Merijke Coenraad, Melissa Cobian, Kristan Beck, Andrew Rasmussen, Sue Krause, Max White, Marco Anaya, et al. 2020. Scratch Encore: The design and pilot of a culturally-relevant intermediate Scratch curriculum. In *Proceedings of the 51st ACM technical symposium on computer science education*. 794–800.
- [12] David Gonzalez-Maldonado, Jonathan Liu, and Diana Franklin. 2025. Evaluating GPT for use in K-12 Block Based CS Instruction Using a Transpiler and Prompt Engineering. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1*.
- [13] Joanna Goode and Jane Margolis. 2011. Exploring computer science: A case study of school reform. *ACM Transactions on Computing Education (TOCE)* 11, 2 (2011), 1–16.
- [14] Google. Retrieved 2023. *CS First: A computer science curriculum that makes coding easy to teach and fun to learn*. <https://csfirst.withgoogle.com/s/en/home>
- [15] Kenneth Holstein and Vincent Alevan. 2022. Designing for human–AI complementarity in K-12 education. *Ai Magazine* 43, 2 (2022), 239–248.
- [16] Xinying Hou, Zihan Wu, Xu Wang, and Barbara J Ericson. 2024. CodeTailor: Personalized Parsons Puzzles are Preferred Over AI-Generated Solutions to Support Learning. *arXiv preprint arXiv:2401.12125* (2024).
- [17] Garima Jain, Fred Martin, Bernardo Feliciano, Hsien-Yuan Hsu, Barbara Fauvel-Campbell, Gillian Bausch, Lijun Ni, and Elizabeth Thomas-Cappello. 2022. CS pathways: A culturally responsive computer science curriculum for middle school. In *2022 IEEE Frontiers in Education Conference (FIE)*. IEEE, 1–9.
- [18] Mengying Jiang and Kristin A Searle. 2024. Developing Culturally Responsive-Sustaining Computer Science Curricula: A Design-Based Research Study. In *Proceedings of the 18th International Conference of the Learning Sciences-ICLS 2024*, pp. 1630-1633. International Society of the Learning Sciences.
- [19] Mollie Jordan, Kevin Ly, and Adalbert Gerald Soosai Raj. 2024. Need a Programming Exercise Generated in Your Native Language? ChatGPT's Got Your Back: Automatic Generation of Non-English Programming Exercises Using OpenAI GPT-3.5. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*. 618–624.
- [20] Breanna Jury, Angela Lorusso, Juho Leinonen, Paul Denny, and Andrew Luxton-Reilly. 2024. Evaluating llm-generated worked examples in an introductory programming course. In *Proceedings of the 26th Australasian Computing Education Conference*. 77–86.
- [21] Juho Leinonen, Paul Denny, Stephen MacNeil, Sami Sarsa, Seth Bernstein, Joanne Kim, Andrew Tran, and Arto Hellas. 2023. Comparing code explanations created by students and large language models. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1*. 124–130.
- [22] Evanfiya Logacheva, Arto Hellas, James Prather, Sami Sarsa, and Juho Leinonen. 2024. Evaluating Contextually Personalized Programming Exercises Created with Generative AI. In *Proceedings of the 2024 ACM Conference on International Computing Education Research-Volume 1*. 95–113.
- [23] Stephen MacNeil, Andrew Tran, Arto Hellas, Joanne Kim, Sami Sarsa, Paul Denny, Seth Bernstein, and Juho Leinonen. 2023. Experiences from using code explanations generated by large language models in a web software development e-book. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1*. 931–937.
- [24] Joseph Mintz, Wayne Holmes, Leping Liu, and Maria Perez-Ortiz. 2023. Artificial Intelligence and K-12 education: Possibilities, pedagogies and risks. , 325–333 pages.
- [25] John O'Brien. 1993. Action research through stimulated recall. *Research in science education* 23, 1 (1993), 214–221.
- [26] William R Penuel. 2017. Research–practice partnerships as a strategy for promoting equitable science teaching and learning through leveraging everyday science. *Science Education* 101, 4 (2017), 520–525.
- [27] James Prather, Juho Leinonen, Natalie Kiesler, Jamie Gorson Benario, Sam Lau, Stephen MacNeil, Narges Norouzi, Simone Opel, Vee Pettit, Leo Porter, et al. 2025. Beyond the Hype: A Comprehensive Review of Current Trends in Generative AI Research, Teaching Practices, and Tools. *2024 Working Group Reports on Innovation and Technology in Computer Science Education* (2025), 300–338.
- [28] Prerna Ravi, John Masla, Gisella Kakoti, Grace C. Lin, Emma Anderson, Matt Taylor, Anastasia K. Ostrowski, Cynthia Breazeal, Eric Klopfer, and Hal Abelson. 2025. Co-designing Large Language Model Tools for Project-Based Learning with K12 Educators. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, 1–25. <https://doi.org/10.1145/3706598.3713971>
- [29] Johnny Saldaña. 2021. *The coding manual for qualitative researchers*. SAGE publications Ltd.
- [30] Sami Sarsa, Paul Denny, Arto Hellas, and Juho Leinonen. 2022. Automatic generation of programming exercises and code explanations using large language models. In *Proceedings of the 2022 ACM Conference on International Computing Education Research-Volume 1*. 27–43.
- [31] Kyoungwon Seo, Mina Yoo, Samuel Dodson, and Sung-Hee Jin. 2025. Augmented teachers: K–12 teachers' needs for artificial intelligence's complementary role in personalized learning. *Journal of Research on Technology in Education* 57, 4 (2025), 876–893.
- [32] Minh Tran, David Gonzalez-Maldonado, and Diana Franklin. 2026. Conjuror: A GenAI-Powered Tool to Support Teachers Customizing a Scratch Curriculum. In *Proceedings of the 31st ACM Conference on Innovation and Technology in Computer Science Education V. 1*. 450–456.
- [33] Minh Tran, David Gonzalez-Maldonado, Elaine Zhou, and Diana Franklin. 2025. Can GPT Help? Supporting Teachers to Brainstorm Customized Instructional Scratch Projects. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1*.
- [34] Minh Tran, Heather Killen, Jen Palmer, David Weintrop, and Diana Franklin. 2024. Harmonizing Scratch Encore: Scaffolding K-8 Teachers in Customizing Culturally Responsive Computing Materials. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*. 1335–1341.
- [35] Aman Yadav and Marc Berges. 2019. Computer science pedagogical content knowledge: Characterizing teacher performance. *ACM Transactions on Computing Education (TOCE)* 19, 3 (2019), 1–24.
- [36] Aman Yadav, Sarah Gretter, Susanne Hambrusch, and Phil Sands. 2016. Expanding computer science education in schools: understanding teacher experiences and challenges. *Computer science education* 26, 4 (2016), 235–254.