An Iterative Design ‘by proxy’ Method for Developing Educational Music Interfaces

Corey Ford
Creative Technologies Laboratory
Department for Computer Science
and Creative Technologies
UWE Bristol
corey2.ford@live.uwe.ac.uk

Chris Nash
Creative Technologies Laboratory
Department for Computer Science
and Creative Technologies
UWE Bristol
chris.nash@uwe.ac.uk

ABSTRACT

Iterative design methods involving children and educators are difficult to conduct, given both the ethical implications and time commitments understandably required. The qualitative design process presented here recruits introductory teacher training students, in order to discover useful design insights relevant to music education technologies ‘by proxy’. Thus, some of the barriers present in child-computer interaction research are avoided.

As an example, the method is applied to the creation of a block-based music notation system, named Codetta. Building upon successful educational technologies that intersect both music and programming, Codetta seeks to enable child composition, whilst aiding generalist educators’ confidence in teaching music.

Author Keywords

iterative design, music education, child-computer interaction, block-based programming, qualitative methods, educational technology, think-aloud, discourse analysis

CCS Concepts

•**Applied computing** → **Interactive learning environments;** *Sound and music computing;* •**Human-centered computing** → **HCI design and evaluation methods;**

1. INTRODUCTION

When designing an interactive system, a user-centred approach is paramount. In particular, iterative approaches emphasise the importance of the end-user, involving them continuously throughout the development process [18]. However, in the field of child-computer interaction (CCI) there are many barriers towards adopting such an approach. For example, continually working alongside children has understandable ethical implications [21]. Furthermore, the design process must be adopted by educators who, already, are struggling to balance a challenging workload [6, 4].

The method presented and discussed in this paper aims to gather valuable insights towards designing interfaces for music education, whilst avoiding the ethical challenges and time constraints present in CCI research. This method does not intend to replace child-led design processes, nor belittle the importance of ethical approval. Instead, the intention is to contribute an approach to support the development of working prototypes, particularly in situations where engaging with children is very difficult or impossible.

To demonstrate the method, it is applied to the development of a block-based music composition tool, named Codetta. Codetta looks to increase child participation in music composition, in part focusing on the rationale that many generalist educators lack confidence in teaching music; they believe that music is a skill only for knowledgeable specialists [9, 11] and, thus, are deterred from engaging in open-ended activities (such as composition) [10, 8].

In summary, Codetta is developed alongside two student-teacher collaborators instead of directly involving child end-users. An iterative design process is used, triangulating the think-aloud protocol, interviews and discourse analysis (DA). Although the design implications stemming from the student-teacher’s feedback presented here apply to Codetta, the method is applicable for designing any educational music interface.

NIME’20, July 21-25, 2020, Royal Birmingham Conservatoire, Birmingham City University, Birmingham, United Kingdom.

2. BACKGROUND

Before starting to design digital systems for learning music, it is important to consider how children acquire musical skills, and how their skills are developed in pedagogies.

Often, music pedagogies delay the learning of notation and music theory in place of learning musical structures implicitly by ear. For example, a concept central to Gordon’s Music Learning Theory is audiation: one’s ability to recall sounds in their head when no actual sound is present. Gordon argues that audiation is a key characteristic in determining a child’s musical aptitude, and that musical aptitude is not highly correlated with intelligence nor academic achievement [22].

The Kodály Method is a pedagogy which follows a child-developmental approach [7]. This means that the pedagogical content is arranged to follow children’s growing abilities; initially, emphasis is placed on learning rhythms (via physical activities) before moving onto understanding melodies (via singing activities).

In both of these pedagogies, children first develop their basic musical awareness before accumulating enough tools to engage in composition activities. With regard to developing Codetta — where the goal is to facilitate children in composition digitally — the design process must be applied with the assumption that it is possible to adequately accelerate childrens’ musical skills using technology.

2.1 LOGO-Inspired Systems

The MIT mathematician Papert created one of the first digital solutions for engaging children in open-ended tasks [17], which is regarded as a seminal work in CCI. Named LOGO,

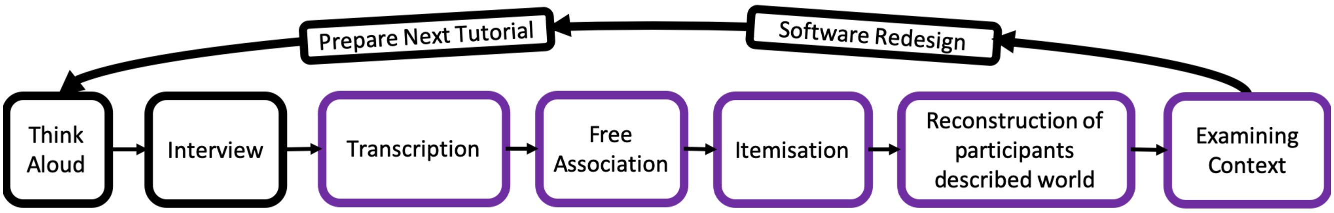

Figure 1: A flow-chart showing the steps of the iterative design method. Those in purple are from [20].

the system allowed children to tinker with simple commands that controlled the movement of a virtual 'turtle'; as the turtle moved it drew lines onto the screen. Thus, as children played with LOGO and explored the many shapes that were possible to draw, they unwittingly accelerated their mathematical, problem-solving and computational-thinking skills.

With regard to music composition, LOGO-inspired systems are ideal as they afford both problem-seeking and problem-solving. Thus, many music-specific adaptations have been successfully developed (see [12]).

Scratch [13] is a block-based system inspired by LOGO that was designed to teach newcomers to programming. In Scratch, users control a 'sprite' by snapping together colourful puzzle-shaped blocks. Widely adopted to support delivery of the computer science curriculum, the block-based drag-and-drop GUI supports ease of use, with the notches, bumps and colours of blocks signifying how they interact with each other [23].

Codetta builds on Scratch, placing the block-based programming paradigm within the context of music creation. The software aims to capitalise on the playful nature of LOGO-inspired systems, whilst taking advantage of the learnability of a block-based graphical interface. It is theorised that the character of LOGO-inspired systems will allow children to compose without needing to build traditional musical skills. Instead, children will intrinsically explore the system and discover what works. A full epistemological discussion is beyond the scope of this paper, although, the interested reader is pointed to [17].

Notably, one example of an explorable interface that has successfully helped novices compose is Manhattan [14]. Manhattan uses the spreadsheet paradigm to embed high-level formulas within a trackers low-level music editing system and, thus, affords a scalable challenge.

2.2 Related Design Methods

Many CCI researchers have explored various design processes to develop educational technologies for creative disciplines. Some methodologies are discussed here with the aim to simply emphasise the challenges of CCI research. An in-depth critique is not intended.

Within computer science, Gomes, Falcão and Tedesco [19] explored how the interaction elements of digital games could contribute to childrens' learning of programming concepts. Notably, many of the games investigated followed the block-based paradigm. Their method consisted of using unstructured interviews, semi-structured interviews and participant observation, which successfully contributed many design ideas. However, the generalisability of the study is limited to the group investigated (four private school groups, aged 5-7 years, living in Brazil), and a considerable commitment was required of both the children's and educator's time.

Using the novelty of music to motivate the learning of computer science concepts, Sonic Pi [1] is a domain specific programming language that has been successfully integrated into a creative pedagogy [2]. The language was commissioned by the Raspberry Pi Foundation who, due to their strategic ambitions, allocated only three-weeks for development. Advantageously, this meant that less burden was placed on the involvement of children and educators. On the other hand, many initial design decisions (such as the ruby-based syntax) consequently coincided with the developer's current research and expertise, but not end-user feedback.

In music education, Nouwen *et al.* [16] developed a participatory design method for creating an educational digital music game, specifically focusing on singing pitches in time with a given rhythm. Although many useful design requirements were found, the development process was not without difficulty: midway through development project partners changed, shifting focus from ages 8-12 to 6-10.

3. METHOD

In developing Codetta, the aim was to devise an iterative design process which placed the majority of the burden on the researcher and did not require undergoing a lengthy ethical approval process. However, the method must still adequately provide insights that are relevant for designing a child-friendly learnable system.

A qualitative approach was devised such that a (resource efficient) smaller-sample size could elicit rich and detailed descriptive data. Two first year BA(Hons) Primary Education students were recruited following an advertisement at the end of an introductory lecture on 'Teaching and Learning'. They will be referred to as P1 and P2 throughout this paper. Theoretically, both the participant's knowledge of taught degree content and experience from primary-school placements would contribute valuable insight concerning Codetta's design, whilst the participants were broadly more available than practising educators.

A brief questionnaire to gather information about the participant's prior musical and technical experience was conducted. The participants were confident about teaching in general, but not so confident in teaching music and even less confident writing music. Additionally, the participants self-doubted their musical experiences, adding comments undermining their achievements. For example, P2 noted that "it's been a while" since they played piano.

In order to ensure that the procedure was mindful of the student-teacher's workload, four 45 minute iterations were to be conducted at times organised with the participants. The session length was agreed to by the head of the university's 'Education and Childhood' department prior to recruitment. Additionally, a pilot session was conducted before the student-teacher's sessions, using two music technology students.

For each iteration, a process was followed triangulating the think-aloud protocol, interviews and Stowell, Plumbley and Bryan-Kinn's DA method [20]. The process is summarised here, with an overview shown in Figure 1. For more precise detail about the individual steps of the DA method, the reader is pointed to [20].

3.1 Think-Aloud Protocol

The think-aloud protocol was chosen for multiple reasons. Firstly, by externalising their thought processes, a deep insight could be obtained from the participants. The participants burden was minimised also; they were only required to perform tasks once whilst their interactions could be replayed by the researcher at any time. Plus, the researcher provided the equipment and controlled the environment, meaning the participants did not have to prepare anything beforehand.

For each session, the student-teachers followed a written tutorial prepared by the researcher, focusing on a particular musical concept. Built into the software, each tutorial ensured that any new developments to the program, or devised hypothesis, were tested. Although no exact plan was followed, the researcher's background likely had a passive influence on the devised didactic material. Furthermore, the participants were isolated from one another and recorded in a random order each time.

3.2 Interview

After the think-aloud, the researcher would also ask some questions to probe any theories uncovered by the results from the previous iteration. Thus, there is a further opportunity for the researcher to gather opinions on design changes or hypothesises. The interviews were informal, short and purposeful (lasting approximately 5 minutes).

3.3 Transcription & Itemisation

The transcription of the think-aloud and interviews provided the raw text for the DA. By performing the full transcription (including noting all visual detail), the researcher advantageously immersed themselves in the text [3]. Therefore, they also noted their immediate thoughts after transcription (see 'Free Association' in [20]).

The Stowell, Plumbley and Bryan-Kinns's DA method [20] was chosen as it targets the evaluation of musical interfaces whilst, like the think-aloud, it provides a low cost in participants' time and resources. To put it simply, the DA method required the researcher to break each 'object' down into an itemised list, in order to identify the most common occurrences in the text.

3.4 Reconstruction of Described World & Examining Context

Finally (as in [20]), a reconstruction of each participant's world is written up as a text description, before being combined into a single comparison ('examining context'). Here the aim is to provide enough detail to develop hypotheses and software implementations for the next iteration.

4. RESULTS

In this section, the highlights of the findings from each iteration are presented, alongside the goals the researcher set for the iteration and the focus of the tutorial. A summary of the key results and consequent design implications are found in Table 1. As it is challenging to visualise the interactions with Codetta from just this paper alone, the reader is invited to view the video demonstrations[1].

4.1 First Iteration (Pilot)

As shown in Figure 2, the first prototype for Codetta consisted of a basic drag and drop interface, where blocks could be connected to one another, notated (using a click-based music notation engine) and executed by pressing play (or

Figure 2: The initial interface used in the pilot study.

cancelled using the stop button). A starting block is indicated by a speaker icon, which needs to be connected to a bar (containing music notation) in order to create sound.

The goal for this iteration was to test the devised methodology with pilot study participants (music technology students). Towards this, the tutorial explored the first steps needed to create sound using Codetta and asked the participants to recreate twinkle-twinkle little star, building their notation skills.

The use of traditional music notation is somewhat inspired by Manhattan [14], aiming to provide a low-level editing mechanism, in order to support novice users. Hence, notes are added to a bar by selecting the desired note length from a pop-up menu (shown in Figure 3), and then the pitch can be changed via two up and down arrows.

4.2 Second & Third Iteration

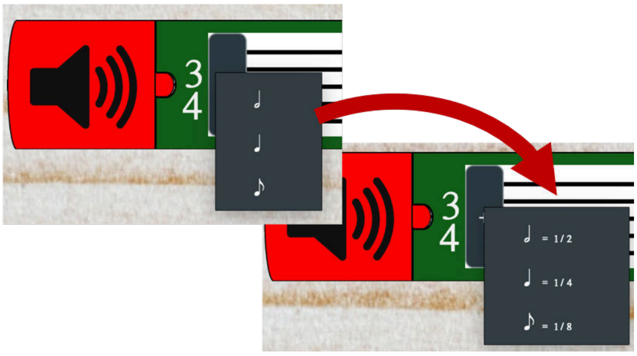

Figure 3: The second iteration's design change for the representation of note values.

The goal for the second iteration was to gather first impressions from the student-teacher participants, using the same tutorial as in the pilot study. As outlined in Table 1, initial usability issues were successfully discovered through the DA. Notably, clues were drawn from participant's body language that were helpful in judging their confidence levels. Furthermore, it was found that P2 was much more comfortable using numeric terminology, informing the revised pop-up menu design (see Figure 3).

The third iteration provided an opportunity to investigate a frequent behaviour exhibited by both P1 and P2. When composing, the participants repeatedly directed their gaze between the tutorial and the workspace. When asked about this during the interview phase, both participants independently agreed that this was not problematic.

[1]https://codetta.codes/NIME2020/

Table 1: The findings for each iteration alongside the consequent design implications.

Iteration	Findings	Design Implications
1 (Pilot)	The note lengths shown on the pop-up menu were hard to differentiate between.	The pop-up menu size was increased.
	The tutorial was difficult to read when positioned horizontally.	The position of the tutorial's viewport was moved to a vertical pane.
2	The participants spent lots of time darting between the tutorial and workspace.	This was queried in the next iteration.
	Terminology was frequently misused. P2 preferred to use numeric terminology (such as quarter note or half note).	The numeric fractions for note values were added to the pop-up menu.
	The participants continually disconnected and reconnected bars from the starting block, in order to compare different notes.	
	The participant's body language suggested when they were struggling; P1 would frown or shake their head and P2 would occasionally purse their lips.	
3	P2 praised the consistent use of mouse-based interactions as "you have no idea how much experience a kid will have [typing]".	
	The participants praised the addition of fractional values to the pop-up menu. Notably, P1 suggested that the fractions could push stronger children "if they did not already know about these".	
	The participants found it difficult to distinguish between tempo-related blocks.	One of these blocks was distinctly coloured orange.
	P1 praised the "little symbol" on the block that controls the global project tempo, suggesting it helps to denote the block's purpose.	All of the instrument blocks implemented in the next iteration were given a "little symbol".
	The participants still struggled with terminology.	An info-bar was developed, displaying extra information for each of Codetta's interactions.
4	The info-bar was praised by both participants.	
	The colour of blocks helped participants avoid having to use precise, specialist terminology.	
	To change the global project tempo, both participants needed to follow a cumbersome sequence of interactions, starting with dragging a 'global tempo' block into the workspace.	The 'global tempo' block was replaced with a static widget in the top right corner of the interface. This made it clearer that one object controlled the *entire* project tempo and minimised the number of clicks that were required.
	The participants were happiest when listening back to their creations, particularly when exploring different timbres.	The play button was enlarged and made a brighter colour, in order to encourage the participants to use it more frequently.
5	The task set was very challenging.	
	The usage of terminology was much stronger in this session, with P1 faultlessly referring to time signatures by name (four-four, three-four). Codetta's support for this commended was the use of colour, organisation of blocks and the info-bar.	
	P2 often remedied their gaps in knowledge by using the play button to continually confirm the results of their tweaking. P1 also realised similarly that the play button was useful for checking music as it is entered.	
	The participants still had some difficulty in distinguishing between the (yellow) tempo-related blocks.	The global tempo widget and other tempo-related blocks were all coloured 'honeydew', so that they were distinct, but also clearly related.
	The introduction of clefs was the most confusing concept so far.	
	The global tempo widget was praised by P1 and P2.	

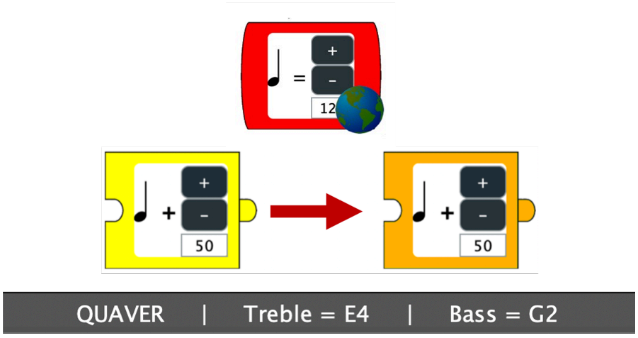

Figure 4: The third iteration's "little globe symbol" (top - see Table 1), design changes for the time-related blocks (middle) and info-bar (bottom).

Furthermore, the tutorial engaged the participants in exploring time (tempo) in music. The final task set was to recreate Steve Reich's Piano Phase as its simple process yields impressive results with minimal effort, providing motivation to early-stage learners [15]. Even without a prior understanding of time-signatures or metre, the participants were able to understand and practically apply such concepts by tinkering with Codetta's tempo-related blocks. However, the participants found it hard to distinguish between the different time-related blocks. Thus, one block was distinctly coloured orange (from yellow) (see Figure 4).

As alluded to in the previous iteration, using terminology was a struggle for P1 and P2 throughout the process also. Thus, an info-bar was added to the bottom of Codetta, adding extra information to all interactions (see Figure 4).

4.3 Fourth & Fifth Iteration

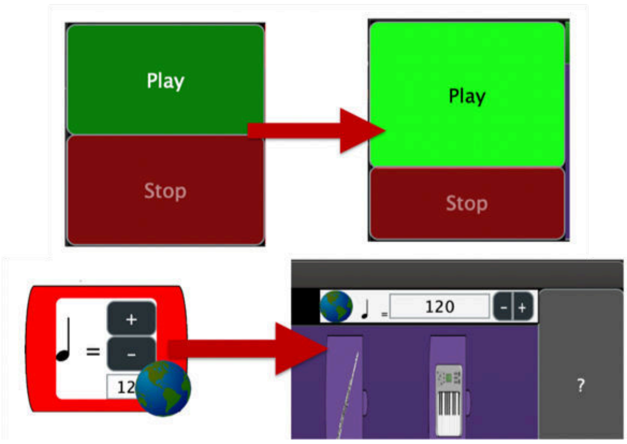

Figure 5: The fourth iteration's design changes for both the playback and tempo controls (described in Table 1).

For the fourth iteration, focus was placed on understanding which elements of the interface the participants found most engaging, whilst the tutorial focused on dynamic contrasts and different timbres. The previous iterations had focused on time and pitch, so timbre was an unexplored dimension of music. It was found that the participants were happiest when hearing their creations, particularly when exploring different timbres. P1 asked to try out more instruments during the interview, and P2 exclaimed "Woah, we have different instruments now! That is so cool!" when first discovering the feature.

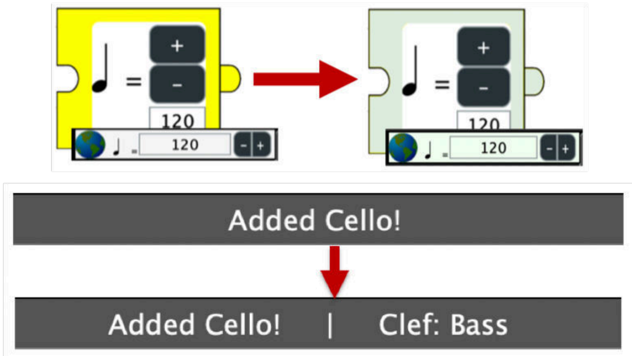

Figure 6: The fifth iteration's design changes for the tempo blocks and info-bar.

In the fifth iteration, the goal was to re-assess the participant's comfort with specialist terms. Both bass and treble clefs were also introduced in the tutorial, however, were challenging for the participants to understand. It is possible that such a concept is too disengaged from current child-friendly pedagogies and more related to traditional music theory. To remedy this, the info-bar was updated to display a recommended clef (see Figure 6). Furthermore, there was still some confusion in distinguishing between the different time-related blocks; the colour of the tempo-related UI features were consequently matched (see Figure 6).

5. DISCUSSION & FUTURE WORK

This paper contributes an iterative design method that was used to develop Codetta: a block-based music system to support children in composition. The devised method triangulated the think-aloud protocol with DA and interviews, yielding useful design recommendations for educational musical interfaces via 'proxy' users. This was achieved without obtruding on educator's already challenging workloads, or undergoing a lengthy ethical review process.

The think-aloud protocol successfully produced rich and detailed descriptive results, capturing both the participant's interactions and affective responses whilst using Codetta. Notably, facial expressions provided useful non-verbal clues that would not have been captured by other methods (such as data logging).

However, the iterative design method placed a large burden on the researcher who had to not only transcribe and analyse each think-aloud recording, but also develop the software and accompanying tutorials within a single iteration. This would be challenging to maintain over a longer period, or with a larger number of participants. Alternatively, each task could be spread across several researchers and/or accelerated using data analysis software.

Furthermore, a lightweight alternative to the analytically heavy DA technique may elicit just as useful results. Thematic analysis (as in [3]), for example, is a faster and easier to understand alternative, which has been successfully used by researchers intersecting interaction design and music [5].

Although the 'proxy' design method avoids many of the challenges present in CCI research, it does not encompass the opinions of actual children, nor practising educators. It is recommended that the method is applied in tandem with other evaluative studies, in order to ensure the end-user is still heavily involved in the development process. Indeed, ongoing work with Codetta aims to evaluate its real-world use in the classroom. The results of such work aim to not just understand Codetta's ability to support child interaction in digital composition, but also validate the effectiveness of the iterative method presented here.

A further weakness of this work is that the proposed iterative design process was applied to an already functioning base of a LOGO-inspired composition tool. Inevitably, the researchers own background and opinions would have (albeit passively) informed this design. Researchers adopting this method should bear this in mind and strive to be as reflexive as possible. Moreover, directly applying this method to the assumption that LOGO systems are a good fit for music education would strengthen this work.

Overall, the proposed method provides an effective solution to developing software for child users, whilst avoiding many of the barriers slowing the process of conducting research involving children. If we want to nurture childrens' creativity, why not ensure that, as developers/researchers, we can use multiple efficient techniques, in order to provide children with the best creative tools?

6. ACKNOWLEDGEMENTS

The authors would like to thank the participants and lecturers from the University of the West of England's education department for supporting this interdisciplinary research.

7. ETHICAL STANDARDS

This study was approved by the University of the West of England's ethics committee. All participants provided informed consent.

8. REFERENCES

[1] S. Aaron and A. F. Blackwell. From sonic Pi to overtone: creative musical experiences with domain-specific and functional languages. In *FARM '13: Proceedings of the first ACM SIGPLAN workshop on Functional art, music, modeling & design*, pages 35–46, New York, USA, 2013. ACM Press.

[2] S. Aaron, A. F. Blackwell, and P. Burnard. The development of Sonic Pi and its use in educational partnerships: Co-creating pedagogies for learning computer programming. *Journal of Music, Technology and Education*, 9(1):75–94, may 2016.

[3] V. Braun and V. Clarke. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2):77–101, jan 2006.

[4] S. Bridges and A. Searle. Changing workloads of primary school teachers: 'I seem to live on the edge of chaos'. *School Leadership & Management*, 31(5):413–433, nov 2011.

[5] N. Bryan-Kinns, W. Wang, and Y. Wu. Thematic Analysis for Sonic Interaction Design. In *Proceedings of British HCI 2018*, pages 1–3. BCS Learning and Development Ltd., 2018.

[6] G. Butt and A. Lance. Modernizing the roles of support staff in primary schools: changing focus, changing function. *Educational Review*, 57(2):139–149, may 2005.

[7] L. Choksy. *The Kodály Method: Comprehensive Music Education From Infant To Adult.* Prentice-Hall Inc, Englewood Cliffs, New Jersey, 1974.

[8] S. Eastburn, T. Floyd, A. Gowan-Webster, V. Johnson, S. Mackay, and J. Robinson. National Music Educators' Survey: Findings. Technical report, London, 2019.

[9] S. Hennessy. Overcoming the red-feeling: the development of confidence to teach music in primary school amongst student teachers. *British Journal of Music Education*, 17(2):183–196, jul 2000.

[10] M. Hickey and S. D. Lipscomb. How different is good? How good is different? The assessment of children's creative musical thinking. In I. Deliège and G. Wiggins, editors, *Musical Creativity: Multidisciplinary Research in Theory and Practice*, chapter 6, pages 97–110. Psychology Press, East Sussex, 2006.

[11] H. Holden and S. W. Button. The teaching of music in the primary school by the non-music specialist. *The British Journal of Music Education*, 23(1):23–38, 2006.

[12] S. Holland. Artificial Intelligence in music education : a critical review. In E. R. Miranda, editor, *Readings in Music and Artificial Intelligence (Contemporary Music Studies)*, chapter 13, pages 239–274. Routledge, New York, 2000.

[13] J. Maloney, M. Resnick, N. Rusk, B. Silverman, and E. Eastmond. The scratch programming language and environment. *ACM Transactions on Computing Education*, 10(4):16, 2010.

[14] C. Nash. Manhattan: End-User Programming for Music. In K. Tahiroğlu, R. Fiebrink, A. Tana, and B. Caramiaux, editors, *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 221–226, London, 2014. Goldsmiths.

[15] C. Nash. Manhattan: Serious Games for Serious Music. In *Music, Education and Technology (MET)*, London, 2016. Sempre.

[16] M. Nouwen, S. Schepers, K. Mouws, K. Slegers, N. Kosten, and P. Duysburgh. Designing an educational music game: What if children were calling the tune? *International Journal of Child-Computer Interaction*, 9-10:20–32, dec 2016.

[17] S. Papert. *Mindstorms: children, computers, and powerful ideas.* Basic Books, New York, 1980.

[18] Y. Rogers, H. Sharp, and J. Preece. *Interaction Design: Beyond human-computer interaction.* John Wiley & Sons, Inc., West Sussex, 3rd edition, 2011.

[19] T. C. Simões Gomes, T. Pontual Falcão, and P. Cabral de Azevedo Restelli Tedesco. Exploring an approach based on digital games for teaching programming concepts to young children. *International Journal of Child-Computer Interaction*, 16:77–84, jun 2018.

[20] D. Stowell, M. D. Plumbley, and N. Bryan-Kinns. Discourse analysis evaluation method for expressive musical interfaces. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 81–86, Casa Paganini, Genova Italy, 2008.

[21] K. Tisdall, J. M. Davis, and M. Gallagher. *Researching with Children and Young People: Research Design, Methods and Analysis.* Sage Publications, 2008.

[22] D. L. Walters. Edwin Gordon's music aptitude work. *The Quarterly Journal of Music Teaching and Learning*, 2(1-2):65–73, 1991.

[23] D. Weintrop and U. Wilensky. To block or not to block, that is the question: students' perceptions of blocks-based programming. In *Proceedings of the 14th International Conference on Interaction Design and Children - IDC '15*, pages 199–208, New York, USA, 2015. ACM Press.